

## Technische Spezifikation KLE 2.0

Version 20260306

Anforderungen Receiver

Diese Richtlinien wurden in Zusammenarbeit mit den Mitgliedern des Vereins Swissdec erarbeitet:

- Suva
- Verein eAHV / IV mit den Mitgliedern:
  - Konferenz der kantonalen Ausgleichskassen
  - Vereinigung der Verbandsausgleichskassen
- Schweizerische Steuerkonferenz mit den Mitgliedern:
  - Eidgenössische Steuerverwaltung
  - Kantonale Steuerverwaltungen
- Bundesamt für Statistik
- Schweizerischer Versicherungsverband

**Herausgeber**

Swissdec  
Fluhmattstrasse 1  
6004 Luzern

<https://www.swissdec.ch>

Versionsgeschichte

|  |            |          |
|--|------------|----------|
| Version 2.0                              | 06.03.2026 | ile, mas |
| Erste Ausgabe Spezifikation für KLE 2.0. |            |          |

**Konventionen in diesem Dokument**

Folgende Schriftarten werden in diesem Dokument verwendet:

|        |                                   |
|--------|-----------------------------------|
| Text   | Dokumentation                     |
| Text   | Code                              |
| <Text> | XML-Element                       |
| [Text] | Referenz auf ein anderes Dokument |

Die Verbindlichkeit von Anforderungen ist wie folgt definiert:

| Verbindlichkeit | Wort                 |
|-----------------|----------------------|
| Pflicht         | <b>muss</b>          |
| Wunsch          | <i>soll (sollte)</i> |
| Absicht         | <i>wird</i>          |
| Vorschlag       | <i>kann</i>          |

Tabelle 1. Verbindlichkeit von Anforderungen

**Achtung**

Für das konzeptionelle Verständnis werden teilweise schematische Darstellungen verwendet, die nicht zwingend den aktuellen Stand widerspiegeln.

Verbindlich sind ausschliesslich die jeweils veröffentlichten offiziellen XML-Schemata.

**Achtung**

Bei Abweichungen zwischen der deutschen, französischen und italienischen Ausgabe der Richtlinien ist die deutsche Version massgebend.

# Inhaltsverzeichnis

|  |    |
|--|----|
| 1. Einleitung .....  | 1  |
| 1.1. Aufbau des Dokuments .....  | 1  |
| 1.2. Leitgedanke Bausteine .....   | 2  |
| 2. Geschäftsprozess zur Übermittlung von Leistungsfällen mittels KLE .....           | 3  |
| 3. Use Cases und zugehörige Operationen .....  | 5  |
| 3.1. Übersicht .....   | 5  |
| 3.2. Grafische Darstellung der Geschäftsprozesse und ihren Use Cases .....           | 5  |
| 3.3. Uses Cases und ihre Operationen .....   | 7  |
| 4. Ping .....  | 8  |
| 5. RegisterOrganizationAuthentication .....  | 12 |
| 5.1. Übersicht .....   | 12 |
| 5.2. Initiale synchrone Meldung .....  | 12 |
| 5.3. Datenstruktur der initialen Meldung .....                                       | 14 |
| 5.4. Datenstruktur der initialen Antwort .....                                       | 15 |
| 5.5. Datenstruktur der Synchronize-Meldung .....                                     | 16 |
| 5.6. Datenstruktur der Synchronize-Antwort .....                                     | 16 |
| 6. SubscribeOrganization .....   | 17 |
| 6.1. Übersicht .....   | 17 |
| 6.2. Initiale synchrone Meldung .....  | 17 |
| 6.3. Datenstruktur der initialen Meldung .....                                       | 19 |
| 6.4. Datenstruktur der initialen Antwort .....                                       | 20 |
| 6.5. Datenstruktur der Synchronize-Meldung .....                                     | 21 |
| 6.6. Datenstruktur der Synchronize-Antwort .....                                     | 21 |
| 7. DeclareIncident .....   | 23 |
| 7.1. KLE Ereignismeldung .....   | 23 |
| 7.2. Initiale synchrone Meldung .....  | 23 |
| 7.3. Ablauf / Protokoll .....  | 25 |
| 7.4. Datenstruktur der initialen Meldung .....                                       | 26 |
| 7.5. Datenstruktur der initialen Antwort .....                                       | 27 |
| 7.6. Datenstruktur der Synchronize-Meldung .....                                     | 27 |
| 7.7. Datenstruktur der Synchronize-Antwort .....                                     | 28 |
| 8. Use Cases .....   | 29 |
| 8.1. UC001 Initiale Meldung empfangen .....  | 29 |
| 8.2. UC002 Duplikate behandeln .....   | 31 |
| 8.3. UC003 Testmeldung empfangen .....   | 32 |
| 8.4. UC004 Ersatzmeldung verarbeiten .....   | 32 |
| 8.5. UC005 Synchronisieren .....   | 33 |
| 8.6. UC006 Prozesskontrolle durchführen .....  | 35 |
| 8.7. UC007 Fall schliessen .....   | 36 |
| 8.8. UC008 Stories verarbeiten und quittieren .....                                  | 36 |
| 8.9. UC009 Stories melden .....  | 37 |
| 8.10. UC010 Datenflusskontrolle .....  | 38 |
| 8.11. UC011 Completion .....   | 38 |
| 8.12. UC012 Dialog Message abwickeln .....   | 42 |
| 8.13. UC013 Anmeldeprozess (SubscribeOrganizationConsumer) .....                     | 42 |
| 8.14. UC014 Unternehmensauthentifizierung (RegisterOrganizationAuthentication) ..... | 43 |
| 8.15. UC015 Security anwenden .....  | 43 |
| 8.16. UC016 Wartungsfenster .....  | 44 |
| 8.17. UC017 Supportanfrage bearbeiten .....  | 46 |
| 8.18. UC018 Erreichbarkeit prüfen .....  | 48 |
| A. Referenzierte Dokumente .....   | 51 |
| B. Glossar .....   | 52 |
| C. Detailspezifikation Swissdec Unternehmens-Authentifizierung SUA .....             | 54 |
| C.1. Einleitung .....  | 54 |

|   |     |
|---|-----|
| C.2. SUA Prozess .....  | 54  |
| C.3. Technische Anforderungen an die Zertifikate .....                      | 56  |
| D. Detailspezifikation DialogMessages .....                                 | 59  |
| D.1. Einleitung .....   | 59  |
| D.2. Die Verwendung von DialogMessages .....                                | 59  |
| D.3. Standard DialogMessages .....  | 65  |
| E. Detailspezifikation Completion URL .....                                 | 66  |
| E.1. Aufbau und Verarbeitung der Completion URL .....                       | 66  |
| F. Bausteine .....  | 69  |
| F.1. AB-01 Konfiguration .....  | 69  |
| F.2. AB-02 Erreichbarkeit .....   | 69  |
| F.3. AB-03 Interoperabilität .....  | 69  |
| F.4. AB-04 Anmeldung .....  | 70  |
| F.5. AB-05 Explizite Adressierung (direkte Adressierung) .....              | 70  |
| F.6. AB-06 Implizite Adressierung (Gruppenadressierung) .....               | 71  |
| F.7. AB-07 Absichtserklärung .....  | 72  |
| F.8. AB-08 Identifikation eines Geschäftsvorfalls .....                     | 72  |
| F.9. AB-09 Ersatzmeldung .....  | 73  |
| F.10. AB-10 Request und Response IDs .....                                  | 73  |
| F.11. AB-11 Testmeldung .....   | 74  |
| F.12. AB-12 Plausibilisierung .....   | 74  |
| F.13. AB-13 Filterung .....   | 75  |
| F.14. AB-14 Versionsmapping .....   | 76  |
| F.15. AB-15 Doublettenerkennung .....                                       | 76  |
| F.16. AB-16 Anonymisierung .....  | 76  |
| F.17. AB-17 Meldungsfragmentierung .....                                    | 77  |
| F.18. AB-18 Swissdec Unternehmensauthentifizierung (SUA) .....              | 77  |
| F.19. SB-01 Transportsicherheit (HTTPS / TLS) .....                         | 77  |
| F.20. SB-02 Transport Authentication (TLS Mutual Authentication) .....      | 78  |
| F.21. SB-03 Nutzdatenverschlüsselung (WS-Encryption) .....                  | 78  |
| F.22. SB-04 Nutzdatensignatur (WS-Signature) .....                          | 78  |
| F.23. SB-05 Unternehmens-Authentifizierung SUA Signatur (WS-Signatur) ..... | 79  |
| F.24. SB-06 Nichtabstreitbarkeit .....                                      | 79  |
| F.25. PB-01 Synchrone Verteilung .....                                      | 79  |
| F.26. PB-02 Asynchrone Verteilung .....                                     | 80  |
| F.27. PB-03 Synchronisierung .....  | 80  |
| F.28. PB-04 Dialogisierung .....  | 81  |
| F.29. PB-05 Completion .....  | 81  |
| F.30. PB-06 Prozessnavigation .....   | 82  |
| F.31. PB-07 Digitalisierungsbereich .....                                   | 82  |
| F.32. PB-08 Datenanfordern .....  | 83  |
| G. Technische Dokumentationen .....   | 84  |
| G.1. Schema Dokumentation IncidentDeclarationConsumerServiceTypes.xsd ..... | 84  |
| G.2. Schema Dokumentation IncidentDeclarationConsumerContainer.xsd .....    | 84  |
| G.3. Schema Dokumentation IncidentDeclarationContainer.xsd .....            | 87  |
| G.4. Schema Dokumentation IncidentDeclaration.xsd .....                     | 95  |
| G.5. Schema Dokumentation SwissdecComponents.xsd .....                      | 117 |
| G.6. Schema Dokumentation Common.xsd .....                                  | 132 |

## Abbildungsverzeichnis

|  |    |
|--|----|
| 2.1. Prozessübersicht Swissdec KLE-Standard .....  | 3  |
| 3.1. Use Cases Initiale Meldung .....  | 5  |
| 3.2. Use Cases Synchronisierung .....  | 6  |
| 3.3. Andere Use Cases .....  | 6  |
| 4.1. Schemabild PingConsumer .....   | 8  |
| 4.2. Schemabild PingConsumerResponse .....   | 10 |
| 5.1. Synchroner Meldungsprozess .....  | 13 |
| 5.2. Schemabild RegisterOrganizationAuthenticationConsumer .....                         | 14 |
| 5.3. Schemabild Addressee .....  | 15 |
| 5.4. Schemabild RegisterOrganizationAuthenticationConsumerResponse .....                 | 15 |
| 5.5. Schemabild SynchronizeRegisterOrganizationAuthenticationConsumer .....              | 16 |
| 5.6. Schemabild SynchronizeRegisterOrganizationAuthenticationConsumerResponse .....      | 16 |
| 6.1. Synchroner Meldungsprozess .....  | 18 |
| 6.2. Schemabild SubscribeOrganizationConsumer .....                                      | 19 |
| 6.3. Schemabild Addressee .....  | 20 |
| 6.4. Schemabild SubscribeOrganizationConsumerResponse .....                              | 20 |
| 6.5. Schemabild SynchronizeSubscribeOrganizationConsumer .....                           | 21 |
| 6.6. Schemabild SynchronizeSubscribeOrganizationConsumerResponse .....                   | 21 |
| 7.1. Synchroner Meldungsprozess .....  | 24 |
| 7.2. DeclareIncident Protokollstatus .....   | 25 |
| 7.3. Schemabild DeclareIncidentConsumer .....  | 26 |
| 7.4. Schemabild Addressees .....   | 27 |
| 7.5. Schemabild DeclareIncidentConsumerResponse .....                                    | 27 |
| 7.6. Schemabild SynchronizeDeclareIncidentConsumer .....                                 | 27 |
| 7.7. Schemabild SynchronizeDeclareIncidentConsumerResponse .....                         | 28 |
| 8.1. URL-Encoding in der Completion .....  | 41 |
| 8.2. Aufbau URL gemäss RFC3986 .....   | 41 |
| 8.3. Schemabild PingConsumer .....   | 45 |
| 8.4. Schemabild PingConsumerResponse .....   | 46 |
| D.1. DialogMessage XML-Schema Struktur .....   | 60 |
| D.2. Vorgehen bei der Darstellung der DialogMessage (Aktivitäts Diagramm) .....          | 63 |
| D.3. Paragraph/Answer/DateTime-Element, andere Answer/<Type>-Elemente entsprechend ..... | 64 |
| D.4. Struktur des DialogMessage/Previous-Elementes .....                                 | 64 |
| E.1. URL-Encoding in der Completion .....  | 67 |
| E.2. Aufbau URL gemäss RFC3986 .....   | 68 |
| F.1. AddresseeType .....   | 71 |

## Tabellenverzeichnis

|   |    |
|---|----|
| 1. Verbindlichkeit von Anforderungen .....  | iv |
| 1.1. Beteiligte am Swissdec-Übermittlungsprozess .....                                      | 1  |
| 3.1. Use Cases und Operationen .....  | 7  |
| 4.1. Feldbeschreibungen PingConsumer .....  | 8  |
| 4.2. Feldbeschreibungen UserAgent .....   | 9  |
| 4.3. Feldbeschreibungen PingConsumerResponse .....  | 10 |
| 4.4. Feldbeschreibungen UserAgent .....   | 10 |
| 5.1. Feldbeschreibungen RegisterOrganizationAuthenticationConsumer .....                    | 14 |
| 5.2. Feldbeschreibungen RequestContext .....  | 14 |
| 5.3. Feldbeschreibungen Job .....   | 15 |
| 5.4. Feldbeschreibungen Addressee .....   | 15 |
| 5.5. Feldbeschreibungen RegisterOrganizationAuthenticationConsumerResponse .....            | 15 |
| 5.6. Feldbeschreibungen SynchronizeRegisterOrganizationAuthenticationConsumer .....         | 16 |
| 5.7. Feldbeschreibungen SynchronizeRegisterOrganizationAuthenticationConsumerResponse ..... | 16 |
| 6.1. Feldbeschreibungen SubscribeOrganizationConsumer .....                                 | 19 |
| 6.2. Feldbeschreibungen RequestContext .....  | 19 |
| 6.3. Feldbeschreibungen Job .....   | 20 |
| 6.4. Feldbeschreibungen Addressee .....   | 20 |
| 6.5. Feldbeschreibungen SubscribeOrganizationConsumerResponse .....                         | 20 |
| 6.6. Feldbeschreibungen SynchronizeSubscribeOrganizationConsumer .....                      | 21 |
| 6.7. Feldbeschreibungen SynchronizeSubscribeOrganizationConsumerResponse .....              | 22 |
| 7.1. Feldbeschreibungen DeclareIncidentConsumer .....                                       | 26 |
| 7.2. Feldbeschreibungen RequestContext .....  | 26 |
| 7.3. Feldbeschreibungen Job .....   | 27 |
| 7.4. Feldbeschreibungen Addressees .....  | 27 |
| 7.5. Feldbeschreibungen DeclareIncidentConsumerResponse .....                               | 27 |
| 7.6. Feldbeschreibungen SynchronizeDeclareIncidentConsumer .....                            | 28 |
| 7.7. Feldbeschreibungen SynchronizeDeclareIncidentConsumerResponse .....                    | 28 |
| 8.1. UC001 Initiale Meldung empfangen .....   | 29 |
| 8.2. UC002 Duplikate behandeln .....  | 31 |
| 8.3. UC003 Testmeldung empfangen .....  | 32 |
| 8.4. UC004 Ersatzmeldung verarbeiten .....  | 32 |
| 8.5. UC005 Synchronisieren .....  | 33 |
| 8.6. UC006 Prozesskontrolle durchführen .....   | 35 |
| 8.7. UC007 Fall schliessen .....  | 36 |
| 8.8. UC008 Stories verarbeiten und quittieren .....   | 36 |
| 8.9. UC009 Stories melden .....   | 37 |
| 8.10. UC010 Datenflusskontrolle .....   | 38 |
| 8.11. UC011 Completion .....  | 38 |
| 8.12. UC012 Dialog Message abwickeln .....  | 42 |
| 8.13. UC013 Anmeldungsprozess (SubscribeOrganizationConsumer) .....                         | 42 |
| 8.14. UC014 Unternehmensauthentifizierung (RegisterOrganizationAuthentication) .....        | 43 |
| 8.15. UC015 Security anwenden .....   | 43 |
| 8.16. UC016 Wartungsfenster .....   | 44 |
| 8.17. UC017 Supportanfrage bearbeiten .....   | 46 |
| 8.18. UC018 Erreichbarkeit prüfen .....   | 48 |
| F.1. AB-01 Konfiguration .....  | 69 |
| F.2. AB-02 Erreichbarkeit .....   | 69 |
| F.3. AB-03 Interoperabilität .....  | 69 |
| F.4. AB-04 Anmeldung .....  | 70 |
| F.5. AB-05 Explizite Adressierung (direkte Adressierung) .....                              | 70 |
| F.6. AB-06 Implizite Adressierung (Gruppenadressierung) .....                               | 71 |
| F.7. AB-07 Absichtserklärung .....  | 72 |
| F.8. AB-08 Identifikation eines Geschäftsvorfalles .....                                    | 72 |



|   |    |
|---|----|
| F.9. AB-09 Ersatzmeldung .....  | 73 |
| F.10. AB-10 Request und Response IDs .....                                  | 73 |
| F.11. AB-11 Testmeldung .....   | 74 |
| F.12. AB-12 Plausibilisierung .....   | 74 |
| F.13. AB-13 Filterung .....   | 75 |
| F.14. AB-14 Versionsmapping .....   | 76 |
| F.15. AB-15 Doublettenerkennung .....                                       | 76 |
| F.16. AB-16 Anonymisierung .....  | 76 |
| F.17. AB-18 Swissdec Unternehmensauthentifizierung (SUA) .....              | 77 |
| F.18. SB-01 Transportsicherheit (HTTPS / TLS) .....                         | 77 |
| F.19. SB-02 Transport Authentication (TLS Mutual Authentication) .....      | 78 |
| F.20. SB-03 Nutzdatenverschlüsselung (WS-Encryption) .....                  | 78 |
| F.21. SB-04 Nutzdatensignatur (WS-Signature) .....                          | 78 |
| F.22. SB-05 Unternehmens-Authentifizierung SUA Signatur (WS-Signatur) ..... | 79 |
| F.23. SB-06 Nichtabstreitbarkeit .....                                      | 79 |
| F.24. PB-01 Synchrone Verteilung .....                                      | 79 |
| F.25. PB-02 Asynchrone Verteilung .....                                     | 80 |
| F.26. PB-03 Synchronisierung .....  | 80 |
| F.27. PB-04 Dialogisierung .....  | 81 |
| F.28. PB-05 Completion .....  | 81 |
| F.29. PB-06 Prozessnavigation .....   | 82 |
| F.30. PB-07 Digitalisierungsbereich .....                                   | 82 |
| F.31. PB-08 Datenanfordern .....  | 83 |

## 1 Einleitung

Dieses Dokument enthält funktionale, technische und zusätzliche Anforderungen an Endreceiver, die Daten vom Swissdec Distributor erhalten wollen. Es adressiert die technischen Aspekte, nicht die fachliche Logik. Ein Endreceiver wird dazu verwendet, Meldungen, die vom Sendersystem via Distributor elektronisch versendet wurden, zu empfangen und auf diese zu reagieren.

Es sind die mitgeltenden Dokumente im Anhang zu beachten. Besonders in den fachlichen Richtlinien sind für den Endreceiver wesentliche Aspekte mit spezifiziert.

Hierbei ist es wichtig, zwischen den verschiedenen Akteuren im Swissdec-Prozess zu unterscheiden:

|                 |  |
|-----------------|--|
| Sendersystem    | Beim Sendersystem handelt es sich um ein System, das Daten aufbereitet und für den Versand an Endempfänger bereitstellt. Hier werden vor allem die fachlichen Anforderungen technisch korrekt umgesetzt (Beispiel: ERP).   |
| Transmitter     | Der Transmitter übermittelt die vom Sendersystem bereitgestellten Daten an den Swissdec Distributor und empfängt die erhaltenen Antworten (Responses), um diese zu verifizieren und an das Sendersystem weiterzuleiten.  |
| Distributor     | Der Distributor ist das zentrale System, das Daten vom Transmitter erhält, validiert, plausibilisiert und an die adressierten Endreceiver weiterleitet, sowie die erhaltenen Antworten zurück an den Transmitter sendet. Er übernimmt die Filterung und Verteilung der Daten.                                |
| Endreceiver     | Der Endreceiver ist das technische Gegenstück zum Transmitter. Er empfängt und validiert die vom Distributor erhaltenen Daten und leitet diese an ein Empfängersystem weiter. Die vom Empfängersystem bereitgestellten Antworten werden danach vom Endreceiver in die Response an den Distributor eingefügt. |
| Empfängersystem | Das Empfängersystem erhält seine Daten vom Endreceiver um diese dann auf fachlicher Ebene verarbeiten zu können. Es stellt ausserdem die Antworten bereit, die danach über den Endreceiver zurück an den Distributor und das Sendersystem übermittelt werden sollen. (Beispiel: Versicherer, Behörde)        |

Tabelle 1.1. Beteiligte am Swissdec-Übermittlungsprozess

Die elektronische Übermittlung erfolgt via Webservice (SOAP) und erfolgt jeweils via Request-Response-Kommunikation. Die Kommunikation findet jeweils zwischen Transmitter und Distributor sowie zwischen Distributor und Endreceiver statt.

Dieses Dokument behandelt die Schnittstelle zwischen Distributor und Endreceiver, nicht aber jene zwischen Transmitter und Distributor.

### 1.1 Aufbau des Dokuments

Dieses Dokument beschreibt im Nachfolgenden:

- Die Geschäftsprozesse, die den zu zertifizierenden Standard definieren. Dieses Kapitel beschreibt, wie die fachlich definierten Prozesse aus technischer Sicht umgesetzt werden.
- Die Operationen, die es ermöglichen, die Geschäftsprozesse abzubilden.
- Die Use Cases, die die Grundlage für die Operationen bilden.
- Im Anhang befinden sich ausserdem detaillierte Beschreibungen der Bausteine, aus denen die Bausteine definiert werden.

Der Grundgedanke bei der Beschreibung der Operationen besteht darin, einen Bezugspunkt zwischen fachlicher Dokumentation und technischer Spezifikation wie WSDL und XSD zu liefern. Dank grafischer Darstellung der XSDs und Beschreibungen der einzelnen Elemente, sollten die Operationen auf verständliche Weise den Geschäftsprozessen zugeordnet werden können.

Die Use Cases gehen von einem grundlegend technischen Ansatz aus und beschreiben die diversen Anforderungen an die Übermittlungen im Detail. Hier werden zusätzlich zum erfolgreichen Durchlaufen eines Prozesses auch mögliche alternative Abläufe und Fehlersituationen beschrieben. Ausserdem findet man hier zusätzliche Anforderungen und Informationen zur Umsetzung von Spezialfällen.

Die Use Cases setzen sich zwar aus den Bausteinen zusammen, aber es ist nicht zwingend notwendig, alle einzelnen Bausteine zu kennen und zu verstehen. Ihre Auflistung im Anhang dient nur zum besseren Verständnis, wie der vorliegende Swissdec Standard aufgebaut ist.

## 1.2 Leitgedanke Bausteine

Sämtliche Swissdec Standards basieren auf Bausteinen, die für alle Standards gleich sein müssen. Diese Bausteine sind Grundkonstrukte, auf denen die Use Cases aufbauen.

Es gibt drei Gruppen von Bausteinen:

- **Architekturbausteine:**

Die Architekturbausteine bilden das "Fundament" und beschreiben die grundlegenden Funktionen, die für die Umsetzung eines Standards nötig sind.

- **Securitybausteine:**

Die Securitybausteine ermöglichen eine sichere Übermittlung zwischen allen beteiligten Systemen. Sie beschreiben sowohl das Anwenden von Security-Elementen auf die zu übermittelnden Daten, als auch die Analyse der Security auf erhaltenen Daten.

- **Prozessbausteine:**

Die Prozessbausteine bieten Grundbausteine für das Zusammenfügen eines Übermittlungsprozesses. Sie beschreiben die Grundelemente, aus denen die verschiedenen Übermittlungschoreografien eines Standards zusammengestellt werden können.

## 2 Geschäftsprozess zur Übermittlung von Leistungsfällen mittels KLE

Der KLE Standard ist Teil des Datenraums Leistung und unterstützt den Meldungs-Prozess bei Unfall oder Krankheit. Der SOLL-Prozess berücksichtigt dabei vier Rollen:

- der/die MitarbeiterIn (indirekt)
- das Unternehmen (Arbeitgeber)
- der Versicherer
- Dritte wie der Leistungserbringer

Der Schwerpunkt der Kommunikation liegt zwischen Unternehmen und Versicherer.

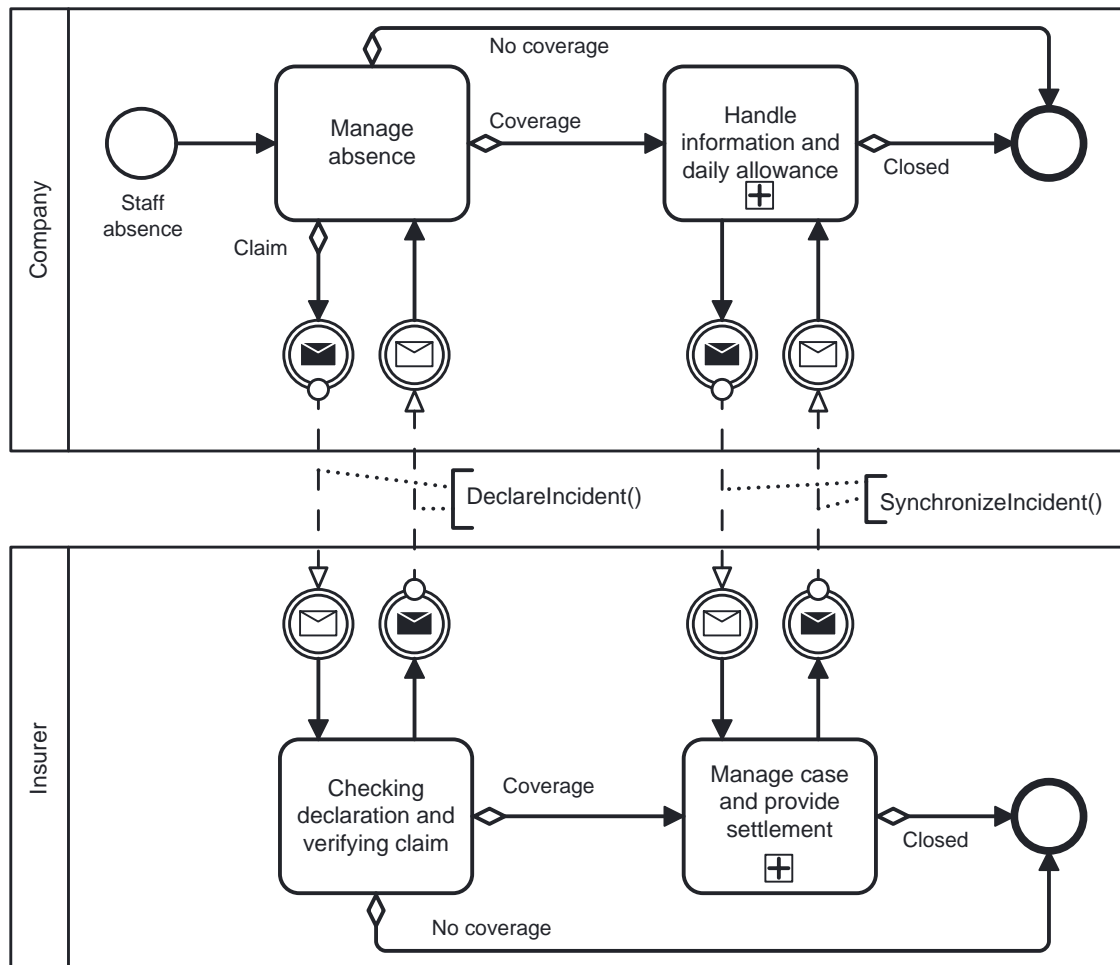


Abbildung 2.1. Prozessübersicht Swissdec KLE-Standard

Der SOLL-Prozess beschreibt den standardisierten Ablauf eines Leistungsfalls vom Ereignis bis zur Leistungserbringung. Auslöser ist, dass ein Mitarbeitender aufgrund von Krankheit, Unfall, Mutterschaft oder Rückfall arbeitsunfähig wird und das Unternehmen informiert. Das Unternehmen prüft im Absenzenmanagement, ob eine Meldung nötig ist, und erfasst die Ereignismeldung direkt im Swissdec-zertifizierten ERP-System. Wird dem Versicherer von Dritten eine Meldung auf separatem Weg zugestellt und kein entsprechendes Ereignis erfasst ist, dann informiert der Versicherer das Unternehmen, das wiederum im Rahmen des Absenzenmanagements die Anmeldung beim Versicherer prüft.

Sobald der Versicherer die Ereignismeldung empfängt, wird ein Dossier eröffnet und die entsprechende *InsuranceCaseID* dem Unternehmen zurückgemeldet. Im Falle eines Rückfalls kann auch ein bestehendes Dossier reaktiviert werden. Falls Informationen fehlen, kann das Unternehmen, der/die MitarbeiterIn oder Drit-

te (Ärzte) zusätzliche Angaben liefern. Liegen dem Versicherer alle Unterlagen vor, entscheidet dieser über den Leistungsanspruch.

Wird der Anspruch anerkannt, beginnt die Fallführung und die Leistungserbringung durch den Versicherer. Währenddessen tauschen Unternehmen und Versicherer fortlaufend Ereignisinformationen aus, wie Änderungen der Arbeitsunfähigkeit oder Rückfragen. Auch der Mitarbeitende ist verpflichtet über Genesung oder Arbeitsfähigkeit zu informieren.

Der Datenaustausch zwischen Unternehmen und Versicherer erfolgt in strukturierter Form während des gesamten Fallverlaufs, bei dem beide Seiten den gleichen Informationsstand zum Ereignis aufrechterhalten. Änderungen im Verlauf des Falls werden nicht als spezielle Korrekturmeldungen behandelt, sondern durch wiederholte Übermittlung aktualisierter Daten synchronisiert. Der Absender übermittelt dabei jeweils vollständige Informationen, während der Empfänger diese mit seinen vorhandenen Daten vergleicht und Abweichungen als Anpassungen erkennt. Ergänzend zeigt ein Statusmechanismus dem Unternehmen jederzeit den Bearbeitungsstand des Ereignisses und die noch offenen Schritte im Prozess.

Zur eindeutigen Zuordnung eines Ereignisses verwendet KLE zudem mehrere Fallkennungen auf Ebene Unternehmen (CompanyCaseID), Versicherer (InsuranceCaseID) und Übermittlung (IncidentCaseID). Diese IDs werden während der gesamten Fallführung mitgeführt und stellen sicher, dass alle Beteiligten denselben Leistungsfall korrekt referenzieren können.

Der KLE-Prozess kann ausschliesslich durch den Versicherer abgeschlossen werden, sobald der Mitarbeitende wieder arbeitsfähig ist, austritt, verstirbt, sämtliche Taggelder vollständig ausgerichtet wurden, eine erste Rente zugesprochen wird oder der Versicherer das Ereignis ablehnt.

## 3 Use Cases und zugehörige Operationen

### 3.1 Übersicht

Ein Anwendungsfall (Use Case) bündelt alle möglichen Szenarien, die eintreten können, wenn ein Akteur versucht, mithilfe des betrachteten Systems ein bestimmtes fachliches Ziel zu erreichen. Er beschreibt, was inhaltlich bei dem Versuch der Zielerreichung passieren kann und abstrahiert von konkreten technischen Lösungen.

In diesem Kapitel werden die Geschäftsprozesse des Leistungsstandards KLE in Use Cases aufgeteilt, die es ermöglichen sollen, die Prozesse mit möglichen Alternativszenarien abzubilden.

Während die Operationen standardspezifisch sind, werden die Use Cases generisch gehalten und können in verschiedenen Swissdec-Standards äquivalent wiederverwendet oder eingeschränkt werden.

### 3.2 Grafische Darstellung der Geschäftsprozesse und ihren Use Cases

Die initiale Meldung besteht darin, das Ereignis vom Transmitter an den Distributor zu übermitteln ([Abschnitt 8.1, „UC001 Initiale Meldung empfangen“](#)). Die Meldung kann dabei als Testfall ([Abschnitt 8.3, „UC003 Testmeldung empfangen“](#)) gekennzeichnet werden. Für eine erfolgreiche Übermittlung müssen Request und Response den Security-Richtlinien entsprechen ([Abschnitt 8.15, „UC015 Security anwenden“](#)), was im Falle der Response vom Transmitter verifiziert werden **muss**.

Die Use Cases [Abschnitt 8.2, „UC002 Duplikate behandeln“](#) und [Abschnitt 8.4, „UC004 Ersatzmeldung verarbeiten“](#) werden von KLE nicht unterstützt.

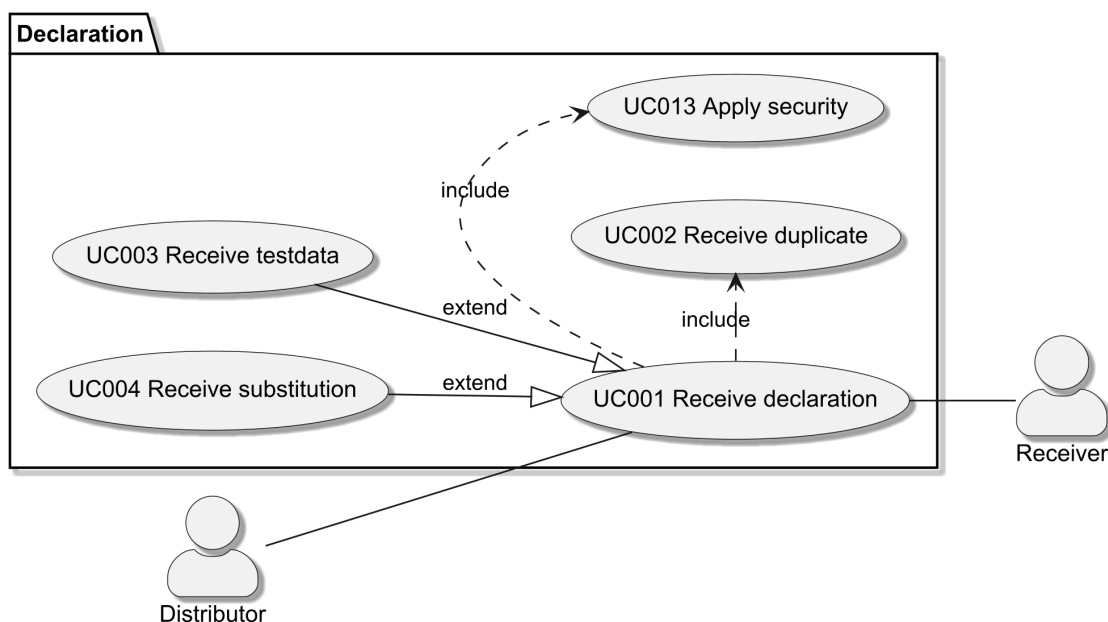


Abbildung 3.1. Use Cases Initiale Meldung

Nach erfolgreicher Deklaration werden Synchronisierungen ([Abschnitt 8.5, „UC005 Synchronisieren“](#)) durchgeführt, währenddessen werden Stories gemeldet und quittiert ([Abschnitt 8.8, „UC008 Stories verarbeiten und quittieren“](#)), aber auch beim Empfänger abgeholt und verarbeitet ([Abschnitt 8.9, „UC009 Stories melden“](#)). Dies soll anhand der Prozesskontrolle ([Abschnitt 8.6, „UC006 Prozesskontrolle durchführen“](#)) funktionieren, wobei [Abschnitt 8.12, „UC012 Dialog Message abwickeln“](#) auch Teil davon ist. Der Geschäftsfall wird solange synchronisiert bis der Fall mit dem State *CommunicationStopped* abgeschlossen wird ([Abschnitt 8.7, „UC007 Fall schliessen“](#)).

Um eine Überlastung der betroffenen Systeme zu verhindern, können alle beteiligten Systeme den Datenfluss kontrollieren ([Abschnitt 8.10, „UC010 Datenflusskontrolle“](#)). Für eine erfolgreiche Synchronisierung müssen

Request und Response den Security-Richtlinien entsprechen ( [Abschnitt 8.15, „UC015 Security anwenden“](#) ), was im Falle der Response vom Transmitter verifiziert werden **muss**.

Der Use Case [Abschnitt 8.11, „UC011 Completion“](#) wird von KLE nicht unterstützt.

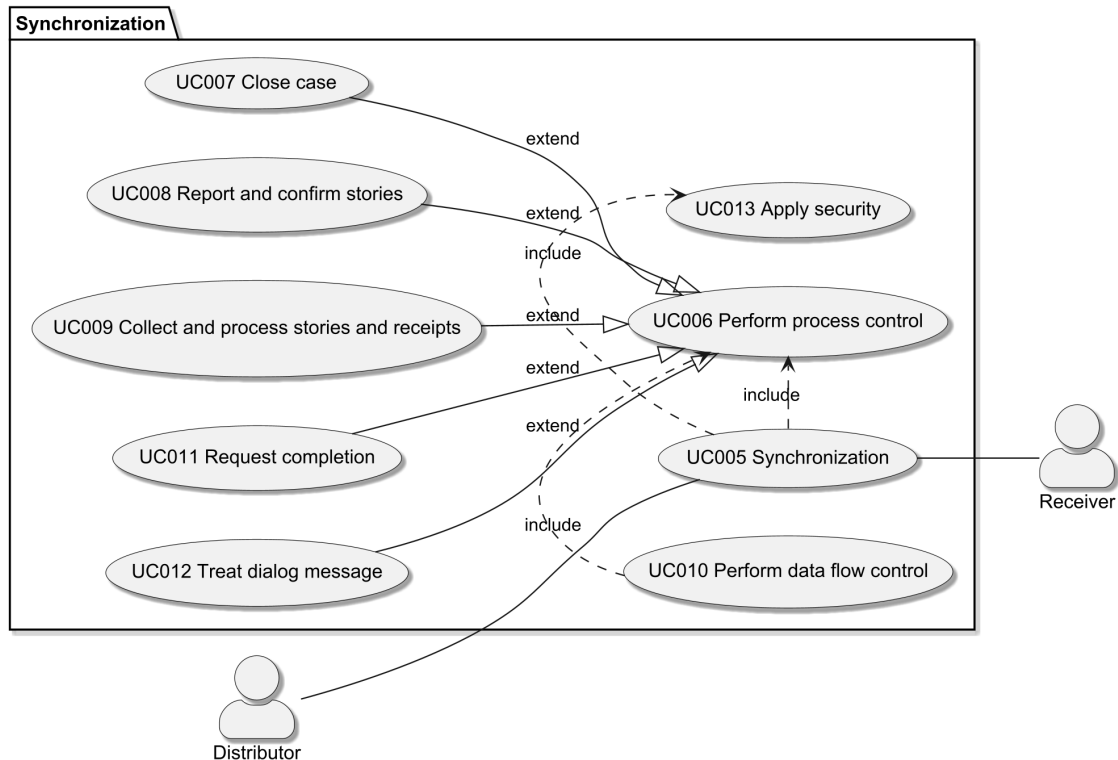


Abbildung 3.2. Use Cases Synchronisierung

Weitere Use Cases umfassen das vorgängige Konfigurieren eines Transmitters für die Übermittlung, d.h. das Einpflegen der aktuell gültigen Versicherungsprofile ( [Abschnitt 8.16, „UC016 Wartungsfenster“](#) ) und die Darstellung von Supportinformationen ( [Abschnitt 8.17, „UC017 Supportanfrage bearbeiten“](#) ) im Fehlerfall.

Des Weiteren gibt es einen Erreichbarkeitstest ( [Abschnitt 8.18, „UC018 Erreichbarkeit prüfen“](#) ), der vom Distributor regelmässig durchgeführt wird.

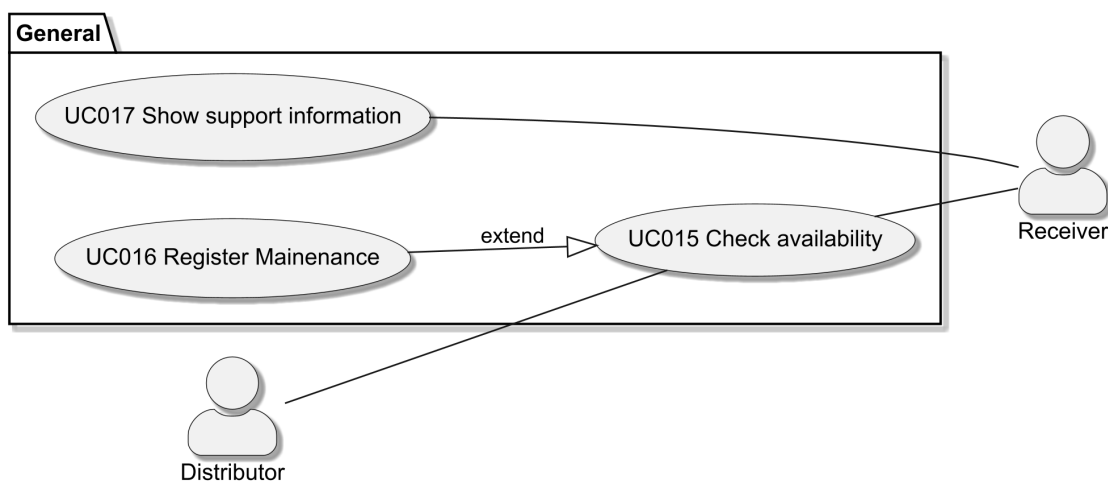


Abbildung 3.3. Andere Use Cases

### 3.3 Uses Cases und ihre Operationen

In der nachfolgenden Tabelle werden die Operationen des Leistungsstandards auf die generisch gehaltenen Use Cases abgebildet. Dies soll dem besseren Verständnis der Zusammenhänge zwischen den fachlichen Geschäftsprozessen und den technischen Strukturen und Abläufen dienen.

| Use Case                                  | Operation / Element   |
|---|---|
| UC001 Initiale Meldung senden             | <ul style="list-style-type: none"> <li>• DeclareIncidentConsumer: Operation zur Registrierung eines Ereignis (Unfall oder Krankheit)</li> <li>• DeclareIncidentConsumerResponse: Rückmeldung zur Registrierung eines Ereignis</li> <li>• IncidentDeclarationConsumerFault: Rückmeldung im Fehlerfall</li> </ul>   |
| UC005 Synchronisieren                     | <ul style="list-style-type: none"> <li>• SynchronizeDeclareIncidentConsumer: Aktualisierung des Ereignisses</li> <li>• SynchronizeDeclareIncidentConsumerResponse: Rückmeldung der Aktualisierungsanfrage</li> <li>• IncidentDeclarationConsumerFault: Rückmeldung im Fehlerfall</li> </ul>   |
| UC013 Anmeldeprozess                      | <ul style="list-style-type: none"> <li>• SubscribeOrganizationConsumer: Anmeldeprozess</li> <li>• SubscribeOrganizationConsumerResponse: Rückmeldung zum Anmeldeprozess</li> <li>• SynchronizeSubscribeOrganizationConsumer: Aktualisierung des Anmeldeprozesses</li> <li>• SynchronizeSubscribeOrganizationConsumerResponse: Rückmeldung der Aktualisierungsanfrage</li> <li>• IncidentDeclarationConsumerFault: Rückmeldung im Fehlerfall</li> </ul>  |
| UC014 Unternehmensauthentifizierung (SUA) | <ul style="list-style-type: none"> <li>• RegisterOrganizationAuthenticationConsumer: SUA-Unternehmensauthentifizierung</li> <li>• RegisterOrganizationAuthenticationConsumerResponse: Rückmeldung zur SUA-Unternehmensauthentifizierung</li> <li>• SynchronizeRegisterOrganizationAuthenticationConsumer: Aktualisierung des SUA-Prozesses</li> <li>• SynchronizeRegisterOrganizationAuthenticationConsumerResponse: Rückmeldung der Aktualisierungsanfrage</li> <li>• IncidentDeclarationConsumerFault: Rückmeldung im Fehlerfall</li> </ul> |
| UC015 Erreichbarkeit prüfen               | <ul style="list-style-type: none"> <li>• PingConsumer</li> <li>• PingConsumerResponse</li> </ul>  |

Tabelle 3.1. Use Cases und Operationen



## 4 Ping

Die Erreichbarkeit des Endempfängers muss geprüft werden. Dazu wird eine einfache Anfrage vom Distributor an den Endempfänger geschickt. Die Antwort des Endempfängers bestätigt die Erreichbarkeit.

Mit dem Ping-Aufruf wird die Systemzeit übermittelt, sodass es möglich ist, die Zeiten von Distributor und Endempfängers zu vergleichen. Damit lassen sich Timestamp-Probleme aufdecken. Dieser Use Case dient der Qualitätssicherung bei der Installation und der Entwicklung.

Die Anforderungen sind im [Abschnitt 8.15, „UC015 Security anwenden“](#) beschrieben.

Der Endempfänger kann in der Antwort auch auf allfällige Wartungsfenster in der Zukunft hinweisen.

Die Anforderungen sind im [Abschnitt 8.16, „UC016 Wartungsfenster“](#) beschrieben.

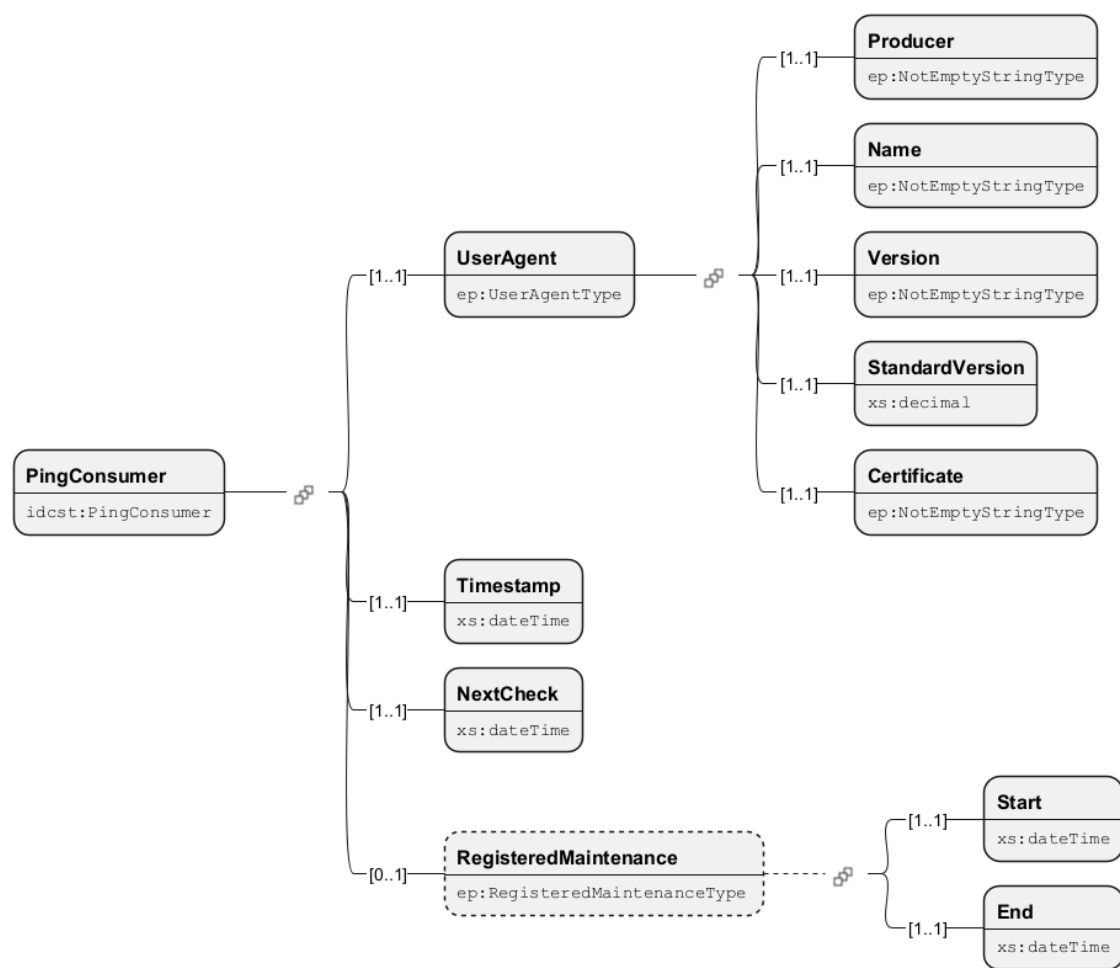


Abbildung 4.1. Schemabild PingConsumer

| Feldname  | Beschreibung   | Typ                              |
|-----------|--|----------------------------------|
| UserAgent | Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert | <a href="#">ep UserAgentType</a> |
| Timestamp | Ein Zeitstempel (englisch timestamp) wird benutzt, um einem Ereignis einen eindeutigen Zeitpunkt zuzuordnen.     | xs dateTime                      |

| Feldname              | Beschreibung | Typ  |
|-----------------------|--------------|--|
|                       |              |  |
| NextCheck             |              | xs dateTime                                  |
| RegisteredMaintenance |              | <a href="#">ep_RegisteredMaintenanceType</a> |

Tabelle 4.1. Felddescriptions PingConsumer

| Feldname        | Beschreibung   | Typ                                    |
|-----------------|--|--|
| Producer        | Hersteller der Applikation   | <a href="#">ep_NotEmptyString-Type</a> |
| Name            | Name des Produktes + eventuelle Zusatzinformationen (Kauftransmitter, etc.)  | <a href="#">ep_NotEmptyString-Type</a> |
| Version         | Produkt-Version  | <a href="#">ep_NotEmptyString-Type</a> |
| StandardVersion | Die Version des Standard-CH (z.B. ELM, KLE, SUA, usw.) nach der die Zertifizierung durchgeführt wurde.(z.B. 1.0) Achtung: Der Distributor kann zum Teil Versionen transformieren und wird dann automatisch eine tiefere Version setzen (z.B. 2.2). | xs decimal                             |
| Certificate     | Zertifikatsnummer, xxxx.yy wie auf dem physischen Zertifikat abgebildet  | <a href="#">ep_NotEmptyString-Type</a> |

Tabelle 4.2. Felddescriptions UserAgent

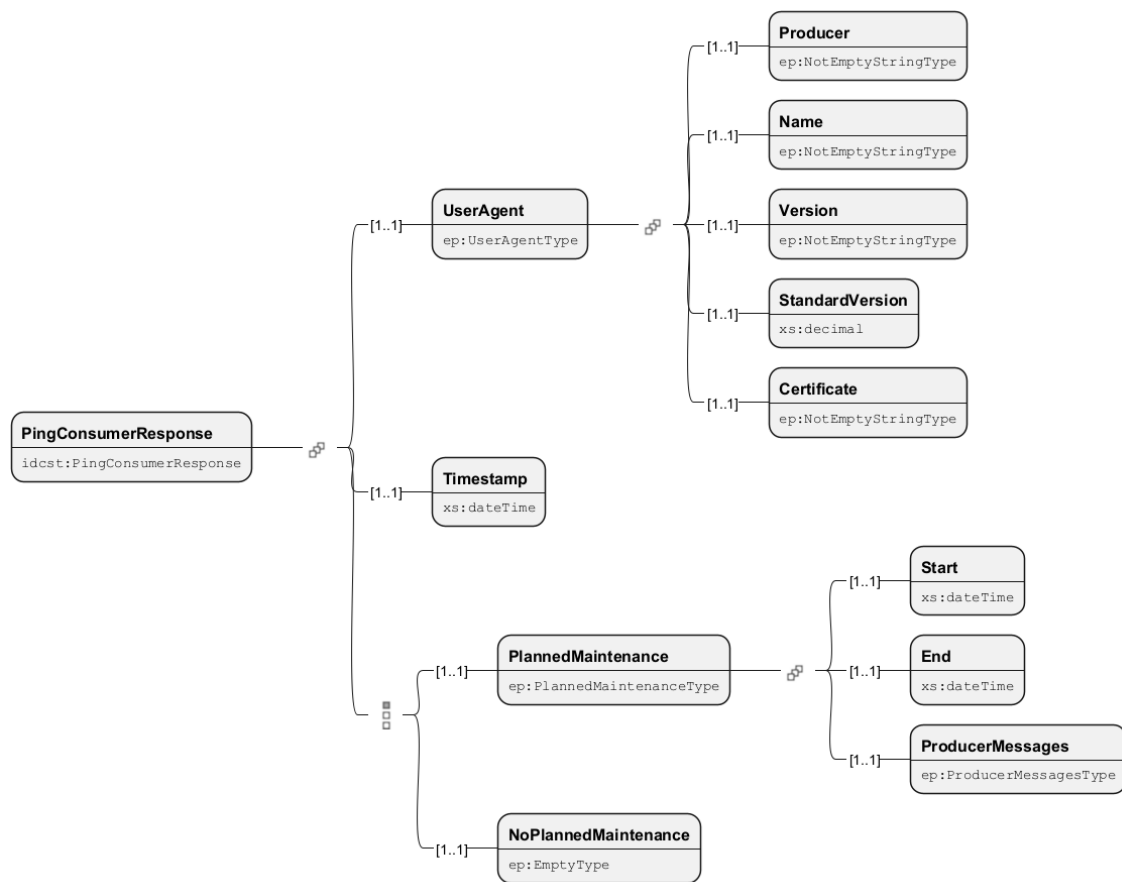


Abbildung 4.2. Schemabild PingConsumerResponse

| Feldname             | Beschreibung   | Typ                                       |
|----------------------|--|---|
| UserAgent            | Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert | <a href="#">ep UserAgentType</a>          |
| Timestamp            |  | xs dateTime                               |
| PlannedMaintenance   |  | <a href="#">ep PlannedMaintenanceType</a> |
| NoPlannedMaintenance |  | <a href="#">ep EmptyType</a>              |

Tabelle 4.3. Feldbeschreibungen PingConsumerResponse

| Feldname        | Beschreibung  | Typ                                   |
|-----------------|---|---------------------------------------|
| Producer        | Hersteller der Applikation  | <a href="#">ep NotEmptyStringType</a> |
| Name            | Name des Produktes + eventuelle Zusatzinformationen (Kauftransmitter, etc.)   | <a href="#">ep NotEmptyStringType</a> |
| Version         | Produkt-Version   | <a href="#">ep NotEmptyStringType</a> |
| StandardVersion | Die Version des Standard-CH (z.B. ELM, KLE, SUA, usw.) nach der die Zertifizierung durchgeführt wurde.(z.B. 1.0) Achtung: Der | xs decimal                            |

| Feldname    | Beschreibung   | Typ                                    |
|-------------|--|--|
|             | Distributor kann zum Teil Versionen transformieren und wird dann automatisch eine tiefere Version setzen (z.B. 2.2). |  |
| Certificate | Zertifikatsnummer, xxxx.yy wie auf dem physischen Zertifikat abgebildet  | <a href="#">ep_NotEmptyString-Type</a> |

Tabelle 4.4. Feldbeschreibungen UserAgent

## 5 RegisterOrganizationAuthentication

### 5.1 Übersicht

RegisterOrganizationAuthentication ist der unterstützende Prozess für die Swissdec Unternehmens-Authentifizierung SUA. Die Verwendung der vorliegenden Operationen wird im [Anhang C, Detailspezifikation Swissdec Unternehmens-Authentifizierung SUA](#) beschrieben.

Der Endreceiver unterstützt den SUA Prozess im Rahmen der Registrierung zur Überprüfung der Identität des Unternehmens basierend auf der bestehenden Vertragsbeziehung. Der Endreceiver prüft die übermittelte Vertragsbeziehung und meldet dem Distributor, ob die Vertragsbeziehung gültig ist. In der Antwort werden die Stammdaten des Unternehmens an den Distributor gemeldet. Zu diesen Informationen gehören der Name des Unternehmens, die UID, die hinterlegte Unternehmensadresse, die Kontaktperson sowie allfällige Stellvertreter. Diese Informationen sind essenziell für die Sicherheit des SUA Prozesses und werden vom Distributor benötigt, um die Identität des Unternehmens zu überprüfen.

Nach der erfolgreichen Registrierung wird der SUA Prozess hauptsächlich vom Distributor gehandhabt. Die SUA Zertifikate werden auf dem Distributor erstellt und signiert. Der Distributor behält sich aber vor, bei einer Erneuerung eines SUA Zertifikats die Vertragsverbindung wieder zu prüfen. Deshalb bleibt die Registrierung längerfristig als offener Fall bestehen.

### 5.2 Initiale synchrone Meldung

Die Aktivitätsdiagramme sind ohne den Transmitter dargestellt. Die wesentliche Receiverfunktionalität soll dargestellt werden. Die Anzeigen erfolgen wahlweise in Receiver oder im Empfängersystem.

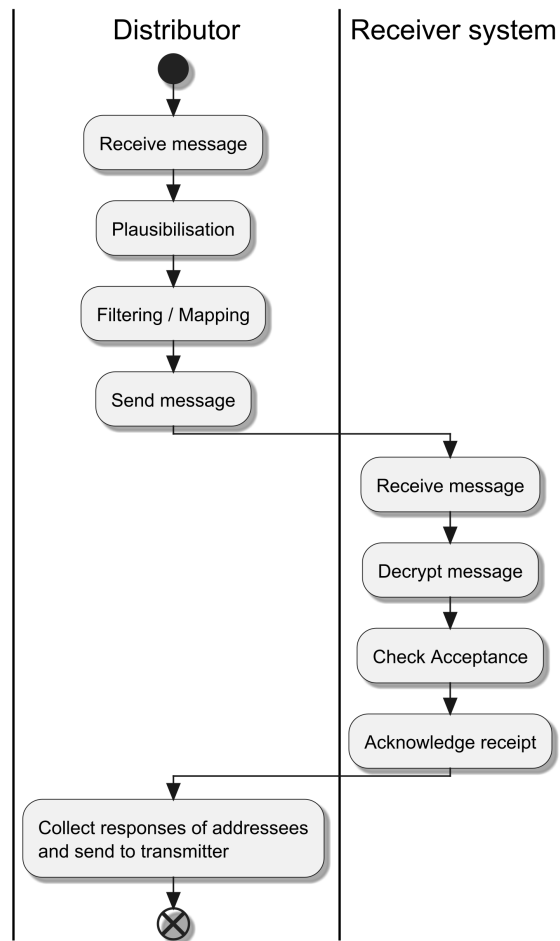


Abbildung 5.1. Synchroner Meldungsprozess

## 5.3 Datenstruktur der initialen Meldung

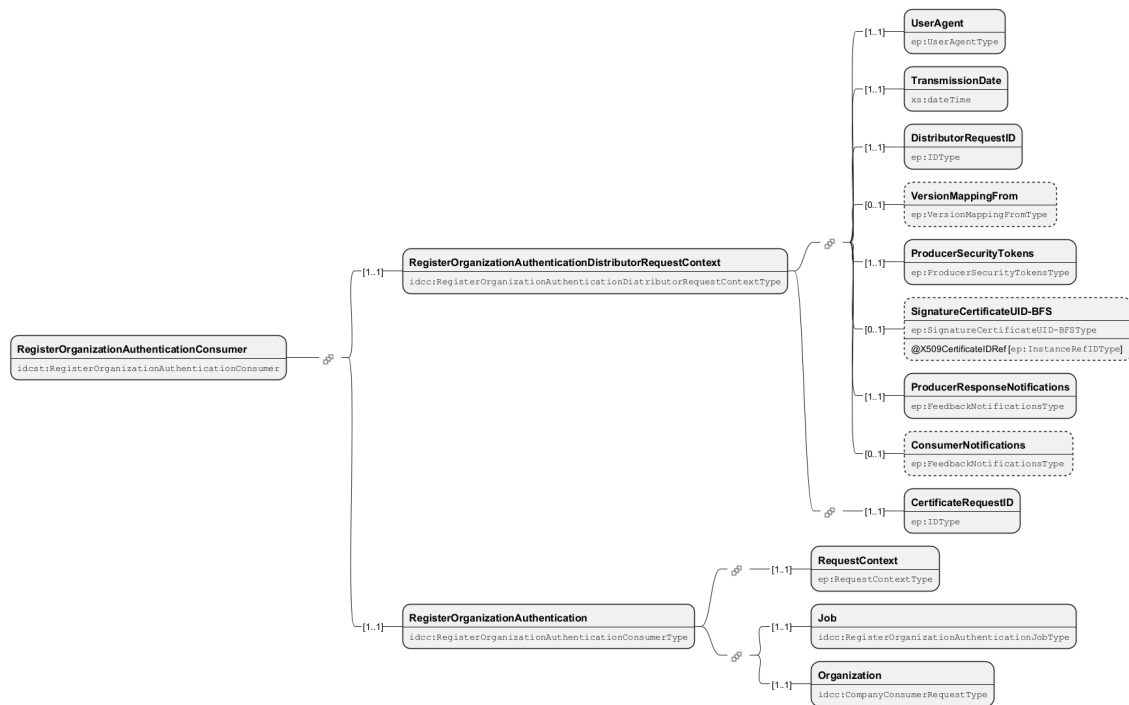


Abbildung 5.2. Schemabild RegisterOrganizationAuthenticationConsumer

| Feldname  | Beschreibung | Typ  |
|---|--------------|--|
| RegisterOrganizationAuthenticationDistributorRequestContext |              | <a href="#">idcc_RegisterOrganizationAuthenticationDistributorRequestContextType</a> |
| RegisterOrganizationAuthentication                          |              | <a href="#">idcc_RegisterOrganizationAuthenticationConsumerType</a>                  |

Tabelle 5.1. Feldbeschreibungen RegisterOrganizationAuthenticationConsumer

| Feldname         | Beschreibung  | Typ                                   |
|------------------|---|---------------------------------------|
| UserAgent        | Beschreibung der wesentlichen System-Identifikation-Daten   | <a href="#">ep_UserAgentType</a>      |
| CompanyName      | Beschreibung der wesentlichen Unternehmensdaten   | <a href="#">ep_NotEmptyStringType</a> |
| TransmissionDate | Übertragungszeitpunkt   | xs dateTime                           |
| RequestID        | AuftragsID  | <a href="#">ep_IDType</a>             |
| LanguageCode     | Sprachcode  | <a href="#">ep_LanguageCodeType</a>   |
| MonitoringID     | Die MonitoringID erlaubt es, auf den Testsystemen Übermittlungen den Benutzern zuzuweisen. Sie ist in der Produktion überflüssig. | <a href="#">ep_MonitoringIDType</a>   |

| Feldname | Beschreibung | Typ |
|----------|--------------|-----|
|          |              |     |

Tabelle 5.2. Feldbeschreibungen RequestContext

| Feldname  | Beschreibung                  | Typ                                 |
|-----------|-------------------------------|-------------------------------------|
| Addressee | Adressierung der Endempfänger | <a href="#">ep AddresseeJobType</a> |
| TestCase  |                               | <a href="#">ep EmptyType</a>        |

Tabelle 5.3. Feldbeschreibungen Job

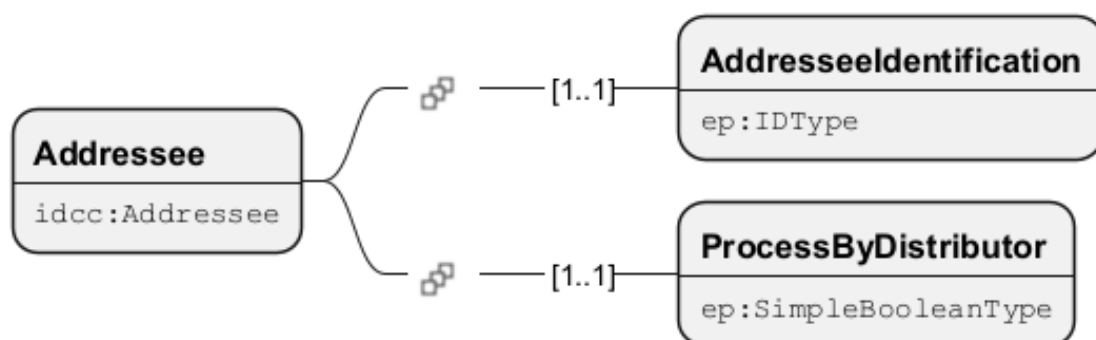


Abbildung 5.3. Schemabild Addressee

| Feldname                | Beschreibung                  | Typ                                  |
|-------------------------|-------------------------------|--------------------------------------|
| AddresseeIdentification | Identifikation des Adressaten | <a href="#">ep IDType</a>            |
| ProcessByDistributor    | Verteilung durch Distributor  | <a href="#">ep SimpleBooleanType</a> |
| @addresseeID            | Referenz auf Adressat         | <a href="#">ep InstanceRefIDType</a> |

Tabelle 5.4. Feldbeschreibungen Addressee

## 5.4 Datenstruktur der initialen Antwort



Abbildung 5.4. Schemabild RegisterOrganizationAuthenticationConsumerResponse

| Feldname                                   | Beschreibung | Typ  |
|--|--------------|--|
| RegisterOrganizationAuthenticationResponse |              | <a href="#">c RegisterOrganizationAuthenticationAddresseeSuccessJobStateType</a> |

Tabelle 5.5. Feldbeschreibungen RegisterOrganizationAuthenticationConsumerResponse



## 5.5 Datenstruktur der Synchronize-Meldung

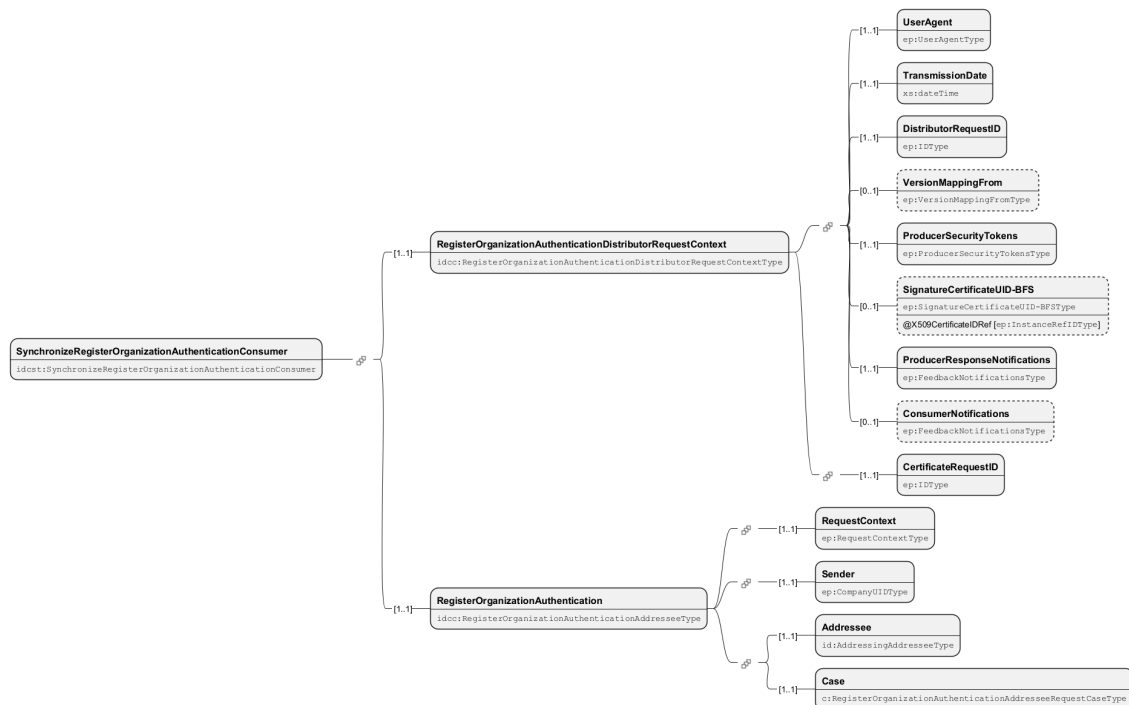


Abbildung 5.5. Schemabild SynchronizeRegisterOrganizationAuthenticationConsumer

| Feldname  | Beschreibung | Typ  |
|---|--------------|--|
| RegisterOrganizationAuthenticationDistributorRequestContext |              | <a href="#">idcc_RegisterOrganizationAuthenticationDistributorRequestContextType</a> |
| RegisterOrganizationAuthentication                          |              | <a href="#">idcc_RegisterOrganizationAuthenticationAddresseeType</a>                 |

Tabelle 5.6. Feldbeschreibungen SynchronizeRegisterOrganizationAuthenticationConsumer

## 5.6 Datenstruktur der Synchronize-Antwort



Abbildung 5.6. Schemabild SynchronizeRegisterOrganizationAuthenticationConsumerResponse

| Feldname                                      | Beschreibung | Typ  |
|---|--------------|--|
| SynchronizeRegisterOrganizationAuthentication |              | <a href="#">c_SynchronizeRegisterOrganizationAuthenticationAddresseeResponse</a> |

Tabelle 5.7. Feldbeschreibungen SynchronizeRegisterOrganizationAuthenticationConsumerResponse

## 6 SubscribeOrganization

### 6.1 Übersicht

Ein Sendersystem muss sich bei einem Empfängersystem für einen späteren Übermittlungsprozess (z.B. EO) anmelden können. Mit diesem Anmeldeprozess wird der zukünftige Informationsaustausch über den Distributor zwischen den Beteiligten in die Wege geleitet. Der Anmeldeprozess dient dazu, Informationen auf Vertragsebene auszutauschen und keine fachlichen Inhalte. Diese werden je nach Swissdec Standard dann erst nach erfolgreich abgeschlossenem Anmeldeprozess mittels anderer Operationen übermittelt.

Der Anmeldeprozess besteht aus drei Schritten:

- **SubscribeOrganization:** In einem ersten Schritt übermittelt das Unternehmen seine Informationen an den Endempfänger, um sich für den späteren Informationsaustausch anzumelden. Als Antwort darauf erhält der Absender eine SubscriptionID, die in den weiteren Schritten zur Identifikation beim Endempfänger genutzt wird.
- **SynchronizeSubscribeOrganization:** In allen weiteren Schritten können nun mittels SubscriptionID Informationen beim Endempfänger abgeholt werden.
- **SynchronizeSubscribeOrganization:** Es *sollten* etwa 1x wöchentlich weitere Synchronisierungen durchgeführt werden. So wird das Sendersystem zeitnah über Änderungen seitens Endempfänger informiert.

### 6.2 Initiale synchrone Meldung

Die Aktivitätsdiagramme sind ohne den Transmitter dargestellt. Die wesentliche Receiverfunktionalität *soll* dargestellt werden. Die Anzeigen erfolgen wahlweise in Receiver oder im Empfängersystem.

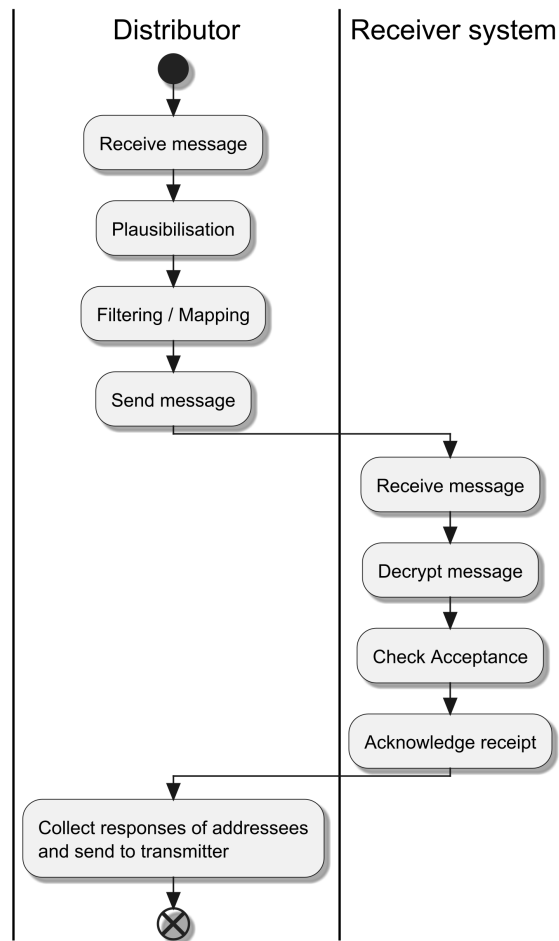


Abbildung 6.1. Synchroner Meldungsprozess

## 6.3 Datenstruktur der initialen Meldung

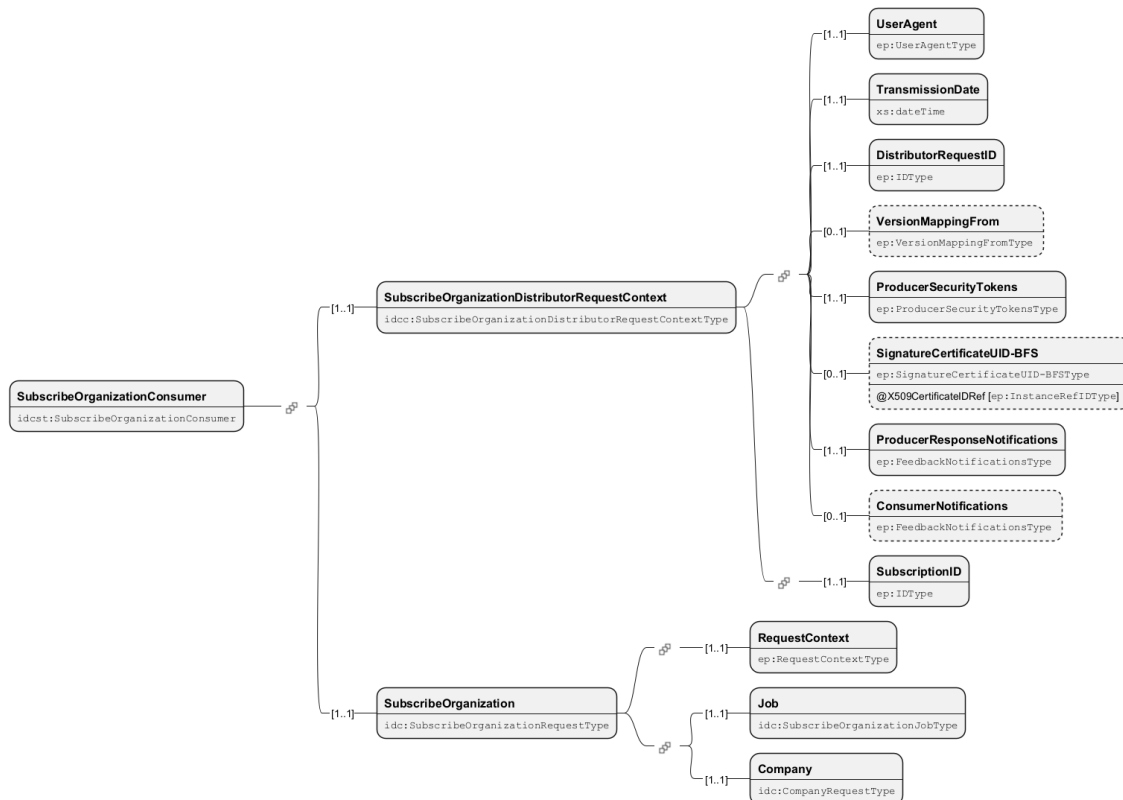


Abbildung 6.2. Schemabild SubscribeOrganizationConsumer

| Feldname                                       | Beschreibung | Typ   |
|--|--------------|---|
| SubscribeOrganizationDistributorRequestContext |              | <a href="#">idcc SubscribeOrganizationDistributorRequestContextType</a> |
| SubscribeOrganization                          |              | <a href="#">idc SubscribeOrganizationRequestType</a>                    |

Tabelle 6.1. Felddescriptions SubscribeOrganizationConsumer

| Feldname         | Beschreibung  | Typ                                   |
|------------------|---|---------------------------------------|
| UserAgent        | Beschreibung der wesentlichen System-Identifikation-Daten | <a href="#">ep UserAgentType</a>      |
| CompanyName      | Beschreibung der wesentlichen Unternehmensdaten           | <a href="#">ep NotEmptyStringType</a> |
| TransmissionDate | Übertragungszeitpunkt                                     | xs dateTime                           |
| RequestID        | AuftragsID  | <a href="#">ep IDType</a>             |
| LanguageCode     | Sprachcode  | <a href="#">ep LanguageCodeType</a>   |

| Feldname     | Beschreibung  | Typ                                 |
|--------------|---|-------------------------------------|
| MonitoringID | Die MonitoringID erlaubt es, auf den Testsystemen Übermittlungen den Benutzern zuzuweisen. Sie ist in der Produktion überflüssig. | <a href="#">ep_MonitoringIDType</a> |

Tabelle 6.2. Feldbeschreibungen RequestContext

| Feldname  | Beschreibung                  | Typ                                 |
|-----------|-------------------------------|-------------------------------------|
| Addressee | Adressierung der Endempfänger | <a href="#">ep_AddresseeJobType</a> |

Tabelle 6.3. Feldbeschreibungen Job

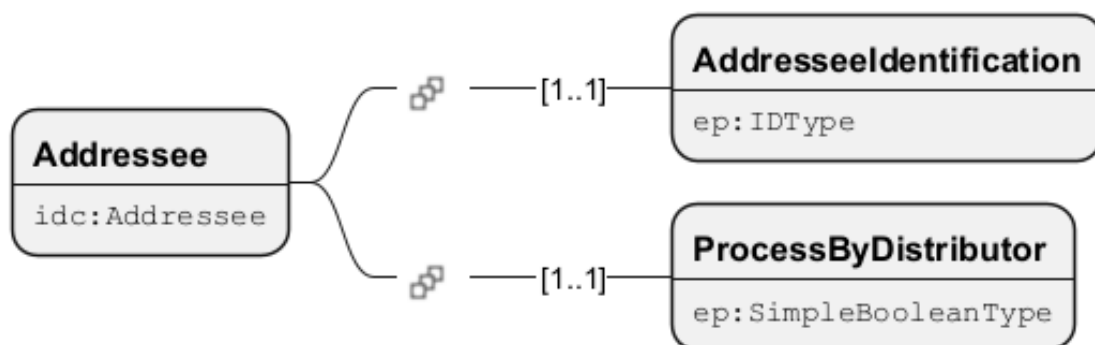


Abbildung 6.3. Schemabild Addressee

| Feldname               | Beschreibung                  | Typ                                  |
|------------------------|-------------------------------|--------------------------------------|
| AdresseeIdentification | Identifikation des Adressaten | <a href="#">ep_IDType</a>            |
| ProcessByDistributor   | Verteilung durch Distributor  | <a href="#">ep_SimpleBooleanType</a> |
| @adresseeID            | Referenz auf Adressat         | <a href="#">ep_InstanceRefIDType</a> |

Tabelle 6.4. Feldbeschreibungen Addressee

## 6.4 Datenstruktur der initialen Antwort



Abbildung 6.4. Schemabild SubscribeOrganizationConsumerResponse

| Feldname                      | Beschreibung                | Typ   |
|-------------------------------|-----------------------------|---|
| SubscribeOrganizationResponse | Initialer Status des Falls. | <a href="#">idc_SubscribeOrganizationAddresseeSuccessJobStateType</a> |

Tabelle 6.5. Feldbeschreibungen SubscribeOrganizationConsumerResponse

## 6.5 Datenstruktur der Synchronize-Meldung

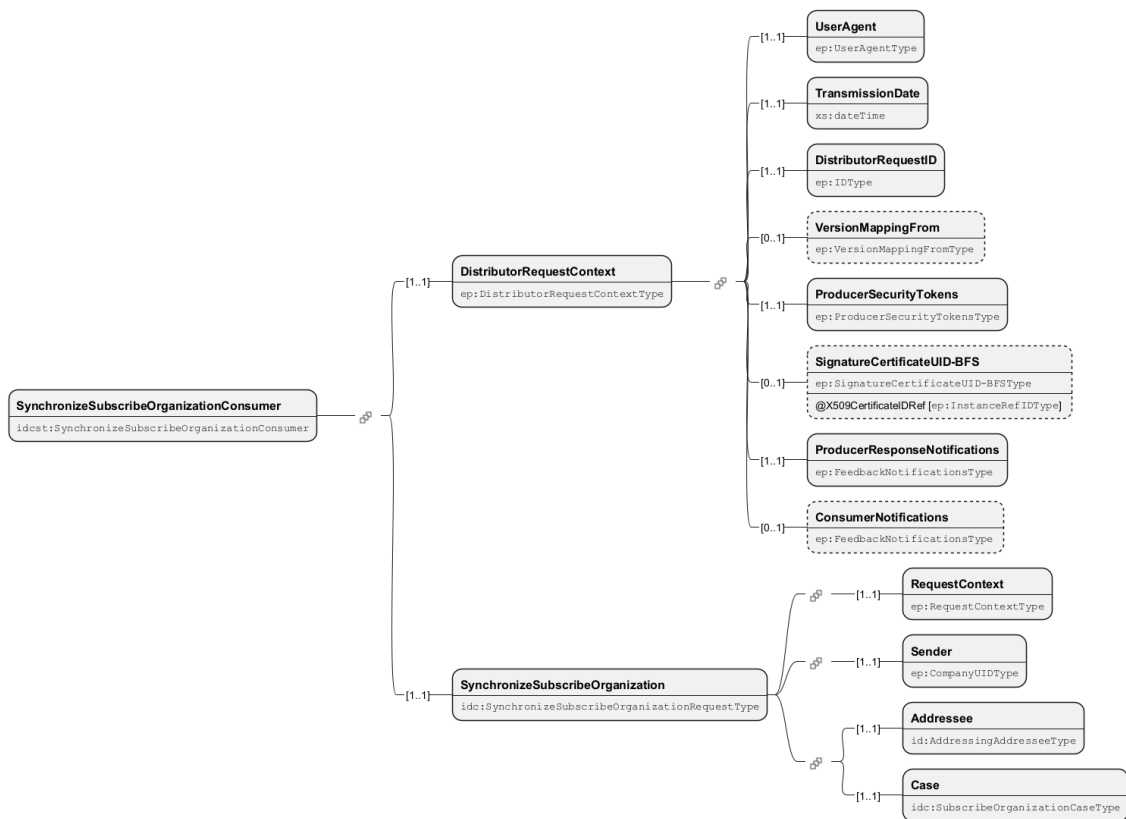


Abbildung 6.5. Schemabild SynchronizeSubscribeOrganizationConsumer

| Feldname                         | Beschreibung | Typ   |
|----------------------------------|--------------|---|
| DistributorRequest-Context       |              | <a href="#">ep DistributorRequestContextType</a>                |
| SynchronizeSubscribeOrganization |              | <a href="#">idc SynchronizeSubscribeOrganizationRequestType</a> |

Tabelle 6.6. Felddescriptions SynchronizeSubscribeOrganizationConsumer

## 6.6 Datenstruktur der Synchronize-Antwort



Abbildung 6.6. Schemabild SynchronizeSubscribeOrganizationConsumerResponse

| Feldname   | Beschreibung | Typ  |
|--|--------------|--|
| SynchronizeSub-<br>scribeOrganization-<br>Response |              | <a href="#">idc SynchronizeSub-<br/>scribeOrganization-<br/>ConsumerType</a> |

Tabelle 6.7. Feldbeschreibungen SynchronizeSubscribeOrganizationConsumerResponse

## 7 DeclareIncident

### 7.1 KLE Ereignismeldung

Die [Richtlinien für Lohndatenverarbeitung](#) beschreiben, welche Daten zu welchem Zeitpunkt übermittelt werden. Dieses Dokument beschränkt sich darauf, die technischen Informationen zur Übermittlung zu liefern.

Die KLE Ereignismeldung findet in mehreren Etappen statt, welche nachfolgend beschrieben werden.

#### 7.1.1 Ereignis melden: DeclareIncidentConsumer

In diesem ersten Schritt wird dem Endempfänger in einer synchronen Übermittlung ein Ereignis gemeldet. Die Ereignismeldung geht vom Transmitter an den Distributor und wird von diesem sofort weitergeleitet zum Endreceiver. Der Endreceiver überprüft die Meldung auf Akzeptanzstufe (z.B. Vertragsnummer) und sendet via Distributor zurück, ob er das Ereignis annimmt. Im Erfolgsfall kann nun vom Sendersystem mit dem nächsten Schritt, der Synchronisierung, fortgefahren werden.

#### 7.1.2 Synchronisierung des Ereignisses: SynchronizeDeclareIncident

Sender- und Empfängersystem kommunizieren nun mittels Stories, in denen zusätzliche Informationen zum Ereignis, sowie Zahlungsinformationen vom Endempfänger übermittelt werden können. Der Endempfänger kann dem Sendersystem jeweils mitteilen, wenn weitere Stories zur Bearbeitung des Falls benötigt werden und auch darüber informieren, wenn Informationen seitens Empfänger zum Abholen bereitstehen. So kann der Fall in einer flexiblen Anzahl Synchronisierungen abgewickelt werden.

#### 7.1.3 Regelmässiges Synchronisieren der Ereignisse

Auch wenn kein konkreter Fall in Arbeit ist, führt das Sendersystem regelmässige Synchronisierungen über ein ganzes Unternehmen durch. So kann der Endempfänger informieren, wenn für ein aktuell inaktives Ereignis neue Informationen zur Verfügung stehen.

### 7.2 Initiale synchrone Meldung

Die Aktivitätsdiagramme sind ohne den Transmitter dargestellt. Die wesentliche Receiverfunktionalität *soll* dargestellt werden. Die Anzeigen erfolgen wahlweise in Receiver oder im Empfängersystem.



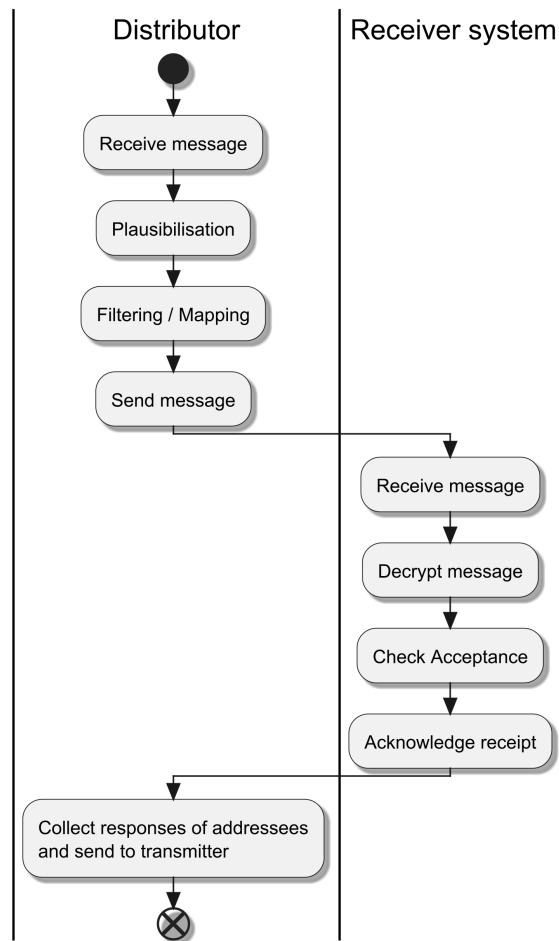


Abbildung 7.1. Synchroner Meldungsprozess

### 7.3 Ablauf / Protokoll

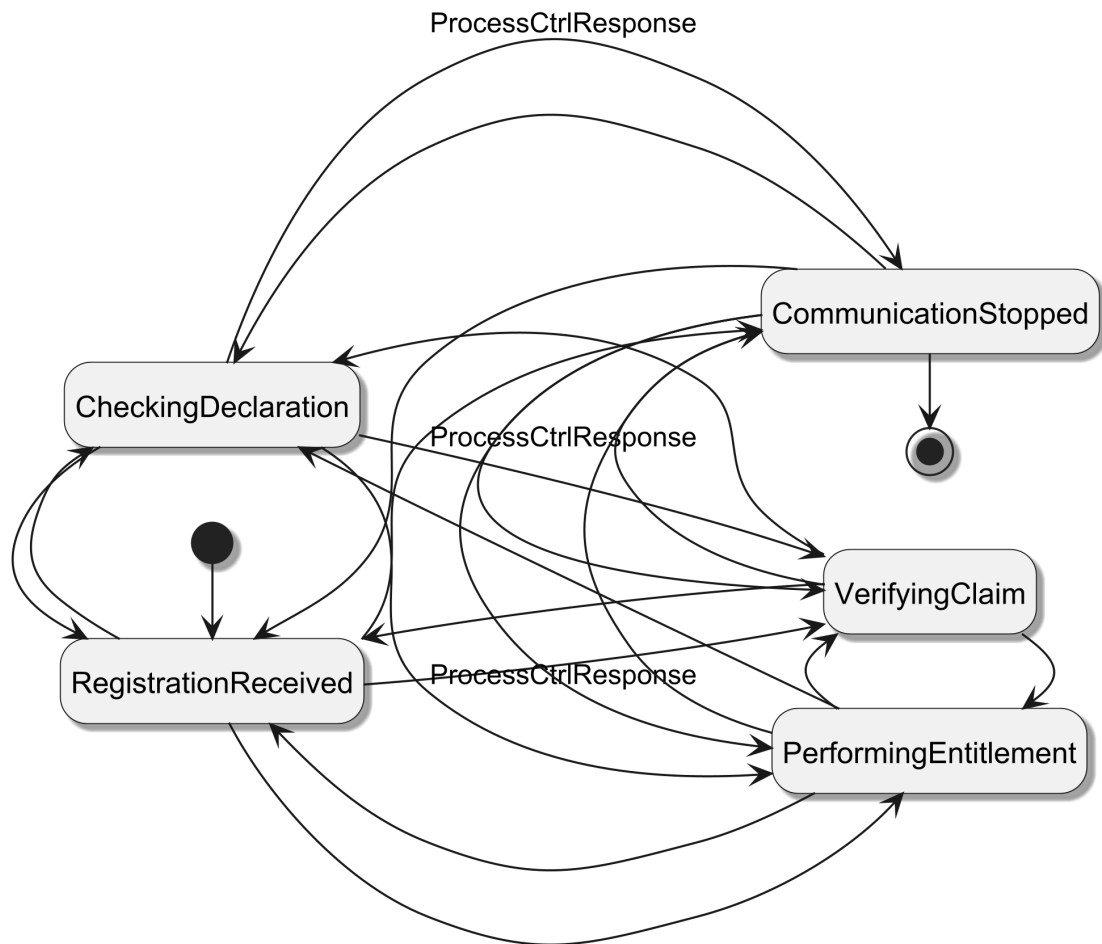


Abbildung 7.2. DeclareIncident Protokollstatus

7.4 Datenstruktur der initialen Meldung

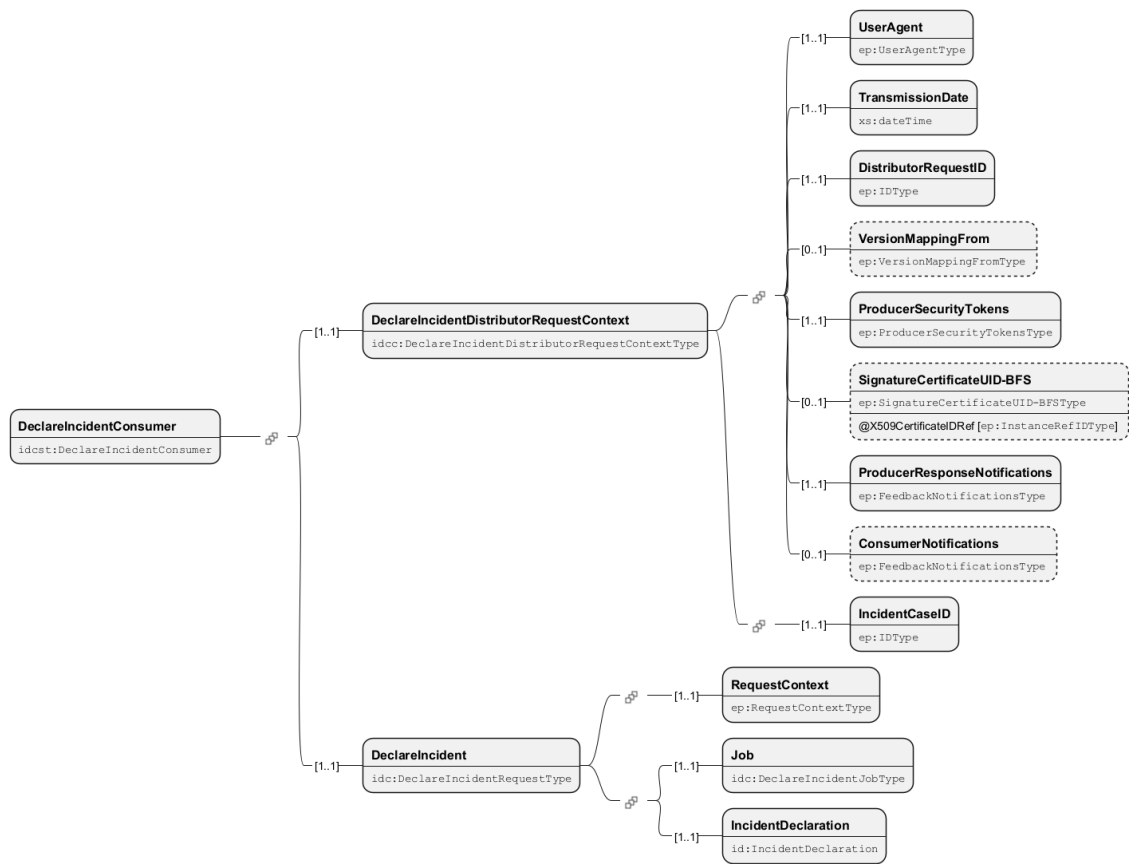


Abbildung 7.3. Schemabild DeclareIncidentConsumer

| Feldname                                   | Beschreibung | Typ   |
|--|--------------|---|
| DeclareIncidentDis-tributorRequestCon-text |              | <a href="#">idcc DeclareIncidentDistributorRequestContextType</a> |
| DeclareIncident                            |              | <a href="#">idc DeclareIncidentRequestType</a>                    |

Tabelle 7.1. Felddescriptions DeclareIncidentConsumer

| Feldname         | Beschreibung  | Typ                                    |
|------------------|---|--|
| UserAgent        | Beschreibung der wesentlichen System-Identifikation-Daten | <a href="#">ep UserAgentType</a>       |
| CompanyName      | Beschreibung der wesentlichen Unternehmensdaten           | <a href="#">ep NotEmptyString-Type</a> |
| TransmissionDate | Übertragungszeitpunkt                                     | xs dateTime                            |
| RequestID        | AuftragsID  | <a href="#">ep IDType</a>              |
| LanguageCode     | Sprachcode  | <a href="#">ep LanguageCode-Type</a>   |

| Feldname     | Beschreibung  | Typ                                 |
|--------------|---|-------------------------------------|
| MonitoringID | Die MonitoringID erlaubt es, auf den Testsystemen Übermittlungen den Benutzern zuzuweisen. Sie ist in der Produktion überflüssig. | <a href="#">ep_MonitoringIDType</a> |

Tabelle 7.2. Feldbeschreibungen RequestContext

| Feldname   | Beschreibung | Typ  |
|------------|--------------|--|
| Addressees |              | <a href="#">idc_DeclareIncident-AddresseesType</a> |
| TestCase   |              | <a href="#">ep_EmptyType</a>                       |

Tabelle 7.3. Feldbeschreibungen Job



Abbildung 7.4. Schemabild Addressees

| Feldname  | Beschreibung                  | Typ                                 |
|-----------|-------------------------------|-------------------------------------|
| Addressee | Adressierung der Endempfänger | <a href="#">ep_AddresseeJobType</a> |

Tabelle 7.4. Feldbeschreibungen Addressees

## 7.5 Datenstruktur der initialen Antwort



Abbildung 7.5. Schemabild DeclareIncidentConsumerResponse

| Feldname                | Beschreibung                | Typ   |
|-------------------------|-----------------------------|---|
| DeclareIncidentResponse | Initialer Status des Falls. | <a href="#">idc_DeclareIncident-AddresseeSuccess-JobStateType</a> |

Tabelle 7.5. Feldbeschreibungen DeclareIncidentConsumerResponse

## 7.6 Datenstruktur der Synchronize-Meldung

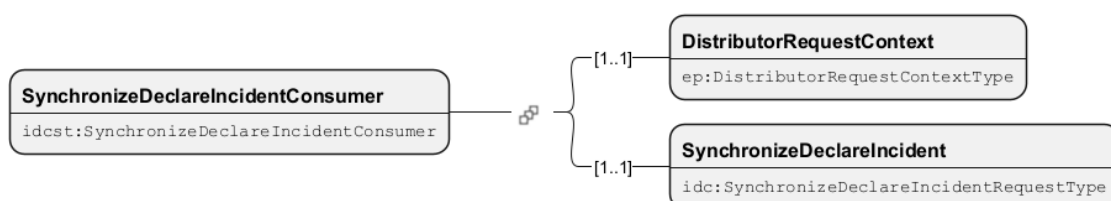


Abbildung 7.6. Schemabild SynchronizeDeclareIncidentConsumer

| Feldname                   | Beschreibung | Typ   |
|----------------------------|--------------|---|
| DistributorRequest-Context |              | <a href="#">ep_DistributorRequestContextType</a>          |
| SynchronizeDeclareIncident |              | <a href="#">idc_SynchronizeDeclareIncidentRequestType</a> |

Tabelle 7.6. Feldbeschreibungen SynchronizeDeclareIncidentConsumer

## 7.7 Datenstruktur der Synchronize-Antwort



Abbildung 7.7. Schemabild SynchronizeDeclareIncidentConsumerResponse

| Feldname                           | Beschreibung | Typ  |
|------------------------------------|--------------|--|
| SynchronizeDeclareIncidentResponse |              | <a href="#">idc_SynchronizeDeclareIncidentConsumerType</a> |

Tabelle 7.7. Feldbeschreibungen SynchronizeDeclareIncidentConsumerResponse

## 8 Use Cases

### 8.1 UC001 Initiale Meldung empfangen

Use Case Diagramm: siehe [Abbildung 3.1, „Use Cases Initiale Meldung“](#).

|                     |   |
|---------------------|---|
| Kurzbeschreibung    | Der Distributor schickt gemäss Job die gefilterte Meldung des Unternehmens an den Endreceiver. Dieser prüft, quittiert und übergibt die Daten zur Freigabe durch den Absender (Buchhalter des Unternehmens)   |
| Akteure             | Sendersystem, Distributor, Endreceiver  |
| Auslöser            | Der Distributor hat eine Meldung vom Sendersystem erhalten und gemäss Adressierung und Job die gefilterten Daten an den Endreceiver transferiert.   |
| Vorbedingungen      | Der Distributor hat die Meldung validiert und plausibilisiert (gem. Swissdec-Standard / <a href="#">[[ACKNOTIF]]</a> ).<br><br>Der Endreceiver ist bereit zum Empfangen einer Meldung.  |
| Nachbedingungen     | Der Endreceiver hat die Daten für den <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a> bereitgestellt und die Antwort mit den Credentials an den Distributor zurückgeliefert.   |
| Included Use Cases  | <a href="#">Abschnitt 8.2, „UC002 Duplikate behandeln“</a> , <a href="#">Abschnitt 8.15, „UC015 Security anwenden“</a>  |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Die vom Distributor gefilterten Meldungsdaten werden empfangen. Eine Validierung der Daten <i>kann</i> durchgeführt werden.</li> <li>2. Die zusätzlichen Distributor-Transportdaten müssen gemäss Distributorkopplung geprüft werden. (Achtung: Dublettendetektierung<sup>a</sup>, ...). Anomalien in der Meldung können geprüft werden.</li> <li>3. Die Daten werden gemäss Swissdec-Standard geprüft (Security und Akzeptanz).</li> <li>4. Sichern der Meldung. Alle personenbezogenen Daten werden gemäss Datenschutz gesichert.</li> <li>5. Credentials mit Key und Password bilden.</li> <li>6. Die Rückantwort zum Distributor bzw. Unternehmen wird gemäss Swissdecstandard (inklusive der vom Distributor bereits erstellten Nachrichten <code>&lt;ProducerResponseNotifications&gt;</code> <a href="#">[[ACKNOTIF]]</a>) aufgebaut.</li> </ol> |
| Alternative Abläufe | <p>{Schritt 2 und 3 Reihenfolge}</p> <p>Schritt 3 darf auch vor Schritt 2 durchgeführt werden.</p> <p>{Schritt 3: Die Stammdaten des Kunden sind nicht verfügbar}</p> <p>Die Meldung kann ungeprüft mit „Akzeptiert“ quittiert werden und eine Prüfung gegen die Stammdaten erfolgt später (manuell oder automatisch). Damit lässt sich eine hohe Verfügbarkeit erzielen.</p> <p>{Schritt 3: Die Meldung wurde vom Sendersystem mittels SUA Zertifikat signiert}</p> <p>Der Distributor liefert die SUA Zertifikatsinformationen im <code>*Distributor-RequestContext/SignatureCertificateUID-BFS</code> mit.</p> <p>{Schritt 5: Dublette wurde erkannt}</p> <p>Vorgehen gemäss <a href="#">Abschnitt 8.2, „UC002 Duplikate behandeln“</a></p> <p>{Schritt 1: geplanter Unterbruch / Dienst ist nicht verfügbar}</p>  |

|             |   |
|-------------|---|
|             | <p>Die Information zum Unterbruch (von-bis) wurde bereits mittels <a href="#">Abschnitt 8.18, „UC018 Erreichbarkeit prüfen“</a> an den Distributor übermittelt. In dieser Zeit werden dann Response-Meldungen (mit dieser Unterbruchinformation) vom Distributor direkt an die anfragende Lohnbuchhaltung zurückgegeben.</p> <p>{Schritt 1: ungeplanter Unterbruch / Dienst ist nicht verfügbar}</p> <p>In dieser Zeit werden dann Response-Meldungen (unbestimmte Dauer) vom Distributor direkt an die anfragende Lohnbuchhaltung zurückgegeben.</p> |
| Fehlerliste | <p>Fachliche Fehler:</p> <ul style="list-style-type: none"> <li>• Die Meldung verstösst gegen die Akzeptierungsregeln gemäss Swissdec-standard (z.B. Kunde ist bei der Institution unbekannt)</li> </ul> <p>Technische Fehler:</p> <ul style="list-style-type: none"> <li>• Meldung ist nicht schemavalid</li> <li>• Meldung kann nicht entschlüsselt werden</li> <li>• Meldung ist zu gross</li> <li>• usw.</li> </ul>   |

<sup>a</sup>Auch bei Dubletten werden die gesamten Daten übermittelt

Tabelle 8.1. UC001 Initiale Meldung empfangen

### Bemerkungen

Wie ist das Verhalten beim Senden von mehreren **unterschiedlichen** Meldungen eines Unternehmens, **bevor** eine dieser Meldungen abgeschlossen wird ( [Abschnitt 8.5, „UC005 Synchronisieren“](#) )?

- **Jede Übermittlung wird eigenständig behandelt.** Der Benutzer (Unternehmen) entscheidet, welche Meldung er abschliesst oder löscht.

Dabei können Anomalien nicht verhindert werden.

Nach einer ersten Meldung könnte zum Beispiel eine Substitutionsmeldung vorliegen, die vom Benutzer abgeschlossen wird. Zusätzlich wird noch die erste Meldung gelöscht, die substituiert werden sollte, d.h. es existiert eine Substitution ohne einen Vorgänger.

Funktion: „Meldungsleichen“ <sup>1</sup> werden nach Ablauf einer bestimmten Zeit gelöscht.

Probleme bzw. Verwirrung entstehen bei einer Freigabe-Mahnung oder einer späteren automatischen Konsumation von diesen „Meldungsleichen“. Vorsicht: Gemäss Swissdecschema kann der Prozess vom Endreceiver gesteuert werden.

Die Completionapplikation (s. [Abschnitt 8.11, „UC011 Completion“](#) ) *sollte* dem Benutzer einen Hinweis über die anstehenden und freigegebenen Meldungen liefern. Damit werden überflüssige Supportanfragen vermieden.

Nach einer Meldungsfreigabe mittels Completion **muss** eine weitere Meldungsübermittlung immer zulässig sein (z.B. Substitutionsmeldung). Optional könnte in der Quittung dazu eine Bemerkung stehen (Info oder Warning).

Im Moment wird auf der Transportebene mit möglichst wenigen Einschränkungen gearbeitet. Evtl. könnte dies später fachliche Probleme erzeugen (z.B. Kunde schickt Daten aus zwei Systemen (Angestellte und Geschäftsleitung) <sup>2</sup> ).

<sup>1</sup>Eine übermittelte Meldung (m2m; Receiver), die nie freigegeben wurde (h2m; Completion)

<sup>2</sup>Achtung: Teillieferungen sind im Standard nicht explizit definiert, d.h. eine Lösung muss mittels separater Vertragsidentifikation erfolgen.

## 8.2 UC002 Duplikate behandeln

|                     |  |
|---------------------|--|
| Kurzbeschreibung    | Die Dubletten werden mittels einer Hashwert-Berechnung bzw. eines Vergleiches bereits auf dem Distributor erkannt. Sinnvollerweise werden nichtfachliche, zeitbezogene Daten bei der Hashwertberechnung nicht berücksichtigt.  |
| Akteure             | Distributor, Endreceiver   |
| Auslöser            | Eine Meldung wird empfangen, die bereits zuvor übermittelt wurde.  |
| Vorbedingungen      | Eine identische Meldung wurde bereits zuvor an den Endreceiver übermittelt.  |
| Nachbedingungen     | Das Auftreten von Dubletten wird vom Distributor mit der *Distributor-RequestContext/Duplicate an den Endreceiver signalisiert.  |
| Included Use Cases  | -  |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Der Distributor berechnet den Hashwert der eingehenden Meldung.</li> <li>2. Der Hashwert wird mit bereits gespeicherten Hashwerten verglichen.</li> <li>3. Bei Übereinstimmung wird die Meldung als Dublette erkannt und entsprechend signalisiert.</li> <li>4. Es müssen immer eine neue Credentialsinformation mit Key und Passwort zurückgegeben werden. Die alten, bestehenden Zugangsdaten können gelöscht werden. Grundsätzlich dürfen nur echte Dubletten (Meldungen mit gleichem Inhalt) „verdeckt“ werden, d.h. die zeitlich erste Meldung könnte sofort gelöscht werden, sofern sie nicht bereits freigegeben wurde. Zusätzlich muss eine Warning/Notification zurückgegeben werden <a href="#">[[ACK-NOTIF]]</a>.</li> </ol> <p>Dies ist unabhängig von der Freigabe im <a href="#">Abschnitt 8.11, „UC011 Completion“</a>. Eine erneute Übermittlung und Freigabe muss immer möglich sein.<sup>a</sup></p> |
| Alternative Abläufe | -  |
| Fehlerliste         | <p>Fachliche Fehler:</p> <ul style="list-style-type: none"> <li>• Bei dieser Art der Detektierung können nur technische (Byte) Dubletten erkannt werden.</li> </ul>  |

<sup>a</sup>Evtl. wurde in der ersten Übermittlung und Freigabe etwas vergessen zu ergänzen.

Tabelle 8.2. UC002 Duplikate behandeln

### Elemente, die bei der Hashwertberechnung nicht einbezogen werden:

- {urn:ch:swissdec:basis:v1:\*:components}TransmissionDate
- {urn:ch:swissdec:basis:v1:\*:components}RequestID
- {urn:ch:swissdec:basis:v1:\*:components}CreationDate
- Attribut {urn:ch:swissdec:basis:v1:\*:components}addresseeID
- Attribut {urn:ch:swissdec:basis:v1:\*:components}addresseeIDRef

Der Distributor meldet dem Endreceiver eine Dublette mithilfe von Duplicate.

### Einschränkungen der Dublettenerkennung:

- Bei dieser Art der Detektierung können nur technische (Byte) Dubletten erkannt werden.
- Unterschiedliche Reihenfolge der Personen, Löhne, etc.. Fachlich wäre dies ein Duplikat, technisch sind die Daten aber unterschiedlich.
- Bei paralleler Übermittlung ist unter Umständen beim Eintreffen der einen Meldung die Prüfung der anderen noch nicht abgeschlossen. Dadurch fehlt der Hashwert und es wird kein Duplikat erkannt.



- Je nach Standard kann eine solche Dublettenerkennung nicht oder nicht vollständig implementiert werden.

### 8.3 UC003 Testmeldung empfangen

|                     |   |
|---------------------|---|
| Kurzbeschreibung    | Testmeldungen dienen zur <b>Lokalisierung</b> von Problemen in der <b>produktiven Übermittlungskette</b> . Dabei werden Meldungen durch die gesamte Automatisierungskette geschleust, ohne einen echten Geschäftsvorfall anzustossen. Es werden keine Folgeprozesse ausgelöst, und keine Aktionen vom Endempfängersystem vorgenommen. |
| Akteure             | Sendersystem, Distributor, Endreceiver, Endempfängersystem  |
| Auslöser            | Das Sendersystem sendet eine als Test markierte Meldung mit dem Element <code>&lt;TestCase/&gt;</code> .  |
| Vorbedingungen      | Die Meldung ist als Testfall gekennzeichnet.  |
| Nachbedingungen     | Die Testmeldung wurde empfangen und verarbeitet, ohne einen produktiven Geschäftsvorfall auszulösen.<br><br>Der Endreceiver <b>muss</b> das TestCase-Element ebenfalls in der Response setzen.<br><br>Alle weiteren Operationen sowie die Responses dazu <b>müssen</b> als Testcase markiert sein.                                    |
| Included Use Cases  | -   |
| Standardablauf      | 1. Die Meldung wird mit dem Element <code>&lt;TestCase/&gt;</code> empfangen.<br>2. Die Meldung wird wie eine normale Meldung verarbeitet, jedoch ohne produktive Auswirkungen.<br>3. Die Response <b>muss</b> ebenfalls als Testfall markiert zurückgesendet.  |
| Alternative Abläufe | -   |
| Fehlerliste         | -   |

Tabelle 8.3. UC003 Testmeldung empfangen

#### Verwendungszweck und Einschränkungen:

Dieser Use Case soll nur in Ausnahmefällen eine Verwendung finden. Als Demo- oder Entwicklungssystem darf er nicht genutzt werden. Für diese Zwecke stehen die Swissdec Testsysteme zur Verfügung.

Das Ziel ist es, möglichst viele potenzielle Endbenutzer zu einer elektronischen Meldung zu motivieren.

Aus diesem Grund soll dieser Use Case nun von den Endbenutzern vermehrt genutzt werden. Damit können diese erste Erfahrungen mit ihrer produktiven Meldung machen, ohne damit einen echten Geschäftsfall zu generieren.

Grundsätzlich sollen dabei nur **korrekte** und **vollständige** Lohnmeldungen geschickt werden. Einzig das optionale Containerelement `<TestCase/>` muss vom Sendersystem bzw. Distributor zusätzlich übermittelt werden. Der Endreceiver **muss** dieses ebenfalls in der Response setzen.

**Wichtig:** Sobald eine Meldung als Testcase gekennzeichnet ist, **müssen** auch alle folgenden Operationen sowie die Responses dazu als Testcase markiert sein.

### 8.4 UC004 Ersatzmeldung verarbeiten

|                  |   |
|------------------|---|
| Kurzbeschreibung | Meldungen, die das Element <code>&lt;Substitution&gt;</code> enthalten, werden vom Unternehmen als Ersatz für bereits übermittelte und freigegebene Meldung |
|------------------|---|

|                     |   |
|---------------------|---|
|                     | gen verschickt. Hierfür werden bestimmte IDs der vorangegangenen (alten) Meldung eingebunden <a href="#">[[CONTAINERXSD]]</a> . Anhand der IDs <b>müssen</b> die Daten der alten Meldung identifiziert und durch die Daten der neuen Meldung <b>ersetzt</b> werden können.<br><br>Ansonsten verhält sich der Ablauf wie bei einer gewöhnlichen Meldung.   |
| Akteure             | Unternehmen, Distributor, Endreceiver   |
| Auslöser            | Eine bereits übermittelte und freigegebene Meldung muss korrigiert werden.  |
| Vorbedingungen      | Eine vorangegangene Meldung wurde bereits übermittelt und freigegeben.  |
| Nachbedingungen     | Die alte Meldung wurde durch die neue Ersatzmeldung ersetzt.  |
| Included Use Cases  | -   |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Die Meldung mit dem Element <code>&lt;Substitution&gt;</code> wird empfangen.</li> <li>2. Anhand der IDs wird die vorangegangene (alte) Meldung identifiziert.</li> <li>3. Die Daten der alten Meldung werden durch die Daten der neuen Meldung ersetzt.</li> <li>4. Die Ersatzmeldung wird wie eine gewöhnliche Meldung weiter verarbeitet.</li> </ol> |
| Alternative Abläufe | -   |
| Fehlerliste         | <p>Fachliche Fehler:</p> <ul style="list-style-type: none"> <li>• Die vorangegangene Meldung konnte anhand der IDs nicht identifiziert werden. (<a href="#">[[ACKNOTIF]]</a>)</li> </ul>  |

Tabelle 8.4. UC004 Ersatzmeldung verarbeiten

## 8.5 UC005 Synchronisieren

Nach jeder Meldung wird mit dem Aufruf `SynchronizeConsumer` der Fall synchronisiert. Dies kann, muss aber nicht, folgende Elemente beinhalten:

- [Abschnitt 8.6, „UC006 Prozesskontrolle durchführen“](#)
- [Abschnitt 8.7, „UC007 Fall schliessen“](#)
- [Abschnitt 8.8, „UC008 Stories verarbeiten und quittieren“](#)
- [Abschnitt 8.9, „UC009 Stories melden“](#)

Bei der Synchronisierung werden Datenpakete (Stories) ausgetauscht. Das Ziel ist immer die Schliessung des Falls, die Stories dienen dazu, alle nötigen Informationen dazu auszutauschen.

|                    |  |
|--------------------|--|
| Kurzbeschreibung   | Der Fall wird synchronisiert.  |
| Akteure            | Sendersystem, Distributor, Endreceiver   |
| Auslöser           | Der Akteur will mit einem Empfänger Stories austauschen. Abhängig vom Standard findet der Austausch nur vom Endempfänger zum Sendersystem statt oder bidirektional.  |
| Vorbedingungen     | <ul style="list-style-type: none"> <li>• <a href="#">Abschnitt 8.1, „UC001 Initiale Meldung empfangen“</a> wurde erfolgreich durchgeführt</li> <li>• Alle Identifikatoren für den Fall sind korrekt</li> </ul> |
| Nachbedingungen    | Der Fall wurde zwischen Sender und Empfänger synchronisiert. Falls die Synchronisierung nicht abgeschlossen ist, ist der Sender darüber in Kenntnis gesetzt worden.  |
| Included Use Cases | <a href="#">Abschnitt 8.6, „UC006 Prozesskontrolle durchführen“</a><br><br><a href="#">Abschnitt 8.7, „UC007 Fall schliessen“</a>  |

|                     |   |
|---------------------|---|
|                     | <a href="#">Abschnitt 8.8, „UC008 Stories verarbeiten und quittieren“</a><br><a href="#">Abschnitt 8.9, „UC009 Stories melden“</a><br><a href="#">Abschnitt 8.15, „UC015 Security anwenden“</a>   |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Das Sendersystem will mit dem Endreceiver den aktuellen Prozess synchronisieren.</li> <li>2. Der Endreceiver verarbeitet die neu erhaltenen Stories. <a href="#">Abschnitt 8.8, „UC008 Stories verarbeiten und quittieren“</a></li> <li>3. Der Endreceiver stellt Stories in der Response zur Verfügung. <a href="#">Abschnitt 8.9, „UC009 Stories melden“</a></li> <li>4. Der Endreceiver gibt den aktuellen State des Prozesses zurück.</li> <li>5. Der Endreceiver quittiert alle für diesen Fall erhaltenen Stories.</li> </ol>   |
| Alternative Abläufe | <ol style="list-style-type: none"> <li>1. Das Element *DistributorRequestContext/SignatureCertificateUID-BFS ist vorhanden.<br/><br/>Der Endreceiver prüft neben den Credentials auch die Zertifikatsinformationen aus der SUA Signatur.</li> <li>2. Der Akteur wählt keinen Fall und keine Stories aus, die er übermitteln will und sendet ein leeres Synchronize.</li> <li>3. Der Endreceiver analysiert alle offenen Fälle dieses Vertrages und sammelt die DomainID's im Available Element.</li> <li>4. Der Akteur analysiert die erhaltene Antwort, und synchronisiert die Fälle aus dem Available mittels Standardablauf.</li> </ol> <p><b>Für ELM:</b></p> <ul style="list-style-type: none"> <li>• Der Endempfänger stellt zur Beantwortung seiner Fragen eine Webseite zur Verfügung (Completion <a href="#">Abschnitt 8.11, „UC011 Completion“</a>). Dazu werden die URL, die Login-Credentials sowie ein allfälliges Ablaufdatum derselben zurückgeliefert.</li> </ul> <p>Hat der Benutzer die Completion erfolgreich durchgeführt, antwortet der Endempfänger auf das nächste Synchronize mit einer Quittung.</p> |
| Fehlerliste         | <p>Fachliche Fehler:</p> <ul style="list-style-type: none"> <li>• Die Meldung verstösst gegen die Plausibilisierungsregeln</li> <li>• Es wird kein Fall mit dem zu synchronisierenden Fall gefunden</li> </ul> <p>Technische Fehler:</p> <ul style="list-style-type: none"> <li>• Fehler beim Signieren oder Verschlüsseln</li> <li>• Der Endreceiver ist nicht erreichbar</li> <li>• Die vom Sendersystem aufbereitete Meldung entspricht nicht dem Schema (Validität nicht gegeben)</li> <li>• Die vom Sendersystem aufbereitete Meldung ist nicht identifizierbar (ungültige Identifikatoren)</li> </ul>   |

Tabelle 8.5. UC005 Synchronisieren

## 8.5.1 Spezielle Anforderungen

### 8.5.1.1 Unique StoryID

Die StoryID's müssen Fallübergreifend eindeutig sein.

### 8.5.1.2 Available (Neue Informationen verfügbar)

Mit dem Available Element kann der Endempfänger signalisieren, dass unter dem gemeldeten Vertrag neue Informationen zu bestehenden Cases vorhanden sind.

### 8.5.1.3 OpenCase

Mit dem OpenCase Element in der Response kann der Endempfänger dem Transmitter signalisieren, dass ein neuer Case eröffnet wurde. Dieser wird jeweils mit einer Prozessspezifischen Identifikation angegeben.

Anschliessend kann der Transmitter den effektiven Case mittels dieser Identifikation synchronisieren.

### 8.5.1.4 Leere Synchronisierung

Mit einem leeren Synchronize (bestehend aus den Elementen RequestContext, Sender und Addressee) wird beim Endempfänger überprüft, ob Aktualisierungen zu bestehenden Cases vorhanden sind.

## 8.6 UC006 Prozesskontrolle durchführen

|                     |  |
|---------------------|--|
| Kurzbeschreibung    | <p>Bei jeder Synchronisierung <b>muss</b> das Empfängersystem den laufenden Prozess kontrollieren, d.h. überprüfen, was auf beiden Seiten noch offen ist:</p> <ul style="list-style-type: none"> <li>• Welche Stories werden vom Endreceiver noch erwartet ( <code>AwaitStory</code> )</li> <li>• Hat der Endreceiver noch Stories aus anderen Prozessen aus dem selben Vertrag, die er dem Sendersystem liefern muss, müssen diese mittels <code>Available</code> zurückgegeben werden.</li> <li>• Hat der Endreceiver noch Prozesse, bei welchem der aktuelle State vom Sendersystem nicht bestätigt wurde, müssen diese mittels <code>Available</code> zurückgegeben werden.</li> <li>• Sind weitere Änderungen notwendig?</li> </ul> |
| Akteure             | Sendersystem, Distributor, Endreceiver   |
| Auslöser            | Das Sendersystem führt eine Synchronisierung durch.  |
| Vorbedingungen      | <ul style="list-style-type: none"> <li>• <a href="#">Abschnitt 8.1, „UC001 Initiale Meldung empfangen“</a> wurde erfolgreich durchgeführt</li> <li>• Alle Identifikatoren für den Fall sind korrekt</li> </ul>   |
| Nachbedingungen     | <p>Der Endreceiver verarbeitet neue Stories und aktualisiert den Status des Falles.</p> <p>Bei einem Fehlschlag:</p> <ul style="list-style-type: none"> <li>• Fehlermeldung</li> </ul>   |
| Included Use Cases  | keine  |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Das Sendersystem führt eine Synchronisierung durch (siehe <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a> ).</li> <li>2. Der Endempfänger überprüft, ob in der Response <code>AwaitStory</code>-Elemente mitzusenden sind. <b>Achtung!</b> Diese Funktion existiert nicht in allen Swissdec-Standards.</li> <li>3. Der Endempfänger überprüft, ob in der Response <code>Available</code>-Elemente zu senden sind. <b>Achtung!</b> Diese Funktion existiert nicht in allen Swissdec-Standards.</li> </ol>  |
| Alternative Abläufe | keine  |
| Fehlerliste         | Analog zu <a href="#">Abschnitt 8.1, „UC001 Initiale Meldung empfangen“</a> , <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a>   |

Tabelle 8.6. UC006 Prozesskontrolle durchführen

## 8.7 UC007 Fall schliessen

|                     |  |
|---------------------|--|
| Kurzbeschreibung    | Ein Fall wird vom Endrezeiver abgeschlossen. Dieses Vorgehen kann je nach Swissdec-Standard unterschiedlich sein und wird in den <a href="#">Richtlinien für Lohn-datenverarbeitung</a> fachlichen Richtlinien genauer beschrieben.  |
| Akteure             | Sendersystem, Distributor, Endrezeiver   |
| Auslöser            | Seitens Endrezeiver ist ein Fall abgeschlossen.  |
| Vorbedingungen      | <ul style="list-style-type: none"> <li>• Es besteht ein Fall, der das Sendersystem und den Endrezeiver betrifft.</li> <li>• Der Endrezeiver hat den Status des Falles auf den finalen Wert gesetzt.</li> <li>• Der Endrezeiver hat keine Stories mehr zu liefern ( &lt;Available&gt; )</li> </ul>                              |
| Nachbedingungen     | <ul style="list-style-type: none"> <li>• Der Fall wurde von beiden Seiten (Sendersystem und Endempfänger) geschlossen.</li> <li>• Der Fall wurde vom Sendersystem archiviert.</li> </ul>   |
| Included Use Cases  | <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a> , <a href="#">Abschnitt 8.6, „UC006 Prozess-kontrolle durchführen“</a>  |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Der Endrezeiver erhält ein SynchronizeConsumer auf einen abgeschlossenen Fall.</li> <li>2. Der Endrezeiver gibt den finalen Status zurück.</li> </ol>  |
| Alternative Abläufe | <p><b>Der Endrezeiver verlangt weitere Stories</b></p> <p>{nach Schritt 2}</p> <ol style="list-style-type: none"> <li>1. Das Sendersystem lässt den Fall offen und bearbeitet es mit weiteren Synchronisierungen, bis es erneut geschlossen werden kann. ( <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a> )</li> </ol> |
| Fehlerliste         | <p>Technische Fehler gemäss <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a></p> <p>Fachliche Fehler gemäss den Richtlinien</p>  |

Tabelle 8.7. UC007 Fall schliessen

## 8.8 UC008 Stories verarbeiten und quittieren

|                     |   |
|---------------------|---|
| Kurzbeschreibung    | Das Sendersystem meldet eine oder mehrere Stories an den Endrezeiver  |
| Akteure             | Sendersystem, Distributor, Endrezeiver  |
| Auslöser            | Das Sendersystem hat Informationen, die an den Endempfänger gelangen sollten.   |
| Vorbedingungen      | <ul style="list-style-type: none"> <li>• Der Fall wurde dem Endrezeiver erfolgreich gemeldet.</li> <li>• Identifikatoren sind bekannt und korrekt gesetzt.</li> </ul>   |
| Nachbedingungen     | <ul style="list-style-type: none"> <li>• Der Endrezeiver hat die Stories empfangen und quittiert.</li> </ul>  |
| Included UCs        | <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a> , <a href="#">Abschnitt 8.6, „UC006 Prozess-kontrolle durchführen“</a>   |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Der Endrezeiver erhält die Stories vom Distributor in einer SynchronizeConsumer Operation und verarbeitet die erhaltenen Stories.</li> <li>2. Die StoryID's aller vom Endrezeiver empfangenen Stories (vom aktuellen Fall), werden im Element &lt;CaseContext&gt;&lt;ReceivedStoryIDs&gt; in der SynchronizeConsumerResponse eingetragen.</li> <li>3. Die Response wird korrekt aufbereitet ( <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a> )</li> </ol> |
| Alternative Abläufe | <i>Keine Stories erhalten</i>   |

|             |  |
|-------------|--|
|             | <ol style="list-style-type: none"> <li>1. Der Endreceiver quittiert nur erhaltene Stories des aktuellen Fall, hat dabei aber keine neuen Stories zu verarbeiten. Der Request enthält nur einen aktualisierten CaseContext aber keine weiteren Stories.</li> <li>2. Den Empfang quittieren</li> </ol>   |
| Fehlerliste | <ol style="list-style-type: none"> <li>1. Es gibt Probleme bei der Übermittlung. Die Übermittlung <i>kann</i> wiederholt werden.</li> <li>2. Im Element &lt;CaseContext&gt;/&lt;SuppressedSenderStoryIDs&gt; sind StoryID's vorhanden:<br/><br/>Die Story mit diesen IDs können vom Endreceiver nicht empfangen werden. (z.B: Mapping)<br/><br/>Die ID's müssen für jeden folgenden SynchronizeConsumer des aktuellen Prozesses versandt werden.</li> <li>3. Im Element &lt;CaseContext&gt;/&lt;SuppressedInstitutionStoryIDs&gt; sind StoryID's vorhanden:<br/><br/>Die Story mit diesen IDs können vom Transmitter nicht empfangen werden. (z.B: Mapping)<br/><br/>Die ID's müssen für jeden folgenden SynchronizeConsumer des aktuellen Prozesses versandt werden, so wird dem Endreceiver signalisiert, dass der Transmitter diese Story nicht unterstützt.</li> <li>4. Es gibt fachliche Probleme bei der Auswahl der Stories.<br/><br/>Im CaseContext können Info und Warning Notification gesetzt werden. Diese können mit der StoryID versehen werden, so dass der Sender informiert werden kann, welche Story Probleme bereitet.</li> </ol> |

Tabelle 8.8. UC008 Stories verarbeiten und quittieren

## 8.9 UC009 Stories melden

|                  |  |
|------------------|--|
| Kurzbeschreibung | Anfrage des Sendersystems, um Stories zu erhalten.   |
| Akteure          | Sendersystem, Distributor, Endreceiver   |
| Auslöser         | Aufgrund einer Synchronisationsanfrage hat der Endreceiver neue Stories für das Sendersystem.  |
| Vorbedingungen   | <ul style="list-style-type: none"> <li>• Der Fall wurde dem Endreceiver erfolgreich gemeldet.</li> <li>• Identifikatoren sind bekannt und korrekt gesetzt.</li> </ul>  |
| Nachbedingungen  | <ul style="list-style-type: none"> <li>• Der Endreceiver stellt die Stories dem Sender zur Verfügung.</li> </ul>   |
| Included UCs     | <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a> , <a href="#">Abschnitt 8.6, „UC006 Prozesskontrolle durchführen“</a>   |
| Standardablauf   | <p><i>Das Sendersystem hat noch keine Stories</i></p> <ol style="list-style-type: none"> <li>1. Das Sendersystem schickt via Distributor eine Anfrage, um die ersten Stories zu bekommen. ( <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a> )</li> <li>2. Der Endreceiver antwortet mit einem ersten Satz von Stories.</li> </ol> <p><i>Das Sendersystem hat bereits Stories und sendet diese mit Antworten</i></p> <ol style="list-style-type: none"> <li>1. Das Sendersystem schickt via Distributor die bekannten Stories mit Antworten.</li> <li>2. Der Endreceiver verarbeitet die Stories mit den Antworten.</li> </ol> |

|                     |  |
|---------------------|--|
|                     | 3. Aufgrund der gelieferten Stories (und den Antworten) werden neue Stories erstellt und an das Sendersystem zurückgeschickt.                                      |
| Alternative Abläufe | 1. Der Endreceiver muss jeweils alle verfügbaren Stories für das Sendersystem zur Verfügung stellen, welche im aktuellen ReceivedStoryIDs vom Sendersystem fehlen. |
| Fehlerliste         | 1. Es gibt Probleme bei der Übermittlung. Die Übermittlung <i>kann</i> wiederholt werden.<br>2. Es gibt fachliche Probleme bei der Auswahl der Stories.            |

Tabelle 8.9. UC009 Stories melden

## 8.10 UC010 Datenflusskontrolle

|                     |  |
|---------------------|--|
| Kurzbeschreibung    | Die Datenflusskontrolle soll verhindern, dass es zu einer Überlastung der betroffenen Systeme der Übermittlung kommt. Der Endreceiver kann beeinflussen, wie viele Informationen er innerhalb einer Response senden will. Das Sendersystem kann den Umfang des Datenflusses kontrollieren, indem nur ein oder wenige Fälle zur Synchronisation ausgewählt werden.  |
| Akteure             | Sendersystem, Distributor, Endreceiver   |
| Auslöser            | Das Sendersystem fragt mittels Synchronize Operation beim Endreceiver nach aktualisierten Fällen an.   |
| Vorbedingungen      | Das Empfängersystem hat zu mehreren Fällen Informationen bereitgestellt.   |
| Nachbedingungen     | Der Datenfluss wurde kontrolliert und die Überlastung der Systeme verhindert.  |
| Included Use Cases  | -  |
| Standardablauf      | 1. Der Endreceiver teilt dem Sendersystem eingeschränkt mit, zu welchen Fällen er noch Informationen zu liefern hat (<Available>).<br><br>Der Endreceiver muss nicht zwingend alle Availables liefern, sondern kann eine beschränkte Anzahl von Availables zurückgeben.<br>2. Das Sendersystem fragt bei den erhaltenen Fällen nach..<br>3. Der Endreceiver liefert die Antworten zu den abgefragten Fällen und reagiert für nicht abgefragte Fälle weiterhin mit <Available>. |
| Alternative Abläufe | -  |
| Fehlerliste         | Technische Fehler:<br><ul style="list-style-type: none"><li>• Überlastung der Systeme durch unkontrollierten Datenfluss.</li></ul>   |

Tabelle 8.10. UC010 Datenflusskontrolle

## 8.11 UC011 Completion

|                  |   |
|------------------|---|
| Kurzbeschreibung | Die Webseite der Completion eines Endreceivers wird aufgerufen.   |
| Akteure          | Sachbearbeiter  |
| Auslöser         | Der Akteur möchte die übermittelte Meldung ergänzen und freigeben.  |
| Vorbedingungen   | <ul style="list-style-type: none"> <li>• Der Fall wurde dem Endreceiver erfolgreich gemeldet</li> <li>• Basis-URL, Key und Passwort aus der Story des Endreceivers sind verfügbar</li> <li>• Das &lt;ExpiryDate&gt; des Logins für die Completion ist noch gültig.</li> </ul> |
| Nachbedingungen  | <ul style="list-style-type: none"> <li>• Die Daten sind gesichert oder gelöscht.</li> <li>• Personenbezogene Daten sind gemäss Datenschutz gesichert.</li> </ul>  |



|                     |   |
|---------------------|---|
|                     | <ul style="list-style-type: none"> <li>• Wurde eine Meldung erfolgreich freigegeben, so ist es im weiteren Verlauf <b>nicht mehr möglich</b>, ein <code>CompletionReleaseIsMissing</code> zu erhalten.</li> <li>• Bei einem Fehlschlag: Webseite nicht erreichbar: Fehlermeldung</li> </ul>   |
| Included UCs        | Keine   |
| Standardablauf      | <p>1. Der Sachbearbeiter meldet sich mit Key und Password am System an (Login kann sichtbar sein und <b>muss vollständig</b> vorbefüllt werden, d.h. <b>Key und Password</b>).</p> <p>Bei korrekter Anmeldung ist es zulässig, den Logindialog zu überspringen und dem Endbenutzer direkt die erste Maske zu zeigen.</p> <p>Ziel ist ein möglichst einfacher Prozess, d.h. die Lohnbuchhaltung startet den Browser mit „Key“ und „Password“ als Parameter in der Completion-URL, und der Endbenutzer muss nur noch den Login-Knopf drücken. Diese Parameternamen <b>müssen</b> case-insensitiv sein, d.h. sowohl „key“ als auch „Key“ sind gültig.</p> <p>2. Der Sachbearbeiter kontrolliert die gesendeten Lohnaten. Dabei sollen nur relevante Summen bzw. Zahlen angezeigt werden. Auf einzelne Personendaten sollte, wenn immer möglich, aus Sicherheitsgründen verzichtet werden. Zusätzlich kann die Meldung ergänzt und gedruckt werden.</p> <p>3. Die Meldung wird freigegeben. Es ist zulässig, die Completion für einen TestCase aus <a href="#">Abschnitt 8.3, „UC003 Testmeldung empfangen“</a> durchzuführen, sofern der TestCase für den Sachbearbeiter ersichtlich wird (in den Masken und auf der PDF- oder HTML-Quittung).</p> <p>4. PDF- oder HTML-Quittung mit Kontrollsummen und den Ergänzungen an den Sachbearbeiter zurückgeben.</p> <p>5. Löschen von kritischen Daten (gemäss Datenschutz)</p> |
| Alternative Abläufe | <p>{Schritt 3: löschen statt freigeben}</p> <p>Die Meldung <b>muss</b> komplett gelöscht werden, d.h. direkt zum Schritt 5.</p> <p>{Schritt 1: Meldung ist bereits gelöscht}</p> <p>Über den Zustand <i>soll</i> eine Information zurückgeben werden, z.B. Meldung ist „freigegeben zur Verarbeitung“ oder ist „gelöscht ohne Freigabe zur Verarbeitung“. Damit werden überflüssige Supportanfragen vermieden.</p> <p>{Schritt 1: ExpiryDate ist abgelaufen}</p> <p>Aus Sicherheitsgründen <b>muss</b> die Gültigkeit der Completionlogins überprüft werden. Abgelaufene Lohnmeldungen werden automatisch gelöscht, d.h. direkt zum Schritt 5</p> <p>Optional kann der Absender (Unternehmen) vor dem Ablauf der Gültigkeit mittels E-Mail erinnert werden.</p> <p>{Schritt 1: Logindialog }</p> <p>Sollte das direkte Login mit Key und Password in der URL fehlschlagen, dann <b>muss</b> die Möglichkeit bestehen, dass sich der User in einem Logindialog einloggen kann.</p>   |
| Fehlerliste         | Technische Fehler:  |



|                             |
|-----------------------------|
| Anmeldung ist nicht möglich |
|-----------------------------|

Tabelle 8.11. UC011 Completion

### 8.11.1 Standardablauf Completion

Zur Verbesserung der Usability wird der vereinfachte Ablauf standardisiert. Dabei könnten auch domänenspezifische Abläufe modelliert werden. Jede Implementierung sollte diesem Ablauf folgen. Ein Buchhalter kann dadurch die verteilten Lohnmeldungen effizient freigeben. Diese Empfehlung ist für eine effiziente und klare Kommunikation zwischen allen Partnern <sup>3</sup> essentiell.

Der Ablauf umfasst minimal folgende Zustände:

- **ReadyForCompletion**

Die Daten wurden vom Endreceiver erfolgreich empfangen, gesichert und mit akzeptiert quittiert. Die Lohn-daten stehen bereit zur Kontrolle und Freigabe. Die übermittelten Lohn-daten können jetzt mittels Completi-on-Applikation ergänzt werden. Die Lohnmeldung ist nun mit dem Login erreichbar.

- **released**

Die evtl. ergänzten Lohn-daten stehen bereit zur eigentlichen Verarbeitung. Diese Freigabe wird noch mit einem PDF-Dokument (Zusammenfassungsdaten und Ergänzungen) für das Unternehmen (Absender) quit-tiert.

Die Lohnmeldung ist nun mit dem Login nicht mehr erreichbar.

- **removed**

Die übermittelte Lohnmeldung wird gelöscht und es findet keine Verarbeitung statt. Die Lohnmeldung ist nun mit dem Login nicht mehr erreichbar.

Folgende Standardteile *sollen* in der Completion implementiert werden:

1. **Zusammenfassung** (nach dem Login)

Hier möchte der Unternehmer (Lohnbuchhalter, Absender) die wesentlichen Daten seiner Lohnmeldung überprüfen. Weiter mit Ergänzungen, Freigabe oder löschen.

2. **Ergänzungen**

Hier kann oder muss der Unternehmer (Lohnbuchhalter, Absender) zusätzliche Angaben machen. Weiter mit Freigabe oder Löschen.

3. **Freigabe**

Die Freigabe wird noch mit einem PDF oder HTML quittiert.

Es werden keine weiteren Anforderungen an den Ablauf definiert, d.h. GUI-Masken und Abläufe sind nur Empfehlungen. Die Elementlabels sind zum Teil im Swissdecstandard [\[\[CONTAINERXSD\]\]](#) beschrieben. Eine Anpassung muss über die entsprechende Swissdec Fachgruppe erfolgen.

### 8.11.2 Spezielle Anforderungen

#### 8.11.2.1 Allgemeines

Die Completionapplikation *sollte* dem Benutzer einen Hinweis über die anstehenden und freigegebenen Lohn-meldungen liefern. Damit werden überflüssige Supportanfragen vermieden.

<sup>3</sup>Fach- und Technikenebene für Unternehmen (Buchhalter), Lohnbuchhaltungshersteller und Empfänger

Nach einer Meldungsfreigabe mittels Completion **muss** eine weitere Meldungsübermittlung immer zulässig sein (z.B. Substitutionsmeldung). Optional könnte in der Quittung dazu eine Bemerkung stehen (Info oder Warning).

### 8.11.2.2 Completion-URL und Credentials

Das Element `<Credentials>` mit `<Key>` und `<Password>` wird in der XML-Reponse **nicht** in eine URL-kodierte Form gebracht. Dies ist notwendig, da die Web-Eingabeformulare für die Completion keine URL-kodierte Form verlangen und der Benutzer auf diese Weise mit direktem Copy-Paste arbeiten kann, falls es Probleme mit der automatischen Vorbefüllung der Eingabemaske gibt.

Hingegen muss die Completion-URL `<Completion>/<Url>` in eine korrekte, URL-kodierte Form gebracht werden, bevor diese an den Transmitter zurückgeschickt wird.

#### Verbotene Sonderzeichen

Die aufgeführten Sonderzeichen dürfen **nicht** verwendet werden:

- In URL, Key und Password: "#", "<" und ">".
- In Key und Password: "&" und "?".

Daraus folgt auch, dass URLs **keine** Fragmente enthalten dürfen.

Ausserdem **muss** darauf geachtet werden, dass bei allen in URL, Key oder Password enthaltenen Sonderzeichen die Character Entity Reference (z.B. `&amp;` anstelle von „&“) verwendet wird, damit es in der XML-Response nicht zu einem Validierungsfehler kommt. Achtung: Es **muss** dringend darauf geachtet werden, dass der Charakter nicht versehentlich doppelt codiert wird!

Ausgangs-URL; im XML so nicht valid:

```
http://www.completion-url.ch/?language=de&info=test
```

Richtig und im XML valid:

```
http://www.completion-url.ch/?language=de&amp;info=test
```

Falsch, da doppelt codiert:

```
http://www.completion-url.ch/?language=de&amp;amp;info=test
```

#### Abbildung 8.1. URL-Encoding in der Completion

Die meisten Entwicklungsplattformen haben in ihren Bibliotheken Tools zur URL-Kodierung. Die URL Kodierung wird in [\[RFC3986\]](#) beschrieben.

Aufbau einer URL gemäss[\[RFC3986\]](#):

#### Abbildung 8.2. Aufbau URL gemäss RFC3986

### 8.11.2.3 Testfall

Wurde eine Meldung mit `<TestCase>` markiert, dann sind in der Completionapplikation die folgenden Anforderungen zu **erfüllen**:

- Auf **jeder** Seite der Completion ist **klar** ersichtlich, dass es sich um einen Testfall handelt.
- Falls eine Quittung (PDF, etc.) erstellt wird, so muss diese ebenfalls einen **klar** ersichtlichen Hinweis auf den Testfall enthalten.
- Am Ende wird der Benutzer darauf hingewiesen, dass die Daten nicht weiter verarbeitet werden.

## 8.12 UC012 Dialog Message abwickeln

|                     |  |
|---------------------|--|
| Kurzbeschreibung    | Die Ergänzung/Freigabe wird mittels DialogMessage durchgeführt.  |
| Akteure             | Sachbearbeiter   |
| Auslöser            | Der Akteur möchte die übermittelte Meldung ergänzen und freigeben.   |
| Vorbedingungen      | <ul style="list-style-type: none"> <li>• Der Fall wurde dem Endreceiver erfolgreich gemeldet</li> <li>• Der Endreceiver hat eine &lt;DialogMessage&gt; Story gesendet</li> </ul>   |
| Nachbedingungen     | DialogMessage wurde komplett abgewickelt, es ist kein weiterer Aufruf nötig.   |
| Included UCs        | -  |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Der Endreceiver sendet mit der &lt;DialogMessage&gt; Story die zu ergänzenden Daten.</li> <li>2. Der Sachbearbeiter ergänzt die geforderten Angaben und sendet diese mit der &lt;DialogMessage&gt; Story mittels &lt;Synchronize&gt; an den Endreceiver.</li> <li>3. Der Endreceiver bestätigt mit &lt;ReceivedStoryIDs&gt; den Erhalt der DialogMessage.</li> </ol> |
| Alternative Abläufe | <p>DialogMessage nicht abgeschlossen</p> <p>{nach Schritt 3}</p> <ol style="list-style-type: none"> <li>1. Der Endreceiver bestätigt mit &lt;ReceivedStoryIDs&gt; den Erhalt der DialogMessage und verlangt weitere Ergänzungsangaben. {Weiter mit Schritt 3}</li> </ol>   |
| Detailspezifikation | <a href="#">Anhang D, Detailspezifikation DialogMessages</a>   |
| Fehlerliste         | <p>Fehler:</p> <ul style="list-style-type: none"> <li>• Die empfangenen Daten können nicht verarbeitet werden</li> </ul>   |

Tabelle 8.12. UC012 Dialog Message abwickeln

## 8.13 UC013 Anmeldeprozess (SubscribeOrganizationConsumer)

|                  |   |
|------------------|---|
| Kurzbeschreibung | Ein Sendersystem muss sich bei einem Empfängersystem für einen späteren Übermittlungsprozess anmelden können. Mit diesem Anmeldeprozess wird der zukünftige Informationsaustausch über den Distributor zwischen den Beteiligten in die Wege geleitet. |
| Akteure          | Sendersystem, Distributor, Endreceiver  |
| Auslöser         | Ein Sendersystem möchte sich für einen Übermittlungsprozess beim Endreceiver anmelden.  |
| Vorbedingungen   | Das Sendersystem verfügt über einen gültigen Vertrag und ein SUA Zertifikat für die Anmeldung.  |
| Nachbedingungen  | Die Anmeldung wurde erfolgreich durchgeführt und der zukünftige Informationsaustausch ist eingerichtet.   |

|                     |   |
|---------------------|---|
| Included Use Cases  | -   |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Das Sendersystem sendet eine Anmeldungsanfrage an den Distributor.</li> <li>2. Der Distributor leitet die Anfrage an den Endreceiver weiter.</li> <li>3. Der Endreceiver verarbeitet die Anmeldung und bestätigt diese.</li> <li>4. Die Bestätigung wird über den Distributor an das Sendersystem zurück-gesendet.</li> </ol> |
| Alternative Abläufe | -   |
| Fehlerliste         | -   |

Tabelle 8.13. UC013 Anmeldeprozess (SubscribeOrganizationConsumer)

## 8.14 UC014 Unternehmensauthentifizierung (RegisterOrganizationAuthentication)

|                     |   |
|---------------------|---|
| Kurzbeschreibung    | Verschiedene Prozesse verlangen eine eindeutige Authentifizierung eines Unternehmens beim Endreceiver. Dies bedingt eine Signierung mit SUA-Zertifikat. Ein solches SUA-Zertifikat muss mittels des SUA-Prozesses angefordert werden können.  |
| Akteure             | Unternehmen, Sendersystem, Distributor, Endreceiver   |
| Auslöser            | Ein Prozess verlangt eine eindeutige Authentifizierung des Unternehmens mit SUA-Zertifikat.   |
| Vorbedingungen      | Das Unternehmen benötigt ein SUA-Zertifikat für die Authentifizierung.  |
| Nachbedingungen     | Das SUA-Zertifikat wurde angefordert und die Authentifizierung kann durchgeführt werden.  |
| Included Use Cases  | -   |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Das Unternehmen fordert ein SUA-Zertifikat an.</li> <li>2. Der Distributor prüft und verifiziert die Angaben (auch gegen BFS) und liefert nur die Vertragsdaten an den Endreceiver.</li> <li>3. Der Endempfänger prüft die erhaltenen Vertragsdaten und generiert Credentials gemäss <a href="#">Abschnitt 8.1, „UC001 Initiale Meldung empfangen“</a>.</li> <li>4. Beim ersten Synchronize gibt der Endreceiver die Adressdaten des Unternehmens gemäss Schema zurück.</li> <li>5. Der Distributor vergleicht die Daten von Unternehmen, BFS und Endempfänger und versendet bei Erfolg einen Brief mit Passwort an das Unternehmen.</li> <li>6. Das Unternehmen kann sein SUA Zertifikat lösen.</li> </ol> |
| Alternative Abläufe | -   |
| Detailspezifikation | <a href="#">Anhang C, Detailspezifikation Swissdec Unternehmens-Authentifizierung SUA</a>   |
| Fehlerliste         | -   |

Tabelle 8.14. UC014 Unternehmensauthentifizierung (RegisterOrganizationAuthentication)

## 8.15 UC015 Security anwenden

|                  |   |
|------------------|---|
| Kurzbeschreibung | Jede Übermittlung zum Endreceiver muss signiert und verschlüsselt werden. Einzelheiten dazu finden sich in den Dokumenten zur Sicherheit auf Receiverseite[ <a href="#">SECRXPDF</a> ]. Ebenfalls zu beachten sind die zusätzlichen |
|------------------|---|

|                     |  |
|---------------------|--|
|                     | <p>Informationen zur doppelten Signierung (SUA) auf Transmitterseite und deren Auswirkungen auf Seite Endreceiver.</p> <p>Der Transport muss zwingend über HTTPS erfolgen. Das TLS kann mit einem akkreditierten oder einem selbstsignierten Zertifikat verschlüsselt sein.</p> <p>Es wird empfohlen den Transport mittels Two-Way-TLS (mTLS) abzusichern.</p> |
| Akteure             | Distributor, Endreceiver   |
| Auslöser            | Eine Meldung wird übermittelt.   |
| Vorbedingungen      | Die notwendigen Zertifikate für Transport, Signierung und Verschlüsselung sind vorhanden.  |
| Nachbedingungen     | Die Meldung wurde signiert und verschlüsselt übermittelt.  |
| Included Use Cases  | -  |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Die Meldung wird signiert.</li> <li>2. Die signierte Meldung wird verschlüsselt.</li> <li>3. Die signierte und verschlüsselte Meldung wird übermittelt.</li> </ol>   |
| Alternative Abläufe | -  |
| Fehlerliste         | <p>Technische Fehler:</p> <ul style="list-style-type: none"> <li>• Zertifikat ist ungültig oder abgelaufen.</li> <li>• Signierung oder Verschlüsselung schlägt fehl.</li> </ul>  |

Tabelle 8.15. UC015 Security anwenden

## 8.16 UC016 Wartungsfenster

|                     |   |
|---------------------|---|
| Kurzbeschreibung    | Das Endreceiversystem <b>muss</b> die Möglichkeit haben, in einem <code>PingConsumerResponse</code> dem Distributor ein Wartungsfenster mitzuteilen. Wartungsfenster zeigen beabsichtigte Betriebsunterbrüche an, während denen der Distributor keine Meldungen an den Endreceiver weiterleitet und den Benutzer über das Wartungsfenster informiert.   |
| Akteure             | Endreceiver, Distributor  |
| Auslöser            | Ein geplanter Betriebsunterbruch steht bevor.   |
| Vorbedingungen      | Der Endreceiver hat einen geplanten Betriebsunterbruch.   |
| Nachbedingungen     | Der Distributor wurde über das Wartungsfenster informiert und leitet während diesem Zeitraum keine Meldungen an den Endreceiver weiter.   |
| Included Use Cases  | -   |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Der Endreceiver registriert ein Wartungsfenster mit dem Element <code>&lt;PlannedMaintenance&gt;</code> in der <code>PingConsumerResponse</code>.</li> <li>2. Der Distributor speichert die Information über das Wartungsfenster.</li> <li>3. Während des Wartungsfensters leitet der Distributor keine Meldungen an den Endreceiver weiter und informiert den Benutzer.</li> </ol> |
| Alternative Abläufe | <p>{Wartungsfenster löschen}</p> <p>Der Endreceiver kann ein bereits registriertes Wartungsfenster mit dem Element <code>&lt;NoPlannedMaintenance&gt;</code> löschen.</p>   |
| Fehlerliste         | <p>Technische Fehler:</p> <ul style="list-style-type: none"> <li>• Das Wartungsfenster konnte nicht registriert werden.</li> </ul>  |

Tabelle 8.16. UC016 Wartungsfenster

Da der Distributor keine Meldungen speichert, muss der Benutzer seine Meldung zu einem späteren Zeitpunkt wiederholen.

### 8.16.1 Notizen zur Operation PingConsumer

Die Operation `PingConsumer` ist eine einfache, synchrone Request Response Kommunikation.

Damit werden folgende Funktionen aus Use Case 018 [Abschnitt 8.18, „UC018 Erreichbarkeit prüfen“](#) abgedeckt:

Die Verfügbarkeit der Systeme (Distributor und die gekoppelten Receiver der Endreceiver) wird periodisch mittels „Polling“ durch den Distributor getestet. Der Distributor liefert die aktuelle Systemzeit. Sofern ein Wartungsfenster registriert wurde, wird zusätzlich **eine** geplante Start- und Endzeit angegeben.

Der Endreceiver antwortet mit seinem aktuellen Timestamp. Er kann in der Response optional mit dem Element `<PlannedMaintenance>` ein geplantes Wartungsfenster registrieren oder mit dem Element `<NoPlannedMaintenance>` ein bereits registriertes Wartungsfenster löschen.

Durch die Angabe des Zeitpunktes im Request Element `<NextCheck>` kann der Endreceiver seinerseits den „minimalen“ Vorlauf zum Aufschalten des Wartungsfensters ersehen.

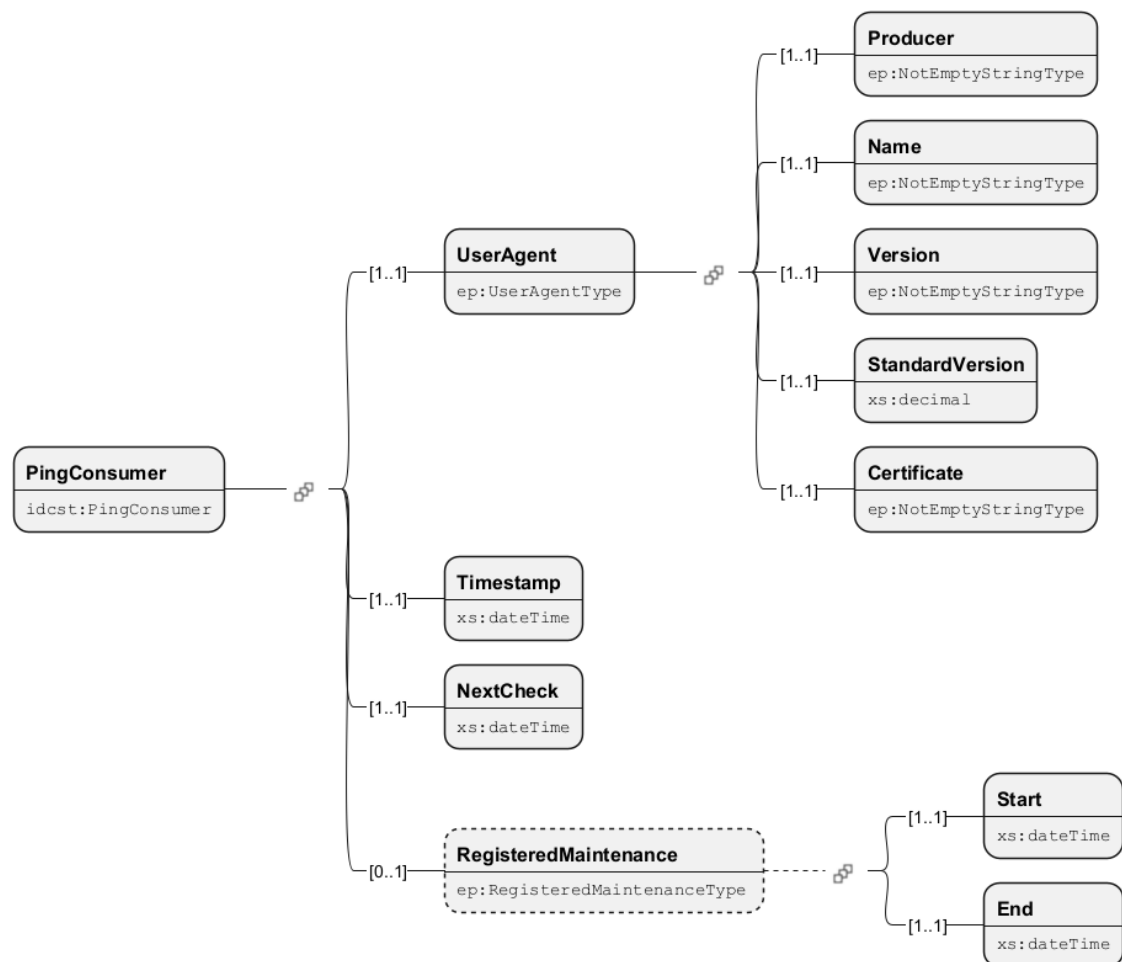


Abbildung 8.3. Schemabild PingConsumer

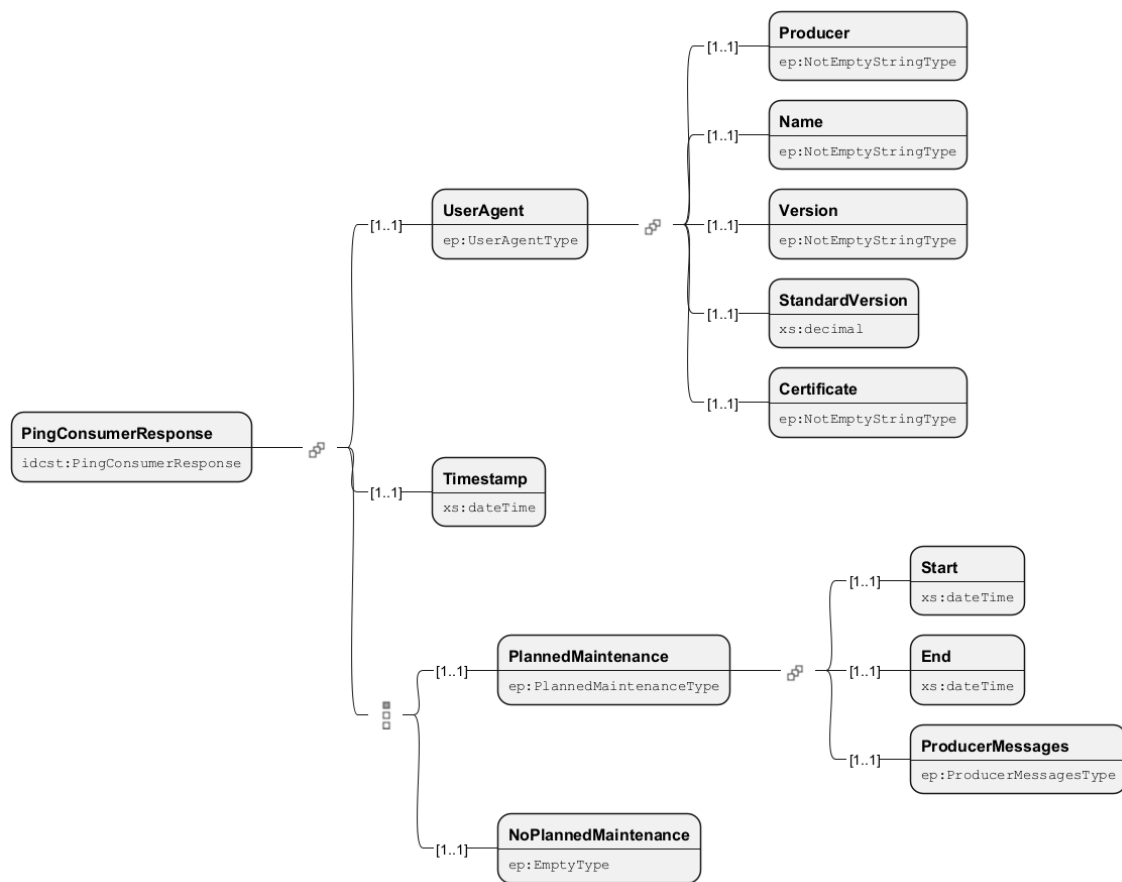


Abbildung 8.4. Schemabild PingConsumerResponse

## 8.17 UC017 Supportanfrage bearbeiten

|                     |  |
|---------------------|--|
| Kurzbeschreibung    | Ausnahmen, Störfall und andere Probleme behandeln  |
| Akteure             | Lohnbuchhalter, IT-Spezialist  |
| Auslöser            | Der Lohnbuchhalter oder IT-Spezialist stellt per Mail oder Telefon eine Supportanfrage   |
| Vorbedingungen      | Keine  |
| Nachbedingungen     | Die Supportanfrage konnte erfolgreich bearbeitet werden  |
| Included Use Cases  | Keine  |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Es wird per Mail oder Telefon eine neue Supportanfrage von einem Lohnbuchhalter oder IT-Spezialist gestellt.</li> <li>2. Das Problem wird analysiert und beantwortet.</li> </ol> |
| Alternative Abläufe | <p>{nach Schritt 1}</p> <ol style="list-style-type: none"> <li>1. Das Problem wird eskaliert und gelangt zum Second oder Third Level Support.</li> </ol> <p>{weiter mit Schritt 2}</p>                                     |
| Fehlerliste         | Keine  |

Tabelle 8.17. UC017 Supportanfrage bearbeiten

Wichtig ist im Umgang mit Supportfällen, dass die Supportinformationen einheitlich kommuniziert werden. Fehler, Warnungen und Informationen müssen gemäss [\[ACKNOTIF\]](#) erstellt und in die Quittung eingefügt werden. Die in diesem Dokument beschriebenen Codes sind verbindlich.

Es **muss** eine Möglichkeit bestehen, im Falle einer Supportanfrage auf die nötigen Informationen für die Behandlung des Problems zuzugreifen, wie zB. mit Hilfe der DeclarationID, der RequestID oder der DistributorRequestID.

### 8.17.1 Support und Reaktionszeit

Es werden nur technische Aspekte zum Support festgelegt, d.h. hier werden nur Informationsstrukturen für alle Systeme in der Prozesskette definiert. Die organisatorische Verwendung dieser Daten erfolgt in einem separaten Dokument.

Der Support **muss** in den Sprachen Deutsch, Französisch und Italienisch für folgende Bereiche bzw. Akteure erbracht werden:

- Unternehmen und ihre Lohnbuchhaltungshersteller
- Endreceiver Institutionen

D.h. auch Fehlermeldungen sind in den entsprechenden Sprachen auszugeben. Siehe in der Meldung:

.../RequestContext /LanguageCode

Zur Bestimmung einer Reaktionszeit werden folgende **Fehlerklassen** definiert

- Critical = 15 Min
- Medium = 4 h
- Uncritical = 1 Tag

Diese Fehlerklassen werden in verschiedenen Systemen (Applikationen, Logfiles, Überwachungstool, ...) später entsprechend verwendet.

Zusätzlich **muss** der 2nd Level Support zu den Applikationsentwicklern koordiniert werden.

### 8.17.2 Spezielle Anforderungen

#### 8.17.2.1 Identifikation eines Geschäftsvorfalles

Der Distributor fügt bei den Meldungen eine Identifikation des Geschäftsvorfalles ein. Für Rückfragen an den Support des Endreceivers **müssen** alle Meldungen eines Geschäftsvorfalles anhand dieser ID zugeordnet werden können. Der Empfänger muss die ID für einen Geschäftsvorfall anzeigen können.

Diese ID dient im Supportfall der Identifikation von Meldungen, die zum selben Geschäftsvorfall gehören. Ein Geschäftsvorfall ist die abgeschlossene Übermittlung einer Meldung. Die Übermittlung einer Ersatzmeldung unterbricht und beendet diesen Geschäftsvorfall. Es beginnt ein eigener, neuer Geschäftsvorfall, für den eine neue ID generiert wird.

### Anmerkung

Der Name der Identifikation des Geschäftsvorfalles ist prozess- beziehungsweise operationsabhängig. In den operationsspezifischen Kapiteln wird auf den Namen hingewiesen.

Die Identifikation des Geschäftsvorfalles ist identisch in allen Requests (nicht vorhanden in: Transmitter → Distributor), Responses, Masken und Pdfs, die zu ein und demselben Geschäftsvorfall gehören. Damit kann sie als Fallnummer für den Support verwendet werden.



Zusätzlich dienen die RequestID und die Distributor-ResponseID - Paare zur Identifikation der einzelnen Requests und Responses innerhalb des Geschäftsvorfalles. Das System muss die RequestID und die ResponseID ebenfalls verwenden und sie der Identifikation des Geschäftsvorfalles zuordnen können.

## 8.18 UC018 Erreichbarkeit prüfen

|                     |  |
|---------------------|--|
| Kurzbeschreibung    | Die Erreichbarkeit des Endreceivers muss geprüft werden. Dazu wird eine einfache Anfrage an den Endreceiver geschickt. Die Rückantwort des Endreceivers bestätigt die Erreichbarkeit.  |
| Akteure             | Distributor, Swissdec Operator, Endreceiver  |
| Auslöser            | Zyklische Überprüfung vom Distributor, Swissdec Operator im Störfall   |
| Vorbedingungen      | Keine  |
| Nachbedingungen     | Keine  |
| Included UCs        | Keine  |
| Standardablauf      | <ol style="list-style-type: none"> <li>1. Der Distributor stösst den Erreichbarkeitstest an.</li> <li>2. Die Anfrage wird an den Endreceiver gesendet. Zusätzlich wird das Intervall des Pollings mitgeteilt. Intervall: zurzeit 30 Minuten (auch während eines Wartungsfensters; Intervall ist damit dynamisch)</li> <li>3. Der Endreceiver antwortet mit seinem aktuellen Timestamp. Optional kann dem Distributor ein geplantes <b>Wartungsfenster</b> (Nichtverfügbarkeit von x bis y) mitgeteilt werden. Diese Funktion <b>muss</b> möglich sein.</li> <li>4. Sollte ein Wartungsfenster gemeldet werden, kann diese Information <b>automatisch</b> einem allfälligen Absender (Unternehmen) direkt vom Distributor weitervermittelt werden.</li> </ol> |
| Alternative Abläufe | Keine  |
| Fehlerliste         | <p>Fehler:</p> <ul style="list-style-type: none"> <li>• der Endreceiver ist nicht erreichbar.</li> </ul> <p>In diesem Fall sendet der Distributor via Mail eine Warnung an die eigens dafür eingerichtete Mailadresse des Endreceivers.</p>  |

Tabelle 8.18. UC018 Erreichbarkeit prüfen

Der Endreceiver ist **verpflichtet**, den Posteingang der entsprechenden Mailadresse **zu überwachen** und bei Nichterreichbarkeit **die notwendigen Massnahmen zu treffen**.

### 8.18.1 Verfügbarkeit

Die Betrachtungseinheit umfasst den Distributor und alle gekoppelten Endreceiver, d.h. das Unternehmen erlebt das ganze System als Einheit. Sollte ein Endreceiver 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 Swissdecstandard

- 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 Endreceiver **müssen** mindestens zum **Empfangen der Daten einen 7x24h Dienst anbieten**.

- **Geplante Unterbrüche** <sup>4</sup> müssen an Randzeiten durchgeführt werden und **müssen** vorher angekündigt werden.
- Nach **ungeplantem Unterbruch** *sollten* betroffene Unternehmen, die eine missglückte Übermittlung hatten, automatisch über die erneute Verfügbarkeit des Empfängers benachrichtigt werden (siehe dazu [Abschnitt 8.1, „UC001 Initiale Meldung empfangen“](#)). Allerdings gibt es in dieser Version keine automatische Benachrichtigung seitens Distributors, d.h. der Endreceiver kann nur die ihm direkt bekannten Unternehmen benachrichtigen.
- Sollten interne Dienste zur Überprüfung der Akzeptierung **nicht zur Verfügung** stehen, *kann* trotzdem mit einer Akzeptierung (inklusive Completion) quittiert werden. Dies *sollte* mit einer Warning/Notification in der Quittung dem Absender mitgeteilt werden. Führt eine spätere Datenprüfung zur Ablehnung der Meldung, muss diese dem Kunden ausserhalb dieser Systemspezifikation mitgeteilt werden (siehe auch [Abschnitt 8.11, „UC011 Completion“](#): Lohnmeldung freigeben (Completion)).
- Sollte die Completion ebenfalls nicht verfügbar sein, *sollte* dies mit einer Warning/Notification in der Quittung dem Absender mitgeteilt 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. Bezüglich Verfügbarkeit ist keine Kontrolle vorgesehen. Deshalb werden hier nur die wesentlichen Richtwerte definiert und entsprechende Grundlagen in den Anhang verschoben.

#### **8.18.2 Definierte Zeitbereiche**

- Betriebszeit des gesamten Systems (Distributor, Kommunikation und Endreceiver; m2m Strecke bis zur Quittungs-Response an das Unternehmen)
  - **7 Tage pro Woche mal 24 Stunden**
  - **Spitzenmonate: Januar und Februar**
  - **Quellensteuer, Statistik und «EMA» werden monatlich gemeldet**
  - **Spitzenzeiten: 6 Uhr bis 20 Uhr und von Dezember bis April**
- Wartungsfenster für Korrekturen und Updates
  - **10 Stunden pro Woche**
  - **Ausserhalb der Spitzenzeiten und Spitzenmonate, wenn möglich zwischen 2 Uhr und 5 Uhr morgens**
- Service- und Supportzeit für die Systemteilnehmer (Distributor und seine Endreceiver wie Versicherungen und Behörden)
  - **Montag bis Freitag von 8 Uhr bis 18 Uhr in den Spitzenmonaten**
  - **In den restlichen Monaten gelten die üblichen Bürozeiten**
  - **Support für Wartungsfenster auf Anmeldung**

#### **8.18.3 Definierte Wertebereiche**

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

<sup>4</sup>Gilt für normale Wartungsarbeiten; ausgenommen ist ein Hotfix oder Patch

Einerseits haben wir pro Unternehmen mit nur einer Übermittlung pro Jahr zu rechnen, d.h. der Endbenutzer hat wenig Erfahrung mit dem gesamten System. Deshalb muss jede Störung<sup>5</sup> vermieden werden. Andererseits möchten Endreceiver mit einem kleineren Kundenstamm keinen erhöhten Aufwand betreiben.

- In den **Spitzenzeiten** soll die Verfügbarkeit der Endreceiver (m2m) mindestens **99,52%** sein.
- In den **Randzeiten** soll die Verfügbarkeit der Endreceiver (m2m) mindestens **93,00%** sein.

---

<sup>5</sup>Störung meint hier: Der Lohnbuchhalter des Unternehmens kann *nicht alles* mit *einem* „einfachen“ Knopfdruck melden. Im Störfall, d.h. ein Endreceiver ist nicht verfügbar, benötigt er Prozesswissen, um dieses Problem zu beheben (Alles noch einmal übermitteln? Nur diesen noch einmal übermitteln? Support anfragen? Endreceiver anfragen?). Diese Unsicherheit führt zu höheren Prozesskosten und einer schlechteren Akzeptanz.

## A Referenzierte Dokumente

[CONTAINERXSD] *Container Schema*. Swissdec.

[ACKNOTIF] *Richtlinien für Lohndatenübermittlung*. Quittierung und Notifications. Swissdec. <https://infopoint.swissdec.ch>.

[UpgrDistri\_Anforderungen] *Upgrade Distributor*. Anforderungen. Swissdec.

[SECPDF] *Richtlinien für Swissdec-Übermittlungen*. Sicherheit (Transmitter). Swissdec. <https://infopoint.swissdec.ch>.

[SECRXPDF] *Richtlinien für Swissdec-Übermittlungen*. Sicherheit (Endreceiver). Swissdec. <https://infopoint.swissdec.ch>.

[Richtlinien für Lohndatenverarbeitung] *Fachliche Swissdec-Richtlinien*. Fachliche Spezifikation. Swissdec. <https://www.swissdec.ch>.

[TFBASIS] *Testfälle Übermittlungstest Basisdienste*. <https://www.swissdec.ch>.

[RFC3986] *Uniform Resource Identifier (URI): Generic Syntax*. ietf. <https://datatracker.ietf.org/doc/html/rfc3986>.

## B Glossar

### D

|             |   |
|-------------|---|
| Distributor | Der Distributor ist das zentrale System, das Daten vom Transmitter erhält, validiert, plausibilisiert und an die adressierten Endreceiver weiterleitet, sowie die erhaltenen Antworten zurück an den Transmitter sendet. Er übernimmt die Filterung und Verteilung der Daten. |
| Domäne      | Organisation, das Daten übermittelt werden. Im Swissdec-Ökosystem bekannte Domänen sind; AHV, FAK, UVG, UVGZ, KTG, BVG, Lohnausweis, Quellensteuer, Grenzgänger und Statistik.  |

### E

|                 |  |
|-----------------|--|
| Empfängersystem | Das Empfängersystem erhält seine Daten vom Endreceiver, um diese dann auf fachlicher Ebene verarbeiten zu können. Es stellt ausserdem die Antworten bereit, die danach über den Endreceiver zurück an den Distributor und das Sendersystem übermittelt werden sollen. (Beispiel: Versicherer, Behörde)       |
| Endreceiver     | Der Endreceiver ist das technische Gegenstück zum Transmitter. Er empfängt und validiert die vom Distributor erhaltenen Daten und leitet diese an ein Empfängersystem weiter. Die vom Empfängersystem bereitgestellten Antworten werden danach vom Endreceiver in die Response an den Distributor eingefügt. |

### I

|                |   |
|----------------|---|
| Initialmeldung | Die erste Meldung, mit der sich ein Sendersystem an den/die Endempfänger wendet.  |
| Institution    | <p>Empfänger, der Daten erhält. Hier handelt es sich um Versicherungen, die den jeweiligen Domänen angehören.</p> <p>Eine Firma kann innerhalb einer Domäne mehrere Institutionen kontaktieren. Eine Institution kann mehrere Domänen unterstützen.</p> |

### S

|              |   |
|--------------|---|
| Sendersystem | Beim Sendersystem handelt es sich um ein System, das Daten aufbereitet und für den Versand an Endempfänger bereitstellt. Hier werden vor allem die fachlichen Anforderungen technisch korrekt umgesetzt (Beispiel: ERP)   |
| Story        | Einheit, die einen Aspekt der gesamten Übermittlung beschreibt. Abhängig vom Standard kann es sich hier um Quittungen, zusätzliche Fallinformationen oder anderes handeln. Stories können vom Sendersystem oder dem Endempfänger aus übermittelt werden und beinhalten Informationen, die die andere Seite zur Weiterverarbeitung des Falls benötigt. |

### T

|             |   |
|-------------|---|
| Transmitter | Der Transmitter übermittelt die vom Sendersystem bereitgestellten Daten an den Swissdec Distributor und empfängt die erhaltenen Antworten (Responses), um diese zu verifizieren und an das Sendersystem weiterzuleiten. |
|-------------|---|

## U

|              |  |
|--------------|--|
| UID          | Jedes in der Schweiz aktive Unternehmen erhält eine einheitliche Unternehmens-Identifikationsnummer (UID). Diese UID wird im Auftrag des Bundes vom Bundesamts für Statistik BFS geführt. Beispiel UID: CHE-111.111.111. |
| UID-Register | Das Bundesamt für Statistik BFS führt das Register ( <a href="https://www.uid.admin.ch">https://www.uid.admin.ch</a> ), wo alle Unternehmen mit ihrer UID aufgelistet sind.  |

## C Detailspezifikation Swissdec Unternehmens-Authentifizierung SUA

### C.1 Einleitung

Verschiedene Swissdec Prozesse verlangen die eindeutige Authentifizierung des übermittelnden Unternehmens. Dies wird im Rahmen der "Swissdec Unternehmens-Authentifizierung" SUA gemacht. Aus technischer Sicht **muss** jedes Unternehmen ein von Swissdec ausgestelltes und signiertes SUA Zertifikat beantragen, das anschliessend für die zusätzliche elektronische Unterschrift (Signatur) der einzelnen Übermittlungen verwendet werden kann.

Vereinfachter Ablauf für jedes Unternehmen:

- Registrierung und Erstellung des SUA Zertifikats
- Verwendung des SUA Zertifikats zur Signierung von Übermittlungen
- Erneuerung oder Sperrung des SUA Zertifikats

Ein SUA Zertifikat kann während seiner Gültigkeitsdauer für alle Standards und Endreceiver verwendet werden. Somit muss ein Unternehmen nur ein SUA Zertifikat beantragen und kann dies für alle anderen Standards und Prozesse wiederverwenden.

Grundlage einer SUA Registrierung ist eine bestehende Geschäftsbeziehung des Unternehmens zu einer Versicherung, welche Identität des Unternehmens bereits geprüft hat. Auf diese überprüfte Identität stützt sich Swissdec für die Identifikation des Unternehmens. Der Distributor prüft während der Registrierung die Angaben zum Unternehmen sowie die bestehende Vertragsbeziehung beim Versicherer.

Folgende Angaben pro Unternehmen müssen bekannt sein:

| Benötigte Angaben  | Validierung  |
|--|--|
| Name des Unternehmens  | Identisch mit Angaben aus dem UID-Register   |
| UID Nummer des Unternehmens  | Identisch mit Angaben aus dem UID-Register<br>Identisch mit Angaben beim Versicherer |
| Bestehende Vertragsverbindung<br>(Vertragsnummer und Kundennummer) | Identisch mit Angaben beim Versicherer   |

### C.2 SUA Prozess

#### C.2.1 Registrierung und Erstellung des SUA Zertifikats

Prozess-Schritt: Ein Unternehmen beweist seine Identität und bekommt das Unternehmen spezifische SUA Zertifikat vom Distributor.

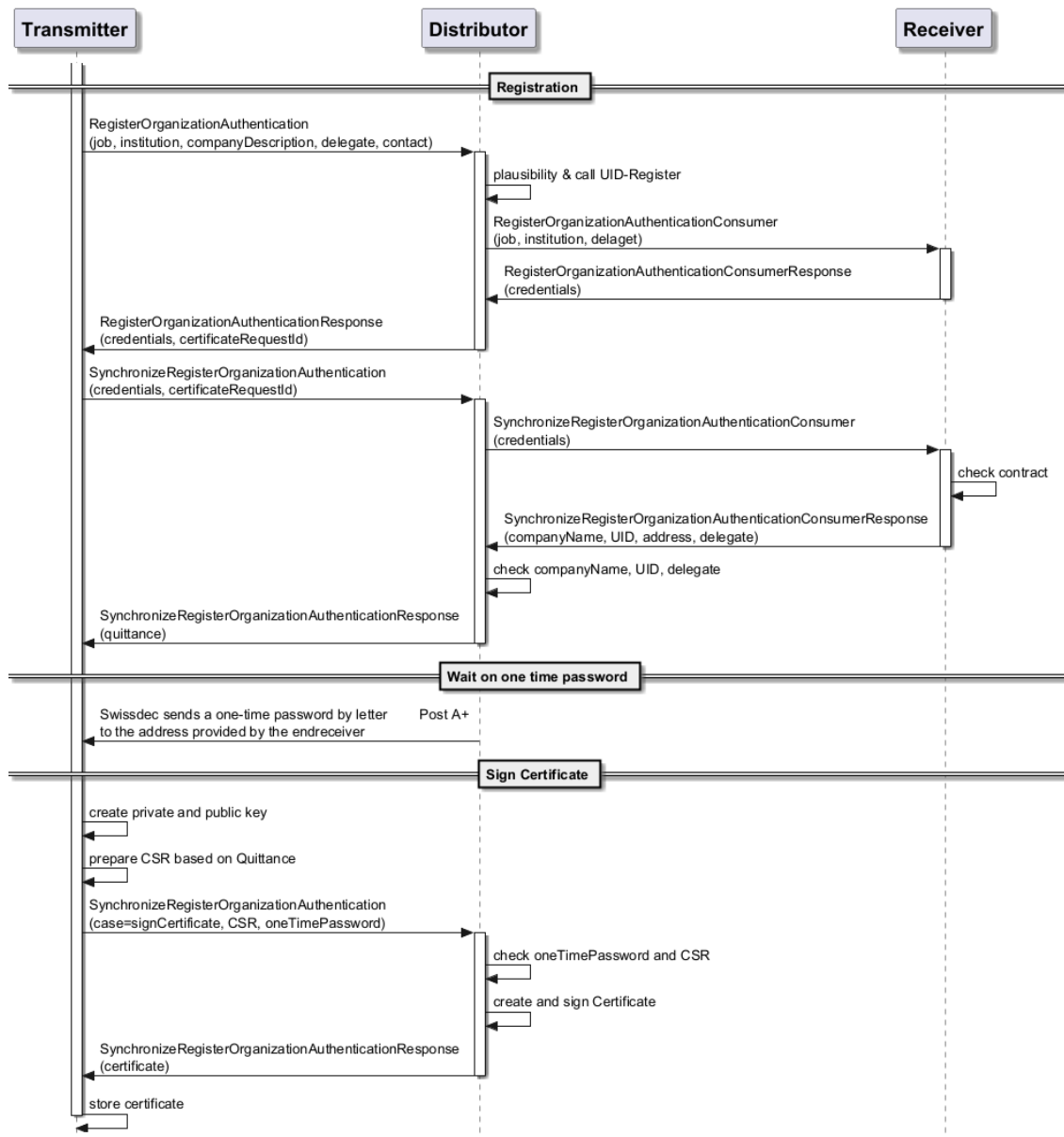
Ein Mitarbeiter des Unternehmens meldet die benötigten Stammdaten des Unternehmens über den Transmitter beim Distributor an. Dafür werden die UID-Nummer, der Firmenname sowie die bestehende Vertragsbeziehung mit dem adressierten Versicherer an den Distributor gemeldet.

Der Distributor überprüft die Stammdaten des Unternehmens und vergleicht diese mit den Angaben aus dem UID-Register. Das Unternehmen muss gemäss UID-Register zu diesem Zeitpunkt aktiv ist. Zusätzlich werden die Angaben zur Vertragsverbindung dem adressierten Versicherer übermittelt. Der Versicherer sendet die Unternehmensdaten und dessen Adresse zurück. Beim Versicherer muss die identische UID des Unternehmens hinterlegt sein. Im Erfolgsfall wird eine Quittung an den Transmitter zurückgesendet.

Nach erfolgreicher Prüfung auf dem Distributor wird dem Transmitter eine Quittung zurückgesendet, die wichtige Informationen zum Erstellen des SUA Zertifikats enthält. Zu diesem Zeitpunkt wird ein Brief an die beim Versicherer hinterlegten Adresse des Unternehmens gesendet. Dieser Brief enthält ein einmaliges Registrie-

rungepasswort sowie das Sperrpasswort. Der physische Versand wird aus Sicherheitsgründen gemacht und stellt sicher, dass das korrekte Unternehmen Zugriff erhält und ein kompromittiertes System oder Angreifer kein SUA Zertifikat beantragen kann.

Das Registrierungspasswort wird zusammen mit der Zertifikatsanfrage, einem [Abschnitt C.3.3, „Certificate Sign Request \(CSR\)“](#) mit den Angaben aus der Quittung, an den Distributor gesendet. Der CSR wird sowie das Passwort werden vom Distributor überprüft. Erst wenn alles korrekt ist, erstellt der Distributor über eine von Swissdec autorisierte Certificate Authority (CA) das SUA Zertifikat und sendet es an den Transmitter zurück. Der Transmitter installiert das SUA Zertifikat und stellt sicher, dass nur das registrierte Unternehmen sein eigenes SUA Zertifikat verwenden kann.



Ist eine Registrierung nicht gültig oder unvollständig, so wird ein entsprechender Fehler zurückgegeben. Diese Fehler sind im Dokument [\[ACKNOTIF\]](#) spezifiziert.

### C.2.1.1 Registrierung als Stellvertreter eins Unternehmens (Delegate)

Als Stellvertreter eines Unternehmens kann ein anderer Benutzer im Auftrag des Unternehmens die Registrierung durchführen. In der Registrierung muss der Stellvertreter entsprechend deklariert werden. Als Sicher-



heitsmassnahme muss der identische Stellvertreter beim Versicherer auch hinterlegt sein, was der Distributor im Rahmen der Registrierung überprüft.

### C.2.1.2 Registrierung als Testcase

Aus Sicherheitsgründen kann der SUA-Prozess als Testfall gestartet, jedoch nicht abgeschlossen werden. Die Registrierung mit Testcase kann informell zur Überprüfung durchgeführt werden, ob eine Registrierung möglich ist. In Falle des Testcase werden alle fachlichen Prüfungen normal durchgeführt jedoch anstelle der positiven Quittung ein spezifischer Fehlercode zurückgegeben, der auf den Testcase hinweist.

### C.2.2 Verwendung des SUA Zertifikats

Ziel: Das Unternehmen verwendet das SUA Zertifikat zum zusätzlichen Signieren von Meldungen, um zu beweisen, dass die Meldung vom Unternehmen gesendet.

Das ausgestellte und gültige SUA Zertifikat wird als digitaler Ausweis des Unternehmens verwendet. Der Transmitter kann Meldung zusätzlich mit dem SUA Zertifikat signieren, dadurch kann der Distributor sowie die Endreceiver eindeutig prüfen, von welchem Unternehmen die Daten stammen und ob sie unterwegs verändert wurden.

Die technische Verwendung des SUA Zertifikats zur Erstellung der zusätzlichen Signatur wird im Dokument [\[SECPDF\]](#) spezifiziert.

### C.2.3 Erneuerung des SUA Zertifikats

Ziel: Das Unternehmen erneuert sein SUA Zertifikat bevor es abläuft.

Das SUA Zertifikat ist zeitlich begrenzt. Kurz vor Ablauf initiiert der Transmitter automatisch einen Erneuerungsprozess, damit die Kommunikation nicht unterbrochen wird. Dabei sendet die Software eine neue Zertifikatsanfrage an den Distributor. Die Erneuerung kann nur während einer bestimmten Frist vor Ablauf des Zertifikats durchgeführt werden, um stetige Erneuerungen zu verhindern.

Eine Neuregistrierung des Unternehmens ist nicht immer erforderlich, da die Identität bereits bekannt ist. Aktuell kann ein SUA Zertifikat nicht unbegrenzt erneuert werden. Ist eine Zertifikat abgelaufen, kann die Erneuerung nicht mehr durchgeführt werden. In diesem Fall muss eine neue Registrierung durchgeführt werden.

Die genauen Anforderungen an die Gültigkeitsdauer, Erneuerungen und Fristen für die Erneuerung sind im Kapitel [Abschnitt C.3.1, „Erneuerung der SUA Zertifikate“](#) spezifiziert.

### C.2.4 Sperrung des SUA Zertifikats

Ziel: Das Unternehmen kann ein SUA Zertifikat von Swissdec sperren lassen

Die Sperrung soll durchgeführt werden, wenn ein ausgestelltes SUA Zertifikat nicht mehr vertrauenswürdig ist. Dies ist bei Schlüsselverlust, bei Sicherheitsvorfällen beim Transmitter oder Unternehmen sowie bei Wechsel des Transmitters notwendig. Es kann auch auf Wunsch des Unternehmens gemacht werden.

Zur Authentisierung verwendet die Firma das bei der Registrierung erhaltene Sperrpasswort und meldet damit die Sperrung an den Swissdec Support. Anschließend wird das Zertifikat bei der Certification Authority widerrufen. In der Fachsprache wird dies als "revoked" bezeichnet. Ab diesem Zeitpunkt soll der Transmitter das gesperrte SUA Zertifikat nicht mehr verwenden. Der Distributor würde jede mit dem gesperrten SUA Zertifikat signierte Meldung ablehnen.

## C.3 Technische Anforderungen an die Zertifikate

Diese nachfolgenden technischen Anforderungen haben dynamischen Character und können im Rahmen eines Distributor Releases den steigenden Sicherheitsanforderungen angepasst werden. Die Anpassungen werden mit dem Release von Swissdec kommuniziert und sind im Dokument [\[SECPDF\]](#) enthalten.

### C.3.1 Erneuerung der SUA Zertifikate

| Kontext                          | Wert            | Beschreibung   |
|----------------------------------|-----------------|--|
| Sicherheitsmerkmal Registrierung | Brief/ A+       | Passwörter versendet an die beim Versicherer hinterlegten Adresse.<br>Zweiter, nicht elektronischer Kanal. |
| Registrierungspasswort           | Min. 12 Zeichen | Zahlen, Gross- und Kleinbuchstaben   |
|                                  | 1 Jahr          | Gültigkeitszeitraum  |
| Sperrpasswort                    | Min. 12 Zeichen | Zahlen, Gross- und Kleinbuchstaben   |
|                                  | 5 Jahr          | Gültigkeitszeitraum  |
| Erneuerungsprozess               | 60 Tage         | Zeitraum vor Ablauf des Zertifikats, ab welchem der Transmitter die Erneuerung starten darf.               |
|                                  | 3x (3 Jahre)    | Anzahl möglicher automatischer Erneuerungen  |

### C.3.2 Aufbau Zertifikat nach X509

Die SUA Zertifikate werden als X509 Zertifikate nach RFC 5280 von einer von Swissdec autorisierten Certificate Authority (CA) ausgestellt. Das dazugehörige Schlüsselpaar (Private + Public Key) muss vom Transmitter in einer sicheren Infrastruktur erzeugt und persistiert werden. Der Transmitter ist für die Sicherheit des Private Keys verantwortlich. Zu Erstellung und Signierung des SUA Zertifikats wird nur der Public Key verwendet, um den CSR zu signieren. Damit wird sichergestellt, dass der Private Key das Transmitter System nie verlassen muss oder darf.

| Attribut              | Wert                                      | Beschreibung                               |
|-----------------------|---|--|
| <b>Zertifikat</b>     |   |  |
| Version               | 3   | Zertifikatsversion gemäss RFC 5280         |
| Serial Number         | {Wert}                                    | Eindeutige Identifikation des Zertifikats. |
| Validity              | 1 Jahr                                    | Zeitraum der Gültigkeit des Zertifikats    |
| Public Key            | RSA 2048<br>RSA 3072<br>RSA 4096          | Akzeptierte Schlüssellänge                 |
| Signature Algorithm   | Sha256WithRSA                             | Verwendeter Signature Algorithmus          |
| Signature Value       | {Wert}                                    | Signatur des Zertifikats                   |
| <b>Issuer</b>         |   |  |
| CN                    | Verein Swissdec<br>Issuing CA by DigiCert | Common Name (OID 2.5.4.3)                  |
| O                     | Verein Swissdec                           | OrganizationName (OID 2.5.4.10)            |
| C                     | CH  | CountryName (OID 2.5.4.6)                  |
| <b>Subject</b>        |   |  |
| C                     | CH  | CountryName (OID 2.5.4.6)                  |
| ST<br><i>Optional</i> | {Kanton des Unternehmens}                 | StateOrProvinceName (OID 2.5.4.8)          |
| L<br><i>Optional</i>  | {Ort des Unternehmens}                    | LocalityName (OID 2.5.4.7)                 |
| CN                    | NTRCH-{UID}@swissdec.ch                   | Common Name (OID 2.5.4.3)                  |
| O                     | {Name aus dem UID Register}               | OrganizationName (OID 2.5.4.10)            |

| Attribut | Wert        | Beschreibung                          |
|----------|-------------|---------------------------------------|
| ORG_ID   | NTRCH-{UID} | OrganizationIdentifier (OID 2.5.4.97) |

### C.3.3 Certificate Sign Request (CSR)

Ein Certificate Signing Request (CSR) ist die Anfrage, um ein digitales Zertifikat zu erhalten. Der CSR enthält den gewünschten Subject und den PublicKey, für den das Zertifikat ausgestellt werden soll. Diese Angaben werden von Distributor überprüft und müssen der Quittung entsprechen, die bei der Registrierung erstellt wurde.

| Attribut            | Beschreibung  |
|---------------------|---|
| Format              | PEM, PKCS #10, gemäss RFC 2986  |
| Public Key          | Der öffentliche Schlüssel, der vom Transmitter generiert wurde  |
| Subject             | Informationen zum Zertifikatsinhaber gemäss Quittung nach <a href="#">Abschnitt C.3.2</a> , „Aufbau Zertifikat nach X509“ |
| Signature Algorithm | Sha256WithRSA   |
| Signature Value     | Signatur des Zertifikats, erstellt mit dazugehörigem Private Key  |

## D Detailspezifikation DialogMessages

### D.1 Einleitung

Eine DialogMessage wird dazu verwendet, einen Dialog zwischen verschiedenen Teilnehmern eines Standards zu ermöglichen. Dabei werden Nachrichten schemavalid und sicher zwischen z.B. Unternehmen und Versicherer versendet. Abhängig vom verwendeten Standard bietet DialogMessage Möglichkeiten vom Austausch einfacher Informationen bis hin zur Umsetzung einer gesicherten Messenger-Applikation.

Hierbei wird zwischen Standard-DialogMessages und freien DialogMessages unterschieden. Standard DialogMessages besitzen eine spezifizierte Struktur mit genau definierten Ids. Freie DialogMessages setzen sich beliebig aus den Schema-Elementen der DialogMessage zusammen.

Freie DialogMessages erhöhen die Flexibilität des swissdec Standards, indem sie neue Strukturen von Fragen und Antworten zwischen den beteiligten Partnern ermöglichen, ohne neue Versionen des swissdec Standards abwarten zu müssen. Falls sich ein Typ einer freien DialogMessage als allgemeiner Bedarf herausstellt, kann dieser Typ bei einer neuen Version dieses Dokuments als neue Standard DialogMessage spezifiziert werden. In einigen Standards sind freie DialogMessages aus Datenschutzgründen nicht gestattet.

Dieser Anhang enthält die Anforderungen an das generelle Vorgehen zur Darstellung von DialogMessages sowie Anforderungen an die Darstellung von vorgegebenen Standard DialogMessages.

### D.2 Die Verwendung von DialogMessages

#### D.2.1 Test und Abnahme

Transmitterseitig wird die DialogMessage im Rahmen der Synchronisierungen abgehandelt. Die richtige Darstellung, sowie die richtige Durchführung eines Dialogs werden von Swissdec geprüft.

Auf Endreceiver-Seite steht es dem Endempfänger frei, welche Form von DialogMessage er verwendet. Er kann auf Standard-Messages zurückgreifen oder eigene, freie Messages definieren. Einschränkungen für bestimmte Domänen und Standards werden separat publiziert.

Standard-Messages werden im Dialogmessage-Katalog publiziert und können vom Swissdec Infopoint (<https://infopoint.swissdec.ch>) heruntergeladen werden.

Auf Transmitter-Seite *sollen* momentan keine Standard oder freien DialogMessages verfasst werden. Der Transmitter *soll* nur jeweils auf die DialogMessages vom Endreceiver reagieren.

#### D.2.2 Darstellung der DialogMessage

Auf Transmitter-Seite **muss** ein generischer Renderer implementiert werden, welcher alle freien und Standard DialogMessages darstellen und beantworten können **muss**. Auch der Endreceiver *sollte* einen generischen Renderer implementieren, da in zukünftigen Versionen auch vom Transmitter her Mitteilungen per DialogMessage möglich werden könnten.

Anstelle des generischen Renderes *können* für Standard DialogMessages spezielle Renderer implementiert werden.

|                                       | Transmitter als Reactor |                         | Endreceiver als Actor                   |
|---------------------------------------|-------------------------|-------------------------|---|
|                                       | Darstellen              | Antworten wenn <Answer> | Verfassen und evtl. Antwort verarbeiten |
| StandardDialogMessage "SimpleMessage" | <b>muss</b>             | Nicht vorgesehen        | <i>kann</i>                             |
| Weitere DialogMessages                | <b>muss</b>             | <b>muss</b>             | <i>kann</i>                             |
| Freie DialogMessages                  | <b>muss</b>             | <b>muss</b>             | <i>kann</i>                             |

## D.2.3 Elemente der Story DialogMessage

Die Elemente (Datenstruktur) aus welchen sich eine DialogMessage zusammenstellen lässt, sind im zugehörigen Schema des entsprechenden Standards und in den entsprechenden [Richtlinien für Lohndatenverarbeitung](#) beschrieben.

Abbildung D.1. DialogMessage XML-Schema Struktur

Die GUI-Darstellung der einzelnen Elemente entspricht in etwa denjenigen eines Formulars z.B. in HTML oder einer Programmiersprache. Die Darstellung kann als einfacher Text oder aber unter Verwendung von GUI-Elementen wie Checkboxes, Dropdowns oder Kalenderkomponenten erfolgen.

Das Element «Paragraph/Label» wird sowohl als Label, als auch als alleinstehender Text verwendet. Zu beachten: Werte unter Paragraph/Answer können einen vorgegebenen Default-Wert haben, welcher dargestellt werden muss.

| Element / Attribut  | Editierbar | Darstellung | Bemerkung   |
|---------------------|------------|-------------|---|
| Creation            | Nein       | Kann        | Datum und Zeit der Erstellung der DialogMessage   |
| StoryID             | Nein       | Kann        | StoryID der DialogMessage: ID muss für den Geschäftsfall eindeutig sein. Ausserhalb eines Geschäftsfalls muss sie ebenfalls eindeutig sein.                 |
| StandardDialogID    | Nein       | Kann        | Kennzeichnung als standardisierte DialogMessage oder «notStandard»  |
| Previous            | Nein       | Kann        | RequestStoryID oder ResponseStoryID der vorangegangenen DialogMessage, je nach dem ob die DialogMessage in einem Request oder einer Response enthalten war. |
| Title               | Nein       | Muss        | Optionaler Titel der DialogMessage. Muss dargestellt werden, wenn vorhanden.  |
| Description         | Nein       | Muss        | Optionale Beschreibung der DialogMessage. Muss dargestellt werden, wenn vorhanden, z.B. als Tooltip   |
|                     |            |             |   |
| Section/sectionID   | Nein       | Kann        |   |
| Section/Heading     | Nein       | Muss        |   |
| Section/Description | Nein       | Muss        |   |
|                     |            |             |   |
| Paragraph           | Nein       | Muss        | Absatz mit Inhalt   |

| Element / Attribut   | Editierbar | Darstellung | Bemerkung  |
|--|------------|-------------|--|
| Paragraph/SectionIDRef   | Nein       | Kann        | Attribut welches die Zugehörigkeit eines Absatzes zu einer Section anzeigt |
| Paragraph/ID   | Nein       | Kann        |  |
| Paragraph/Label  | Nein       | Muss        | Text in einem Absatz, Label für Value oder Answer                          |
| Paragraph/Value/String   | Nein       | Muss        |  |
| Paragraph/Value/Integer  | Nein       | Muss        |  |
| Paragraph/Value/Double   | Nein       | Muss        |  |
| Paragraph/Value/Boolean  | Nein       | Muss        | Bei einer Gruppe von Booleans ist keine ausschliessende Auswahl möglich.   |
| Paragraph/Value/Date   | Nein       | Muss        |  |
| Paragraph/Value/DateTime   | Nein       | Muss        |  |
| Paragraph/Value/YesNoUnknown   | Nein       | Muss        |  |
| <i>Answer/&lt;Type&gt;/Default: darf vom Actor gesetzt, vom Reactor in der Antwort aber nicht verändert werden</i> |            |             |  |
| Paragraph/Answer/String/Default  | Nein       | Muss        |  |
| Paragraph/Answer/String/Value  | Muss       | Muss        |  |
| Paragraph/Answer/Integer/Default   | Nein       | Muss        |  |
| Paragraph/Answer/Integer/Value   | Muss       | Muss        |  |
| Paragraph/Answer/Double/Default  | Nein       | Muss        |  |
| Paragraph/Answer/Double/Value  | Muss       | Muss        |  |
| Paragraph/Answer/Boolean/Default   | Nein       | Muss        | Bei einer Gruppe von Booleans ist keine ausschliessende Auswahl möglich.   |
| Paragraph/Answer/Boolean/Value   | Muss       | Muss        | Bei einer Gruppe von Booleans ist keine ausschliessende Auswahl möglich.   |
| Paragraph/Answer/Date/Default  | Nein       | Muss        | z.B. Kalender mit Textfeld   |
| Paragraph/Answer/Date/Value  | Muss       | Muss        | z.B. Kalender mit Textfeld   |
| Paragraph/Answer/DateTime/Default  | Nein       | Muss        | z.B. Kalender mit Textfeld   |
| Paragraph/Answer/DateTime/Value  | Muss       | Muss        | z.B. Kalender mit Textfeld   |

| Element / Attribut                     | Editierbar | Darstellung | Bemerkung     |
|--|------------|-------------|---------------|
| Paragraph/Answer/Yes-NoUnknown/Default | Nein       | Muss        | z.B. DropDown |
| Paragraph/Answer/Yes-NoUnknown/Value   | Muss       | Muss        | z.B. DropDown |

## D.2.4 Abfolge bei der Darstellung von DialogMessages

Das Aktivitätsdiagramm beschreibt das generische Vorgehen beim Empfang einer Response oder eines Requests mit enthaltenen DialogMessages. Die Antwort auf eine DialogMessage erfolgt, sofern notwendig, zu einem späteren Zeitpunkt in einem neuen Request-Response-Zyklus.

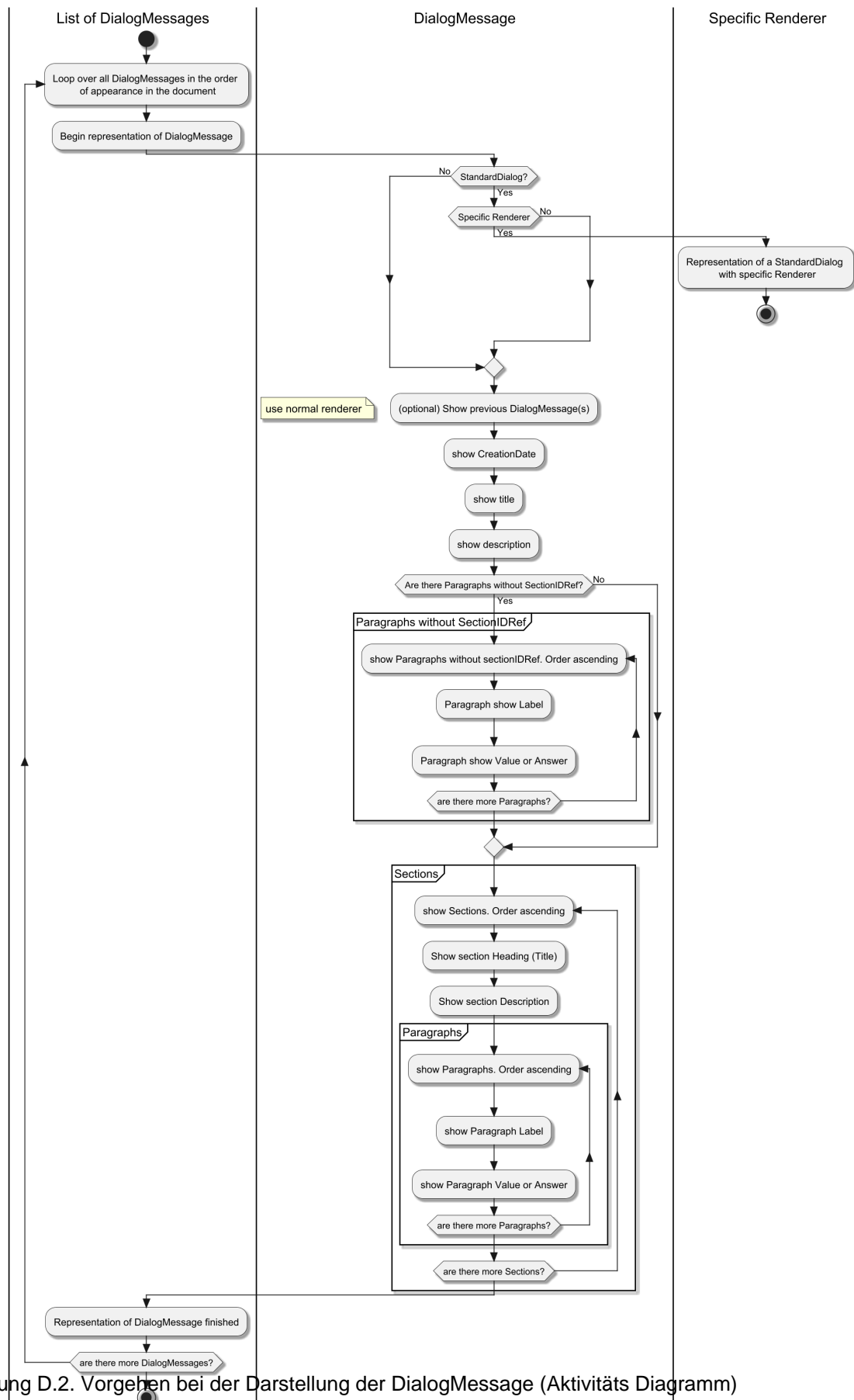


Abbildung D.2. Vorgehen bei der Darstellung der DialogMessage (Aktivitäts Diagramm)



## D.2.5 Antwort als Reactor mittels einer DialogMessage

Bei der Antwort eines Reactors auf eine DialogMessage, wird zwischen folgenden Situationen unterschieden.

1. Es wird eine DialogMessage-Story mit «Paragraph/Answer»-Elementen empfangen. Die DialogMessage muss dargestellt werden und es muss eine Antwort mit der gleichen Struktur gesendet werden. Die Antwort wird wahrscheinlich in einem späteren Request-Response Zyklus geschickt.
2. Es wird eine DialogMessage-Story ohne «Paragraph/Answer»-Elemente empfangen. Die DialogMessage muss dargestellt werden. Auf diese DialogMessage muss nicht geantwortet werden. Es muss keine Datenstruktur kopiert werden.
3. Es wird eine DialogMessage empfangen, welche sich auf eine beliebige, vorangegangene Story bezieht. Der Bezug wird durch das Element «Previous» hergestellt, s. Verkettung.

### D.2.5.1 Antwort auf eine DialogMessage mit enthaltenen Paragraph/Answer-Elementen

Abbildung D.3. Paragraph/Answer/DateTime-Element, andere Answer/<Type>-Elemente entsprechend

Die Antwort auf eine DialogMessage mit enthaltenen «Paragraph/Answer»-Elementen ist wiederum eine DialogMessage mit «Paragraph/Answer»-Elementen. Die Antwort wird wahrscheinlich in einem späteren Request-Response Zyklus erfolgen.

In der Antwort-DialogMessage muss das Element «Previous» verwendet werden, so dass sowohl für den Actor, wie auch für den Reactor nachvollziehbar ist, auf welche DialogMessage-Story sich die Antwort bezieht. Im Transmitter wird immer das Element «Previous/ResponseStoryID» verwendet, da die anfragende DialogMessage vom Endreceiver immer in einer SynchronicIncidentResponse enthalten war. Im Endreceiver wird immer «Previous/RequestStoryID» verwendet, da die anfragende DialogMessage vom Transmitter immer in einem SynchronicIncidentRequest enthalten war.

Bis auf «Creation», «StoryID» und «Previous» wird die gesamte DialogMessage mit enthaltenen «Paragraph/Answer»-Elementen gespiegelt.

Die durch den Dialogteilnehmer geänderten Werte, ansonsten die Default-Werte, müssen bei der Antwort immer in die Paragraph/Answer/<Type>/Value Elemente eingetragen werden, nicht in die «Paragraph/Answer/<Type>/Default»-Elemente. Dabei wird <Type> durch den jeweils übergebenen Werte-Typ ersetzt, z.B. «Paragraph/Answer/Boolean/Value».

«Answer»-Elemente, welche nur optional ausgefüllt werden müssen, sind durch das Attribut «optional» gekennzeichnet. Jedes andere «Paragraph/Answer/<Type>/Value»-Element muss bei der Antwort gesetzt sein. Das bedeutet, dass alle mit «Paragraph/Answer»-Elementen angefragten Werte ohne das Attribut «optional» Pflichtfelder sind, auch wenn diese nach XML-Schema leer zurückgesandt werden könnten.

### D.2.5.2 Verkettung von DialogMessages

Die verschiedenen DialogMessages können miteinander verkettet werden. Sind in einer anfragenden DialogMessage «Answer»-Elemente enthalten, muss die Antwort-DialogMessage mit der anfragenden DialogMessage durch das Element «Previous» verkettet werden. Damit lässt sich in der Visualisierung eine ganze Kette von DialogMessages darstellen. In dieser Sicht werden einzelne DialogMessages für den Benutzer verständlicher, da diese in einem Ablaufkontext stehen und dann betrachtet werden können.

Die Verkettung wird mit dem Element «DialogMessage/Previous» gesteuert.

Abbildung D.4. Struktur des DialogMessage/Previous-Elementes

1. Ein Actor initiiert im ersten Schritt einen Dialog und kann optional im Element «Previous» den Link zu einer bestehenden StoryID setzen. Dabei wird unterschieden zwischen einer beliebigen bestehenden Response- oder RequestStoryID.
2. Ein Reactor muss bei einer Antwort auf eine Actor DialogMessage immer ein «Previous»-Element setzen und zwar die StoryID der zuvor erhaltenen Actor DialogMessage.

### D.3 Standard DialogMessages

Die im DialogMessage-Katalog beschriebenen DialogMessages sind standardisiert und müssen im Rahmen einer Swissdec Zertifizierung nach Tabelle 3 Anforderungen zur Umsetzung von DialogMessages erzeugt und dargestellt werden können. Die definierten Strukturen müssen eingehalten werden.

Eine DialogMessage muss die StandardDialogID ihres Typs im Instanzdokument setzen, s. nachfolgende Tabelle.

| DialogMessage       | StandardDialogID   |
|---------------------|--------------------|
| SimpleMessage       | 0000.0001.0001-001 |
| TaskWithDeadline    | 0000.0001.0001-002 |
| Freie DialogMessage | notStandard        |

Die letzten drei Ziffern sind die Minor-Version der in diesem Dokument spezifizierten StandardDialogMessages. Sollte eine spezifizierte DialogMessage Änderungen erfahren, wird ihre Minor-Version erhöht.

Zu allen spezifizierten Standard DialogMessages gibt es mitgeltende Beispiele in Form von Instanzdokumenten im DialogMessage-Katalog. (<https://infopoint.swissdec.ch>).

## E Detailspezifikation Completion URL

### E.1 Aufbau und Verarbeitung der Completion URL

#### E.1.1 Parsen der Basis-URL

Die Completion-URL führt auf eine Webseite mit Login-Maske. Key und Passwort aus dem XML-Dokument **müssen** als Parameter an die URL angehängt werden. Dies ermöglicht eine Vorbefüllung der Login-Felder. Der Benutzer muss somit Key und Passwort nicht von Hand eingeben.

Die in der Story mitgelieferte Completion-URL darf bereits Parameter wie z.B. Sprachangaben enthalten. Wenn dies der Fall ist, werden Key und Passwort jeweils mit einem „&“ angefügt. Handelt es sich beim Key jedoch um den ersten Parameter der URL, wird dieser mit einem „?“ angefügt.

**Beispiel 1:** `https://www.institutionA.ch`

**Beispiel 2:** `https://www.institutionA.ch?language=fr`

Das System muss dies erkennen können und die URL richtig ergänzen.

**Beispiel 1:** `https://www.institutionA.ch?key=u1&password=cxsy23450dl`

**Beispiel 2:** `https://www.institutionA.ch?language=fr&key=u1&password=cxsy23450dl`

Für den Fall, dass es empfängerseitig Probleme mit der Vorbefüllung der Felder gibt, müssen Key und Passwort dem Benutzer auf der Benutzeroberfläche eindeutig dargestellt werden, sodass er sie notfalls durch Copy-Paste in die Felder der Completion-Maske einfügen kann.

#### E.1.2 Kodierung von URL und Parametern

Bei der Übermittlung von Completion-Informationen **muss** beachtet werden, dass Key, Passwort und URL Sonderzeichen enthalten können, welche im XML oder in der URL nicht dargestellt werden können/dürfen. Die Codierung dieser Sonderzeichen muss korrekt umgesetzt werden, damit die Completion reibungslos funktioniert.

**Beispiel** Completion-Informationen beim Empfänger:

- Completion-URL: `https://institutionA.ch?parameter1=test&parameter2=info`
- Key: `u1#`
- Passwort: `cxsy2%#@=30#dlü`

Diese Completion-URL, sowie Key und Passwort werden in der XML-Response zurückgegeben. Dies kann bedeuten, dass einige Sonderzeichen, welche im XML nicht verwendet werden dürfen, mit Character Entity References codiert werden (z.B. `&amp;` anstelle von „&“ oder `&lt;` anstelle von „<“). Es ist darauf zu achten, dass diese Character Entity References im Sendersystem dem Benutzer wieder in entschlüsselter Form dargestellt werden, und vor allem, dass die Sonderzeichen dem Browser auf korrekte Weise übergeben werden, damit sie richtig interpretiert werden können:

#### **Beispiel**

Completion-URL in der XML-Response:

`https://www.completion-url.ch/?parameter1=test&amp;parameter2=info`

Muss dargestellt/verarbeitet werden als:

`https://www.completion-url.ch/?parameter1=test&parameter2=info`

In einer zweiten Etappe geht es darum, dass Key und Passwort der URL als Parameter angefügt werden, um auf diese Weise die Vorbefüllung der Felder im Completion-Login zu erhalten. Hier ist darauf zu achten, dass

<Key> und <Password> im Element <Credentials> vor dem Anhängen an die Completion-URL vom Transmitter in eine URL-kodierte Form gebracht werden müssen, damit eventuell darin enthaltene Sonderzeichen vom Browser richtig interpretiert werden können.

Dem Endbenutzer dürfen Key und Passwort allerdings nicht in URL-kodierter Form dargestellt werden, sondern exakt so, wie sie in der XML-Quittung enthalten sind. Dies ist notwendig, da die Web-Eingabeformulare keine URL-kodierte Form verlangen und der Benutzer auf diese Weise mit direktem Copy-Paste arbeiten kann.

### Beispiel

Darstellung in Sendersystem:

- Completion-URL:: `https://www.completion-url.ch/?parameter1=test&parameter2=info`
- Key: `u1#`
- Passwort: `cxsy2%@=30#dlü`

Zusammengefügte Completion-URL für den Aufruf der Completion mit vorbefüllten Feldern für Key und Passwort mit URL-kodierten Parametern:

`https://www.institutiona.ch/?key=u1%23&password=cxsy2%25%40%3d30%23d1%c3%bc`

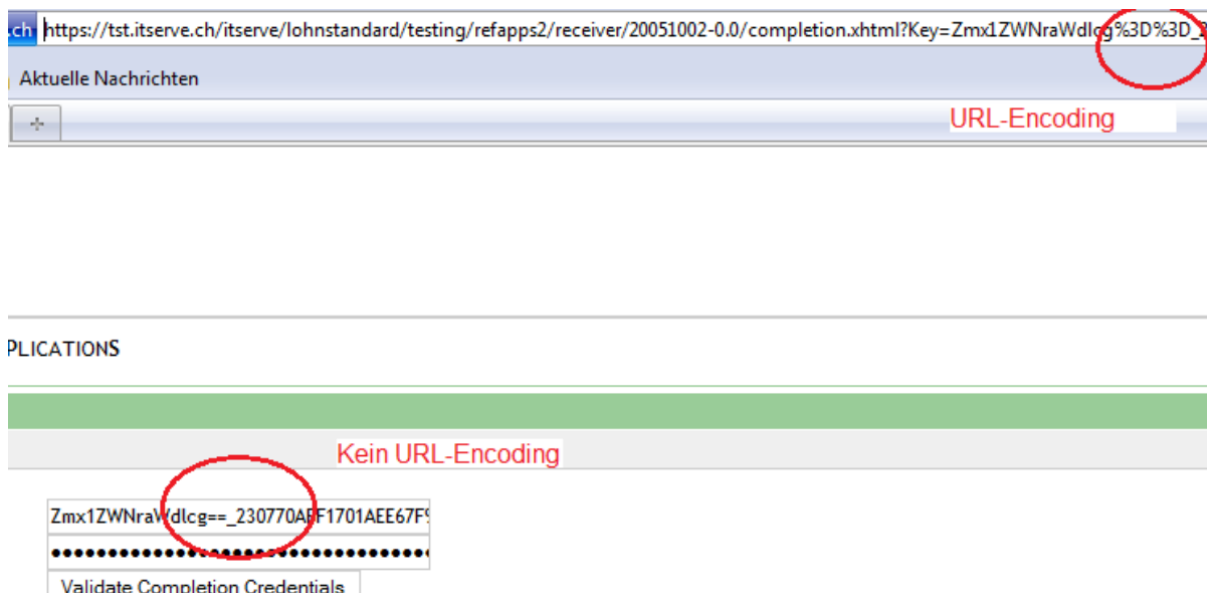


Abbildung E.1. URL-Encoding in der Completion

### Gesamtsituation

Empfänger sendet in der XML-Response:

Completion-URL (mit Character Entity Reference, mit URL-Encoding)

`https://www.completion-url.ch/?parameter1=test&parameter2=info%23`

Key und Passwort (keine Character Entity Reference, kein URL-Encoding)

`u1#; cxsy2%@=30#dlü`

Darstellung der Response im Sendersystem:

Completion-URL (keine Character Entity Reference, mit URL-Encoding)

`https://www.completion-url.ch/?parameter1=test&parameter2=info%23`

Key und Passwort (keine Character Entity Reference, kein URL-Encoding)

`ul#; cxsy2%@=30#dlü`

Key und Passwort werden hier im Original dargestellt, damit sie notfalls per Copy-Paste ins Login-WebFormular eingefügt werden können.

Aufruf der Completion im Browser:

Completion-URL mit angehängten Parametern (keine Character Entity Reference, mit URL-Encoding)

`http://www.institutionA.ch?key=ul%23&password=cxsy2%25%40%3d30%23d1%c3%bc`

Key und Passwort werden hier ebenfalls mit URL-Encoding angehängt!

Die meisten Entwicklungsplattformen haben in ihren Bibliotheken Tools zur URL-Kodierung. Die URL Kodierung wird in [\[RFC3986\]](#) beschrieben.

Aufbau einer URL gemäss[\[RFC3986\]](#):

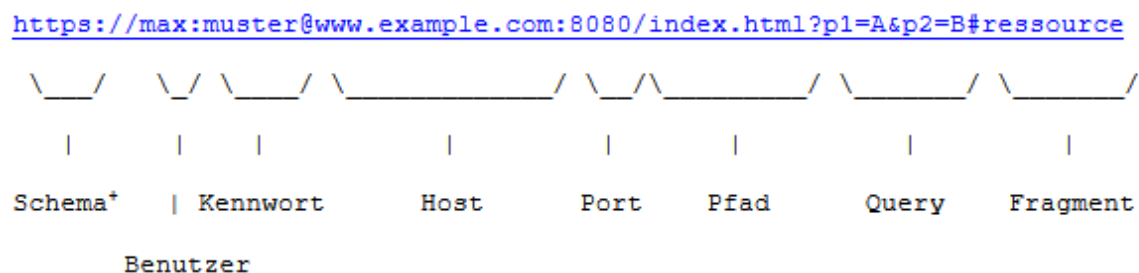


Abbildung E.2. Aufbau URL gemäss RFC3986

## F Bausteine

### F.1 AB-01 Konfiguration

| AB-01: Konfiguration      |  |
|---------------------------|--|
| Fachliche Anforderung     | <p>Das Sendersystem ist für die korrekte Adressierung des Distributors verantwortlich. Dazu <b>muss</b> das Empfängersystem via korrekter URL angesprochen werden. Aus Sicherheitsgründen <i>sollte</i> die Adressierung vom Endbenutzer nicht beliebig verändert werden können.</p> <p>Der Distributor adressiert korrekt die Empfängersysteme und verfügt über die nötigen Zertifikate für die sichere Kommunikation mit diesen.</p> |
| Abhängige Bausteine       | keine  |
| Ausschliessende Bausteine | keine  |
| Anforderungen an Akteure  | 1. Konfiguration des Empfängersystems via URL  |
| Ergebnis                  | Die korrekte Adressierung zwischen Sender und Empfänger ist gewährleistet.   |
| Fehlerbehandlung          | Abhängig von den beteiligten Systemen.   |
| Technische Implikationen  | keine  |

Tabelle F.1. AB-01 Konfiguration

### F.2 AB-02 Erreichbarkeit

| AB-02: Erreichbarkeit     |  |
|---------------------------|--|
| Fachliche Anforderung     | Die Erreichbarkeit des Endempfängers muss geprüft werden. Dazu wird eine einfache Anfrage vom Distributor an den Receiver geschickt. Die Rückantwort des Receivers an den Distributor bestätigt die Erreichbarkeit.  |
| Abhängige Bausteine       | keine  |
| Ausschliessende Bausteine | keine  |
| Anforderungen an Akteure  | <ol style="list-style-type: none"> <li>1. Der Distributor löst die Überprüfung aus.</li> <li>2. Der Distributor sendet eine einfache Serveranfrage (Ping) an die Zieladresse des Receivers.</li> <li>3. Der Distributor wertet die Rückantwort des Receivers aus.</li> </ol> |
| Ergebnis                  | Die Rückantwort des Receivers enthält einen Timestamp mit der Systemzeit des Receivers oder eine Fehlermeldung.  |
| Fehlerbehandlung          | 1. Der Receiver ist nicht erreichbar. Fehlermeldung.   |
| Technische Implikationen  | keine  |

Tabelle F.2. AB-02 Erreichbarkeit

### F.3 AB-03 Interoperabilität

| AB-03: Interoperabilität |  |
|--------------------------|--|
| Fachliche Anforderung    | Die Kommunikation zwischen Sendersystem und Distributor muss geprüft werden. Zu diesem Zweck werden zwei Operanden und ein Umlautstring an den |

| <b>AB-03: Interoperabilität</b> |   |
|---------------------------------|---|
|                                 | Distributor gesendet. Die Rückantwort des Distributors ermöglicht es, Probleme mit Encoding oder Byte-Order zu erkennen.  |
| Abhängige Bausteine             | keine   |
| Ausschliessende Bausteine       | keine   |
| Anforderungen an Akteure        | <ol style="list-style-type: none"> <li>1. Der Akteur löst die Überprüfung aus.</li> <li>2. Das Sendersystem sendet eine Interoperabilitätsanfrage (2 Operanden, Umlautstring) an die Zieladresse des Distributors.</li> <li>3. Das Sendersystem wertet die Rückantwort des Distributors aus.</li> </ol> |
| Ergebnis                        | Die Rückantwort des Distributors enthält Informationen zur distributorseitigen Interoperabilität sowie Ergebnisse, welche transmitterseitig verifiziert werden müssen.  |
| Fehlerbehandlung                | <ol style="list-style-type: none"> <li>1. Distributor nicht erreichbar. Fehlermeldung.</li> <li>2. Fehler in der Interoperabilität. Der Fehler muss dem Benutzer klar verständlich angezeigt werden.</li> </ol>   |
| Technische Implikationen        | keine   |

Tabelle F.3. AB-03 Interoperabilität

## F.4 AB-04 Anmeldung

| <b>AB-04: Anmeldung</b>   |   |
|---------------------------|---|
| Fachliche Anforderung     | Ein Sendersystem muss sich bei einem Empfängersystem für einen späteren Übermittlungsprozess anmelden können. Mit diesem Anmeldeprozess wird der zukünftige Informationsaustausch über den Distributor zwischen den beiden Beteiligten in die Wege geleitet.  |
| Abhängige Bausteine       | keine   |
| Ausschliessende Bausteine | keine   |
| Anforderungen an Akteure  | <ol style="list-style-type: none"> <li>1. Der Akteur stellt die benötigten Informationen für eine Anmeldung bereit.</li> <li>2. Das Sendersystem adressiert den gewünschten Empfänger korrekt.</li> <li>3. Das Sendersystem übermittelt die Anmeldung an den Distributor.</li> <li>4. Die Antwort vom Distributor wird ausgewertet und dem Benutzer angezeigt.</li> </ol> |
| Ergebnis                  | Das Sendersystem konnte sich bei einem Empfänger für einen späteren Übermittlungsprozess anmelden.  |
| Fehlerbehandlung          | <ol style="list-style-type: none"> <li>1. Der Empfänger ist nicht erreichbar. Fehlermeldung.</li> <li>2. Die Anmeldung verläuft nicht erfolgreich. Die Fehlermeldung vom Empfänger/Distributor wird ausgegeben.</li> </ol>  |
| Technische Implikationen  | keine   |

Tabelle F.4. AB-04 Anmeldung

## F.5 AB-05 Explizite Adressierung (direkte Adressierung)

| <b>AB-05: Explizite Adressierung eines Empfängers (direkte Adressierung)</b> |   |
|--|---|
| Fachliche Anforderung  | Rahmenbedingung aus <a href="#">[[UpgrDistri_Anforderungen]]</a> : Die Lohndaten werden sicher und direkt an die vom Kunden (Unternehmen) adressierten Endempfänger |

| <b>AB-05: Explizite Adressierung eines Empfängers (direkte Adressierung)</b> |   |
|--|---|
|  | ger übermittelt. Zusätzliche Intermediäre dürfen aus Datenschutzgründen nicht beliefert werden.   |
| Abhängige Bausteine  | keine   |
| Ausschliessende Bausteine  | keine   |
| Anforderungen an Akteure   | <ol style="list-style-type: none"> <li>1. Der Benutzer definiert die Adressaten</li> <li>2. Der Distributor leitet die Meldung allen Adressaten weiter</li> <li>3. Die Adressaten prüfen, ob Sie die Daten erhalten dürfen (richtige Adressierung)</li> <li>4. Der Benutzer erhält eine Quittung / Fehler pro Adressat</li> </ol> |
| Ergebnis   | Alle Adressaten wurden bedient, der Benutzer hat pro Adressat eine Quittung/Fehler  |
| Fehlerbehandlung   | 1. Der Receiver ist nicht erreichbar. Fehlermeldung.  |
| Technische Implikationen   | <p>Abbildung F.1. AddresseeType</p> <ul style="list-style-type: none"> <li>• Definiertes Feld mit Identifier des Adressaten in den Daten (zum Beispiel InstitutionID, Canton, ...)</li> <li>• Antwortstruktur mit Quittung/Fehler pro Adressat</li> </ul>   |

Tabelle F.5. AB-05 Explizite Adressierung (direkte Adressierung)

## F.6 AB-06 Implizite Adressierung (Gruppenadressierung)

| <b>AB-06 Implizite Adressierung (Gruppenadressierung)</b> |  |
|---|--|
| Fachliche Anforderung                                     | Die Lohndaten werden sicher und direkt an alle Endempfänger, die die Meldung erhalten wollen und dürfen, übermittelt. Zusätzliche Intermediäre dürfen aus Datenschutzgründen nicht beliefert werden.   |
| Abhängige Bausteine                                       | keine  |
| Ausschliessende Bausteine                                 | keine  |
| Anforderungen an Akteure                                  | <ol style="list-style-type: none"> <li>1. Der Benutzer definiert die Adressaten als Gruppe.</li> <li>2. Der Distributor filtert und verteilt die Inhalte an alle Adressaten, wobei er die Adressierung aus spezifischen Inhalten herausliest (z.B. Kanton).</li> <li>3. Die Adressaten prüfen, ob sie die Daten erhalten dürfen (richtige Adressierung)</li> <li>4. Der Benutzer erhält eine Quittung / Fehler pro Adressat</li> </ol> |
| Ergebnis  | Alle Adressaten wurden bedient, der Benutzer hat pro Adressat eine Quittung/Fehler   |
| Fehlerbehandlung  | <ol style="list-style-type: none"> <li>1. Der Distributor ist nicht erreichbar. Fehlermeldung.</li> <li>2. Ein oder mehrere Adressaten sind nicht erreichbar/unbekannt. Verifizieren der Adressierung und erneute Übermittlung.</li> </ol>   |
| Technische Implikationen                                  |  |

Tabelle F.6. AB-06 Implizite Adressierung (Gruppenadressierung)



## F.7 AB-07 Absichtserklärung

| <b>AB-07 Absichtserklärung</b> |   |
|--------------------------------|---|
| Fachliche Anforderung          | Der Absender soll bei grösseren und komplexeren Daten verständlich erklären, was für Daten an wen zu senden sind. Dabei sorgt der Distributor für eine korrekte Filterung und Verteilung der Meldung, d.h. die „identische Meldung“ kann unterschiedlich verteilt werden. Diese Verteilung mit entsprechender Filterung muss nicht durch den Absender vorgenommen bzw. entwickelt werden, sondern wird zentral gelöst.  |
| Abhängige Bausteine            | keine   |
| Ausschliessende Bausteine      | keine   |
| Anforderungen an Akteure       | <ol style="list-style-type: none"> <li>1. Der Absender kann eindeutig auswählen, welche Empfänger Daten erhalten sollen.</li> <li>2. Die Daten für die adressierten Empfänger werden vom System für den Versand aufbereitet und sind valid und plausibel.</li> <li>3. Die ausgewählten Daten werden mit entsprechender Absichtserklärung an den Distributor versendet.</li> <li>4. Für jeden adressierten Empfänger wird eine Antwort erwartet, ausgewertet und dargestellt.</li> </ol> |
| Ergebnis                       | Unabhängig von den in der XML-Datei enthaltenen Daten kann ausgewählt werden, welche Empfänger vom Distributor beliefert werden sollen.   |
| Fehlerbehandlung               | 1. Endempfänger unbekannt. Adressierung überprüfen und erneut übermitteln.  |
| Technische Implikationen       | keine   |

Tabelle F.7. AB-07 Absichtserklärung

## F.8 AB-08 Identifikation eines Geschäftsvorfalles

| <b>AB-08 Identifikation eines Geschäftsvorfalles</b> |  |
|--|--|
| Fachliche Anforderung                                | Ein realer Geschäftsvorfall soll für alle Teilnehmer eindeutig identifizierbar sein. Der Geschäftsvorfall umfasst alle Aktivitäten und den Austausch von Informationen im Rahmen des durchgeführten öffentlichen Geschäftsprozesses. Diese Identifikation wird für verschiedene Aufgaben benötigt (Support, Ersatzmeldung, allgemeine Verbindungen, Dokumentation usw.). |
| Abhängige Bausteine                                  | keine  |
| Ausschliessende Bausteine                            | keine  |
| Anforderungen an Akteure                             | <ol style="list-style-type: none"> <li>1. Der Distributor vergibt eine übergreifende ID, die allen am Prozess beteiligten Teilnehmern bekannt sein muss.</li> <li>2. Dem Empfängersystem muss die Identifikation des Geschäftsvorfalles bekannt sein und es muss damit auf den entsprechenden Fall zugreifen können.</li> </ol>  |
| Ergebnis   | Der Geschäftsvorfall wird von allen an der Übermittlung beteiligten Parteien erkannt und kann eindeutig zugeordnet werden.   |
| Fehlerbehandlung                                     | <ol style="list-style-type: none"> <li>1. Unbekannte Fallnummer. Fehlermeldung.</li> <li>2. Fallnummer nicht mehr gültig. Fehlermeldung mit Statusinformation zum Geschäftsvorfall.</li> </ol>   |
| Technische Implikationen                             | keine  |

Tabelle F.8. AB-08 Identifikation eines Geschäftsvorfalles

## F.9 AB-09 Ersatzmeldung

| <b>AB-09 Ersatzmeldung</b> |   |
|----------------------------|---|
| Fachliche Anforderung      | Eine gesendete und erfolgreich abgeschlossene Meldung enthält systematisch falsche Daten (zum Beispiel aufgrund eines technischen Fehlers). In diesem Fall kann dafür eine Ersatzmeldung übermittelt werden.  |
| Abhängige Bausteine        | AB-05, AB-06, AB-07, AB-08  |
| Ausschliessende Bausteine  | keine   |
| Anforderungen an Akteure   | <ol style="list-style-type: none"> <li>1. Die ursprünglich gesendete Meldung wurde erfolgreich abgeschlossen.</li> <li>2. Es wird eindeutig definiert, für wen die Ersatzmeldung bestimmt ist.</li> <li>3. Es ist eindeutig ersichtlich, welche Meldung ersetzt werden muss.</li> <li>4. Der Akteur löst die Ersatzmeldung aus.</li> <li>5. Das Endsystem erhält die Ersatzmeldung via Distributor.</li> <li>6. Anhand der IDs wird die ursprüngliche Meldung identifiziert und durch die Ersatzmeldung ersetzt.</li> </ol> |
| Ergebnis                   | Die ursprünglich übermittelte Meldung wurde durch die Ersatzmeldung ersetzt.  |
| Fehlerbehandlung           | <ol style="list-style-type: none"> <li>1. Die zu ersetzende Meldung wurde nicht gefunden. Fehlermeldung.</li> <li>2. Die zu ersetzende Meldung war noch nicht abgeschlossen. Fehlermeldung mit Information zum Status der offenen Meldung.</li> <li>3. Die Meldung kann nicht mehr ersetzt werden, weil sie bereits vom Empfänger bearbeitet wurde. Fehlermeldung.</li> </ol>   |
| Technische Implikationen   | DRAFT: Genauere Informationen zur Spezifikation der Ersatzmeldung folgen.   |

Tabelle F.9. AB-09 Ersatzmeldung

## F.10 AB-10 Request und Response IDs

| <b>AB-10 Request und Response IDs</b> |  |
|---------------------------------------|--|
| Fachliche Anforderung                 | Es muss zu jedem Zeitpunkt gewährleistet sein, dass die Nachvollziehbarkeit der gesamten Kommunikation sichergestellt ist. Dies wird mit Hilfe von eindeutigen IDs auf Requests und Responses sichergestellt.                          |
| Abhängige Bausteine                   | keine  |
| Ausschliessende Bausteine             | keine  |
| Anforderungen an Akteure              | <ol style="list-style-type: none"> <li>1. Jeder Request und jede Response muss eindeutig identifizierbar sein (auch bei leeren Synchronisierungen, Polling, etc.)</li> <li>2. Doppelt empfangene IDs müssen erkannt werden.</li> </ol> |
| Ergebnis                              | Jeder übermittelte Request enthält eine RequestID. Jede Response enthält eine ResponseID sowie die RequestID, auf welche sie sich bezieht.   |
| Fehlerbehandlung                      | Keine Vorgaben   |
| Technische Implikationen              | keine  |

Tabelle F.10. AB-10 Request und Response IDs

## F.11 AB-11 Testmeldung

| AB-11 Testmeldung         |   |
|---------------------------|---|
| Fachliche Anforderung     | <p>Es müssen Testdaten übermittelt werden können, die durch ein Element „Test-Case“ gekennzeichnet sind. Dabei werden folgende Ziele verfolgt:</p> <ul style="list-style-type: none"> <li>• Ermöglichung von Tests bei der Installation und Konfiguration von produktiven Anwendungen</li> <li>• Ermöglichung von Tests bei Problemen in der Produktion</li> </ul> <p>Dabei sollen Meldungen vom Unternehmen durch die gesamte Automatisierungskette der beteiligten Systeme (Anwendung, Transmitter, Distributor, Endempfänger und Anwendung) und ihrer Komponenten geschleust werden, <b>ohne einen echten Geschäftsvorfall anzustossen</b> (zum Beispiel ohne Rechnungen auszulösen).</p> <p>Es muss immer und überall (Anwendungsmasken, Output, usw.) klar ersichtlich sein, dass es sich bei dem vorliegenden Geschäftsprozess lediglich um einen Testfall handelt. Die Integrationstiefe bzw. der Prozessumfang kann meist nicht vollumfänglich abgedeckt werden. Im Vordergrund stehen dabei aber die kommunikativen Aspekte des öffentlichen Geschäftsprozesses.</p> <p>Die Verwendung dieses Falls ist auf Ausnahmefälle beschränkt. Eine Nutzung als Demonstrations- oder Entwicklungssystem ist nicht gestattet. Für diese Zwecke stehen die Referenzapplikationen (RefApps, ShowCase, usw.) zur Verfügung.</p> |
| Abhängige Bausteine       | keine   |
| Ausschliessende Bausteine | keine   |
| Anforderungen an Akteure  | <ol style="list-style-type: none"> <li>1. Ein Geschäftsvorfall kann als Testfall markiert werden.</li> <li>2. Ein Testfall muss im Laufe seines Lebenszyklus ein Testfall bleiben.</li> <li>3. Ein Geschäftsvorfall kann im Laufe seines Lebenszyklus nicht zum Testfall werden.</li> <li>4. Jede Meldung, die in Zusammenhang mit dem Geschäftsvorfall versendet wird, muss die Testfall-Markierung tragen.</li> </ol>   |
| Ergebnis                  | Ein Geschäftsvorfall kann von Anfang bis Ende als Testfall durchgeführt werden.   |
| Fehlerbehandlung          | <ol style="list-style-type: none"> <li>1. Testfall-Markierung fehlt auf einem Testfall. Warnung. Fall muss weiterhin als Testfall verarbeitet werden.</li> <li>2. Testfall-Markierung erscheint auf einem produktiven Geschäftsvorfall. Warnung. Die Meldung wird ignoriert.</li> </ol>   |
| Technische Implikationen  | keine   |

Tabelle F.11. AB-11 Testmeldung

## F.12 AB-12 Plausibilisierung

| AB-12 Plausibilisierung |   |
|-------------------------|---|
| Fachliche Anforderung   | <p>Eine Meldung durchläuft drei Prüfungsstufen (Validierung, Plausibilisierung, Akzeptierung). Werden auf einer Prüfungsstufe keine Fehler ausgegeben, wird die Meldung an die nächste Prüfungsstufe weitergegeben (inklusive allfälliger Warnungs- und Infomeldungen).</p> |

| <b>AB-12 Plausibilisierung</b> |  |
|--------------------------------|--|
|                                | <p>Damit soll die Qualität des gesamten Geschäftsprozesses erhöht werden, d.h.</p> <ul style="list-style-type: none"> <li>• mehr Automatisierung und weniger manuelle Arbeiten</li> <li>• geringere Durchlaufzeiten</li> </ul> <p>Die Prüfungsstufen befinden sich auf folgenden Systemteilen:</p> <ul style="list-style-type: none"> <li>• System Distributor<br/>Die eingehenden und ausgehenden Meldungen werden gegen das XML-Schema validiert. Dies verlangt meist einen kompletten DevOps-Lifecycle. Beim Distributor lassen sich neue Checks für den Geschäftsprozess „einfach“ einführen. Die <b>Plausibilisierungen</b> müssen daher nur hier entwickelt und installiert werden.</li> <li>• System Transmitter (Client, Requester)<br/>Die eingehenden und ausgehenden Meldungen werden gegen das XML-Schema validiert. Dies verlangt meist einen kompletten DevOps-Lifecycle.</li> <li>• Endreceiver (Responder, Server)<br/>Die eingehenden und ausgehenden Meldungen werden gegen das XML-Schema validiert. Dies verlangt meist einen kompletten DevOps-Lifecycle. Nur der Endreceiver kann die Akzeptanzstufe durchführen. Diese ist immer fachlich und wird daher hier nicht näher beschrieben.</li> </ul> |
| Abhängige Bausteine            | keine  |
| Ausschliessende Bausteine      | keine  |
| Anforderungen an Akteure       | <ol style="list-style-type: none"> <li>1. Jede Meldung muss eine Plausibilisierung durchlaufen. Diese findet auf dem Distributor statt.</li> <li>2. Der Endreceiver muss die Plausibilisierung auf Akzeptanzstufe durchführen.</li> </ol>  |
| Ergebnis                       | Das Sendersystem erhält eine Auflistung von Fehlermeldungen und Warnungen der Plausibilisierungsstufe.   |
| Fehlerbehandlung               | Entsprechend der aufgetretenen Fehler in der Plausibilisierung werden Fehler und Warnungen ausgegeben. Das Sendersystem stellt diese dem Benutzer dar.   |
| Technische Implikationen       | keine  |

Tabelle F.12. AB-12 Plausibilisierung

## F.13 AB-13 Filterung

| <b>AB-13 Filterung</b>    |   |
|---------------------------|---|
| Fachliche Anforderung     | Nicht nur Daten, sondern auch Geschäftsprozesse können „redundant“ sein. In einem Geschäftsprozess sollen Daten an verschiedene Endempfänger verteilt werden (Relation: Ein Transmitter übermittelt zu mehreren Endreceivern). Eine gleichzeitige Verteilung an mehrere Endreceiver kann dann unter Umständen eine <b>empfängergerechte Filterung der Daten erfordern</b> . |
| Abhängige Bausteine       | AB-05, AB-06, AB-07   |
| Ausschliessende Bausteine | keine   |
| Anforderungen an Akteure  | 1. Keine. Die Filterung wird vom Distributor vorgenommen.   |
| Ergebnis                  | Jeder Empfänger erhält nur die für ihn bestimmten Daten.  |
| Fehlerbehandlung          | Etwasige Probleme mit der Filterung werden vom Distributor geregelt. Siehe auch AB-05 und AB-06.  |

| <b>AB-13 Filterung</b>   |       |
|--------------------------|-------|
| Technische Implikationen | keine |

Tabelle F.13. AB-13 Filterung

## F.14 AB-14 Versionsmapping

| <b>AB-14 Versionsmapping</b> |   |
|------------------------------|---|
| Fachliche Anforderung        | Bei Swissdec müssen verschiedene Versionen eines Standards über längere Zeit „koexistieren“. Aus diesem Grund müssen unterschiedliche Versionen „gemappt“ werden. |
| Abhängige Bausteine          | keine   |
| Ausschliessende Bausteine    | keine   |
| Anforderungen an Akteure     | 1. Der Distributor ist verantwortlich für ein korrektes Mapping zwischen den derzeit unterstützten Versionen.   |
| Ergebnis                     | Erfolgreiches Mapping zwischen derzeit unterstützten Versionen.   |
| Fehlerbehandlung             | Keine seitens Endreceiver   |
| Technische Implikationen     | keine   |

Tabelle F.14. AB-14 Versionsmapping

## F.15 AB-15 Doublettenerkennung

| <b>AB-15 Doublettenerkennung</b> |  |
|----------------------------------|--|
| Fachliche Anforderung            | In der Übermittlung von Daten können unbeabsichtigt „Wiederholungen“ identischer Meldungen entstehen. Diese müssen erkannt und im Geschäftsprozess korrekt behandelt werden. |
| Abhängige Bausteine              | keine  |
| Ausschliessende Bausteine        | keine  |
| Anforderungen an Akteure         | 1. Die Doublettenerkennung wird vom Distributor durchgeführt.<br>2. Entsprechende Warnungen müssen sowohl Sender- als auch Empfängerseitig dargestellt werden.               |
| Ergebnis                         | Doubletten werden vom Distributor erkannt. Entsprechende Warnungen werden Sender und Empfänger ausgegeben.   |
| Fehlerbehandlung                 | Darstellung der entsprechenden Fehlermeldungen und Notifications.  |
| Technische Implikationen         | keine  |

Tabelle F.15. AB-15 Doublettenerkennung

## F.16 AB-16 Anonymisierung

| <b>AB-16 Anonymisierung</b> |  |
|-----------------------------|--|
| Fachliche Anforderung       | Ein Teilnehmer benötigt Datenstrukturen, die nur teilweise verwendet werden. Allerdings basieren diese Daten auf einer bestehenden Struktur, die nicht geän- |

| <b>AB-16 Anonymisierung</b> |   |
|-----------------------------|---|
|                             | dert werden kann/darf. Aus diesen Gründen müssen bestimmte Teile der Struktur anonymisiert werden.  |
| Abhängige Bausteine         | keine   |
| Ausschliessende Bausteine   | keine   |
| Anforderungen an Akteure    | 1. Eine Anonymisierung der Datenstrukturen muss gewährleistet sein, wenn ein Teilnehmer sie nicht vollständig verwenden darf.<br>2. Die Anonymisierung der Daten wird vom Distributor durchgeführt. |
| Ergebnis                    | Die Daten, auf die der Teilnehmer nicht zugreifen darf, sind anonymisiert.  |
| Fehlerbehandlung            | Fehlermeldungen müssen vom Teilnehmer dargestellt werden.   |
| Technische Implikationen    | keine   |

Tabelle F.16. AB-16 Anonymisierung

## F.17 AB-17 Meldungsfragmentierung

Meldungsfragmentierung wird zum aktuellen Zeitpunkt noch nicht umgesetzt.

## F.18 AB-18 Swissdec Unternehmensauthentifizierung (SUA)

| <b>AB-18 Swissdec Unternehmensauthentifizierung (SUA)</b> |   |
|---|---|
| Fachliche Anforderung                                     | Verschiedene Prozesse verlangen eine eindeutige Authentifizierung eines Unternehmens beim Endempfänger. Dies bedingt eine Signierung mit SUA-Zertifikat. Ein solches SUA-Zertifikat muss mittels des SUA-Prozesses angefordert werden können. |
| Abhängige Bausteine                                       | keine   |
| Ausschliessende Bausteine                                 | keine   |
| Anforderungen an Akteure                                  | Der Endreceiver muss die Funktionalitäten zu SUA unterstützen, wenn er Standards empfängt, die SUA verlangen.   |
| Ergebnis  | SUA wird vom Endsystem gemäss <a href="#">Anhang C, Detailspezifikation Swissdec Unternehmens-Authentifizierung SUA</a> umgesetzt.  |
| Fehlerbehandlung  | Fehlermeldungen im Rahmen des SUA-Prozesses müssen zurückgemeldet werden.   |
| Technische Implikationen                                  | Umsetzung gemäss <a href="#">Anhang C, Detailspezifikation Swissdec Unternehmens-Authentifizierung SUA</a> .  |

Tabelle F.17. AB-18 Swissdec Unternehmensauthentifizierung (SUA)

## F.19 SB-01 Transportsicherheit (HTTPS / TLS)

| <b>SB-01 Transportsicherheit (HTTPS / TLS)</b> |       |
|--|-------|
| Fachliche Anforderung                          | keine |
| Abhängige Bausteine                            | keine |
| Ausschliessende Bausteine                      | keine |

| <b>SB-01 Transportsicherheit (HTTPS / TLS)</b> |  |
|--|--|
| Anforderungen an Akteure                       | Der Übermittlungskanal muss verschlüsselt sein. Alle Verbindungen sind mittels TLS gesichert. Der Datenstrom zwischen allen Akteuren ist sicher. |
| Ergebnis                                       | Der Übermittlungskanal zwischen Distributor und Endsystem ist verschlüsselt.   |
| Fehlerbehandlung                               | Fehler in der Transportsicherheit: Abbruch der Übermittlung  |
| Technische Implikationen                       | Siehe <a href="#">[SECPDF]</a>   |

Tabelle F.18. SB-01 Transportsicherheit (HTTPS / TLS)

## F.20 SB-02 Transport Authentication (TLS Mutual Authentication)

| <b>SB-02 Transport Authentication (TLS Mutual Authentication)</b> |  |
|---|--|
| Fachliche Anforderung   | keine  |
| Abhängige Bausteine   | keine  |
| Ausschliessende Bausteine   | keine  |
| Anforderungen an Akteure  | Alle Teilnehmer (auch Klienten) authentifizieren sich über den Transportkanal. Wird zum Zeitpunkt des Erstellens dieses Dokumentes nur in der Verbindung vom Distributor zu den Endempfängern praktiziert. |
| Ergebnis  | TLS Mutual Authentication ist gewährleistet.   |
| Fehlerbehandlung  |  |
| Technische Implikationen  | Siehe <a href="#">[SECPDF]</a>   |

Tabelle F.19. SB-02 Transport Authentication (TLS Mutual Authentication)

## F.21 SB-03 Nutzdatenverschlüsselung (WS-Encryption)

| <b>SB-03 Nutzdatenverschlüsselung (WS-Encryption)</b> |  |
|---|--|
| Fachliche Anforderung                                 | Keine  |
| Abhängige Bausteine                                   | keine  |
| Ausschliessende Bausteine                             | keine  |
| Anforderungen an Akteure                              | Die Vertraulichkeit der Daten ist während der Übermittlung sichergestellt. |
| Ergebnis  | Sämtliche übermittelten Nutzdaten sind verschlüsselt.                      |
| Fehlerbehandlung                                      | Fehler in der Verschlüsselung: Die Übermittlung wird abgebrochen.          |
| Technische Implikationen                              | Siehe <a href="#">[SECPDF]</a>   |

Tabelle F.20. SB-03 Nutzdatenverschlüsselung (WS-Encryption)

## F.22 SB-04 Nutzdatensignatur (WS-Signature)

| <b>SB-04 Nutzdatensignatur (WS-Signature)</b> |       |
|---|-------|
| Fachliche Anforderung                         | keine |
| Abhängige Bausteine                           | keine |
| Ausschliessende Bausteine                     | keine |

| <b>SB-04 Nutzdatensignatur (WS-Signature)</b> |   |
|---|---|
| Anforderungen an Akteure                      | Die Datenintegrität sowie die Authentisierung der Quelle der Meldung sind sichergestellt. |
| Ergebnis                                      | Sämtliche übermittelten Nutzdaten sind signiert.  |
| Fehlerbehandlung                              | Fehler bei der Nachrichtensicherheitsüberprüfung. Abbruch der Übermittlung.               |
| Technische Implikationen                      | Siehe <a href="#">[[SECPDF]]</a>  |

Tabelle F.21. SB-04 Nutzdatensignatur (WS-Signature)

## F.23 SB-05 Unternehmens-Authentifizierung SUA Signatur (WS-Signatur)

| <b>SB-05 Unternehmens-Authentifizierung SUA Signatur (WS-Signatur)</b> |   |
|--|---|
| Fachliche Anforderung  | keine   |
| Abhängige Bausteine  | keine   |
| Ausschliessende Bausteine  | keine   |
| Anforderungen an Akteure   | SUA wird gemäss Spezifikation umgesetzt.  |
| Ergebnis   | Die Informationen zum SUA-Zertifikat werden zurückgeliefert.                              |
| Fehlerbehandlung   | Der Antragssteller ist unbekannt. Fehlermeldung. Es werden keine Informationen geliefert. |
| Technische Implikationen   | Siehe <a href="#">[[SECPDF]]</a>  |

Tabelle F.22. SB-05 Unternehmens-Authentifizierung SUA Signatur (WS-Signatur)

## F.24 SB-06 Nichtabstreitbarkeit

| <b>SB-06 Nichtabstreitbarkeit</b> |  |
|-----------------------------------|--|
| Fachliche Anforderung             | keine  |
| Abhängige Bausteine               | keine  |
| Ausschliessende Bausteine         | keine  |
| Anforderungen an Akteure          | Alle Meldungen sind eindeutig einem spezifischen Teilnehmer zugeordnet und können im Nachhinein nicht geleugnet werden.        |
| Ergebnis                          | Die Nichtabstreitbarkeit ist gewährleistet.  |
| Fehlerbehandlung                  |  |
| Technische Implikationen          | Siehe <a href="#">[[SECPDF]]</a> und <a href="#">Anhang C, Detailspezifikation Swissdec Unternehmens-Authentifizierung SUA</a> |

Tabelle F.23. SB-06 Nichtabstreitbarkeit

## F.25 PB-01 Synchrone Verteilung

| <b>PB-01: Synchrone Verteilung</b> |   |
|------------------------------------|---|
| Fachliche Anforderung              | Die Meldungen werden direkt über den Distributor an alle adressierten Empfänger geschickt. Die Rückantworten werden gesammelt und an den Sender als Quittung zurückgeliefert. |



| <b>PB-01: Synchroner Verteilung</b> |  |
|-------------------------------------|--|
| Abhängige Bausteine                 | Alle Architekturbausteine und Securitybausteine  |
| Ausschliessende Bausteine           | keine  |
| Anforderungen an Akteure            | keine  |
| Ergebnis                            | Die Meldungen wurden verteilt, die Antworten zurückgeliefert und alle fachlichen Daten auf dem Distributor gelöscht. |
| Fehlerbehandlung                    |  |
| Technische Implikationen            | keine  |

Tabelle F.24. PB-01 Synchroner Verteilung

## F.26 PB-02 Asynchroner Verteilung

| <b>PB-02: Asynchroner Verteilung</b> |   |
|--------------------------------------|---|
| Fachliche Anforderung                | Die Meldungen werden direkt vom Distributor an alle adressierten Empfänger geschickt. Die Rückantworten werden gesammelt und der Sender kann diese mit dem JobKey im GetStatus abholen. |
| Abhängige Bausteine                  | Alle Konstruktionsbausteine und Securitybausteine   |
| Ausschliessende Bausteine            | keine   |
| Anforderungen an Akteure             | keine   |
| Ergebnis                             | Die Meldungen wurden verteilt, die Antworten zurückgeliefert und alle fachlichen Daten gelöscht.  |
| Fehlerbehandlung                     |   |
| Technische Implikationen             | keine   |

Tabelle F.25. PB-02 Asynchroner Verteilung

## F.27 PB-03 Synchronisierung

| <b>PB-03: Synchronisierung</b> |   |
|--------------------------------|---|
| Fachliche Anforderung          | Die am Geschäftsprozess beteiligten Systeme müssen über gemeinsame Informationen verfügen, damit sie korrekt zusammenarbeiten können. Dies wird durch eine Synchronisierung von Daten (Story) aufgebaut und regelmässig aktualisiert. |
| Abhängige Bausteine            |   |
| Ausschliessende Bausteine      |   |
| Anforderungen an Akteure       | Die Akteure müssen die Fähigkeit besitzen, ihre Ereignisse (Stories) bereitzustellen (gemäss Spezifikation in Umfang und Aktualität).   |
| Ergebnis                       | Alle Akteure besitzen genügend Informationen, um damit ihren korrekten Teil am Geschäftsprozess zu erfüllen.  |
| Fehlerbehandlung               | Im Fall von Verlusten oder Unstimmigkeiten sollte eine Wiederherstellung möglich sein.  |
| Technische Implikationen       | Es sollte eine einfache Art von „Story-Confirmation“ ermöglicht werden.   |

| <b>PB-03: Synchronisierung</b> |   |
|--------------------------------|---|
|                                | <p>Zusätzlich muss zwischen den technischen und fachlichen Quittungen unterschieden werden.</p> <p>Der Datenaustausch zwischen Unternehmen und Versicherer erfolgt in strukturierter Form während des gesamten Fallverlaufs, bei dem beide Seiten den gleichen Informationsstand zum Ereignis aufrechterhalten.</p> |

Tabelle F.26. PB-03 Synchronisierung

## F.28 PB-04 Dialogisierung

| <b>PB-04: Dialogisierung</b> |   |
|------------------------------|---|
| Fachliche Anforderung        | <p>Ein einfacher und dynamischer Austausch von Informationen sollte jederzeit möglich sein, d.h. grössere IT-Unterstützung ist dann nicht erforderlich (z.B. eine Art „Chat“ mit optionalen Strukturen). Die Benutzung ist mindestens manuell durch den Endbenutzer sicherzustellen. Es kann aber durch eine offizielle Publikation der Definition später auch automatisiert werden.</p> <p>Weitere Details finden sich bei der Swissdec DialogMessage.</p> |
| Abhängige Bausteine          |   |
| Ausschliessende Bausteine    |   |
| Anforderungen an Akteure     | Innerhalb der Automatisierung muss bei neuen DialogMessages auch eine manuelle Verarbeitung möglich sein.   |
| Ergebnis                     | Alle Akteure besitzen auch kurzfristig und dynamisch über genügend Informationen, um damit ihren korrekten Teil am Geschäftsprozess zu leisten.   |
| Fehlerbehandlung             |   |
| Technische Implikationen     | Eine sehr dynamische Lösung muss mit entsprechender Interoperabilität, Sicherheit und Datenschutz betrieben werden.   |

Tabelle F.27. PB-04 Dialogisierung

## F.29 PB-05 Completion

| <b>PB-05: Completion</b>  |  |
|---------------------------|--|
| Fachliche Anforderung     | <p>Es wird ein dynamischer Datenaustausch verlangt, d.h. es sollte ohne Anpassungen der Anwendungen und des Transports in eine separate Webapplication verzweigt werden. Hier kann der Benutzer manuell alle Eingaben und Kontrollen durchführen. Es stehen alle gängigen Browserfunktionen zur Verfügung.</p>   |
| Abhängige Bausteine       |  |
| Ausschliessende Bausteine |  |
| Anforderungen an Akteure  | <p>Einerseits muss dazu eine Webapplication zur Verfügung stehen. Diese Webapplication muss mit dem aktuellen Protokoll verbunden sein, sofern eine Prozessabhängigkeit besteht. Die aktuelle Anwendung verliert sonst den Ablauf. Andererseits muss ein entsprechender Browser mit URL automatisch gestartet werden. Die Sicherheit (Pfad, login usw.) muss dabei immer gewährleistet sein.</p> <p>Weitere Details finden sich bei der Swissdec ELM Completion.</p> |
| Ergebnis                  |  |
| Fehlerbehandlung          |  |

| <b>PB-05: Completion</b> |   |
|--------------------------|---|
| Technische Implikationen | Die Erfahrungen aus der ELM Completion übernehmen. Es ist aber auch möglich einen Link zu einem weiteren System zu vermitteln (ohne Browser; siehe auch KLE Cross Channel Link) |

Tabelle F.28. PB-05 Completion

### F.30 PB-06 Prozessnavigation

| <b>PB-06: Prozessnavigation</b> |  |
|---------------------------------|--|
| Fachliche Anforderung           | In komplexen Abläufen ist auf der Ebene Geschäftsprozess oft eine Navigation wichtig. Es gibt vom Receiver eine fallbezogene, dynamische Steuerung. Der Transmitter bzw. der Endbenutzer muss darüber informiert werden. |
| Abhängige Bausteine             | Es besteht evtl. eine semantische Abhängigkeit oder Überlappung mit den möglichen <b>Protokoll-Zuständen</b> .   |
| Ausschliessende Bausteine       |  |
| Anforderungen an Akteure        | Diese Navigation muss geführt und übermittelt werden.  |
| Ergebnis                        | Alle Teilnehmer können sich dadurch orientieren und gewinnen damit mehr Verständnis, Gewissheit und Vertrauen.   |
| Fehlerbehandlung                |  |
| Technische Implikationen        | Sofern keine Historisierung des Pfades <sup>a</sup> notwendig ist, sollte keine Story verwendet werden.  |

<sup>a</sup>alle zeitlichen Zustände oder Orte aufzeichnen

Tabelle F.29. PB-06 Prozessnavigation

### F.31 PB-07 Digitalisierungsbereich

| <b>PB-07: Digitalisierungsbereich</b> |   |
|---------------------------------------|---|
| Fachliche Anforderung                 | Grundsätzlich sollte der gesamte Prozess digital erledigt werden. Der Digitalisierungsbereich soll aber die Kommunikation bzw. den Prozessablauf von ungleichen Teilnehmern akzeptabel steuern (Usability). Ein Digitalisierungsbereich erlaubt die internen Prozesse beim Teilnehmer schrittweise anzupassen. Damit wird die notwendige Tiefe einer Integration in die Quellsysteme flexibler. |
| Abhängige Bausteine                   |   |
| Ausschliessende Bausteine             |   |
| Anforderungen an Akteure              | Jeder muss die unterschiedlichen Digitalisierungsbereiche kontrollieren und steuern. Dies erhöht die Komplexität im Prozess und Protokoll erheblich.  |
| Ergebnis                              | Unterschiedliche Automatisierungen der Teilnehmer sind erlaubt und ein Standard kann sich unter Umständen schneller durchsetzen (z.B. nur bei grossen Mengen ist der volle Bereich sinnvoll und bei kleinen Mengen wird teilweise manuell gearbeitet).  |
| Fehlerbehandlung                      |   |
| Technische Implikationen              | Sollten beide Seiten Digitalisierungsbereiche verlangen, dann kann es sehr schnell überaus komplex werden und am Ende versteht niemand mehr diesen Prozess.   |

Tabelle F.30. PB-07 Digitalisierungsbereich

## F.32 PB-08 Datenanfordern

| PB-08: Datenanfordern     |  |
|---------------------------|--|
| Fachliche Anforderung     | Aus Gründen des Datenschutzes sollten in einem Prozess nur die dafür notwendigen Informationen ausgetauscht werden. Oft ist dies bei dynamischem Verlauf am Anfang nicht bekannt. Aus diesem Grund können weitere definierte Daten zu einem späteren Zeitpunkt verlangt werden. Für jede verlangte Story lässt sich optional noch ein Fristdatum setzen. |
| Abhängige Bausteine       |  |
| Ausschliessende Bausteine |  |
| Anforderungen an Akteure  | Einerseits müssen diese Stories dynamisch angefordert werden. Andererseits müssen diese angeforderten Daten aufbereitet und als Stories übermittelt werden.  |
| Ergebnis                  |  |
| Fehlerbehandlung          |  |
| Technische Implikationen  | Die Synchronisation muss einerseits die angeforderten X-Stories definieren. Andererseits muss mit einer speziellen AwaitStory/X-Story diese definiert werden, damit diese dann angefordert werden kann.  |

Tabelle F.31. PB-08 Datenanfordern

## G Technische Dokumentationen

### G.1 Schema Dokumentation IncidentDeclarationConsumerServiceTypes.xsd

|                     |   |
|---------------------|---|
| Target Namespace    | urn:ch:swissdec:kle:v2:20260306:incidentdeclaration:consumer:service:types  |
| Declared Namespaces | <ul style="list-style-type: none"> <li>c : urn:ch:swissdec:common:v3:20260306</li> <li>idcc : urn:ch:swissdec:kle:v2:20260306:incidentdeclaration:consumer:container</li> <li>ep : urn:ch:swissdec:basis:v1:20260306:components</li> <li>xs : http://www.w3.org/2001/XMLSchema</li> <li>idcst : urn:ch:swissdec:kle:v2:20260306:incidentdeclaration:consumer:service:types</li> </ul> |
| Version             | 0.0   |

### G.2 Schema Dokumentation IncidentDeclarationConsumerContainer.xsd

|                     |  |
|---------------------|--|
| Target Namespace    | urn:ch:swissdec:kle:v2:20260306:incidentdeclaration:consumer:container   |
| Declared Namespaces | <ul style="list-style-type: none"> <li>c : urn:ch:swissdec:common:v3:20260306</li> <li>idc : urn:ch:swissdec:kle:v2:20260306:incidentdeclaration:container</li> <li>idcc : urn:ch:swissdec:kle:v2:20260306:incidentdeclaration:consumer:container</li> <li>ep : urn:ch:swissdec:basis:v1:20260306:components</li> <li>id : urn:ch:swissdec:kle:v2:20260306:incidentdeclaration</li> <li>xs : http://www.w3.org/2001/XMLSchema</li> </ul> |
| Version             | 0.0  |

#### ComplexType: CompanyConsumerRequestType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: AddressingType“</a>  |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: AddressingType“ ] &lt;Institution&gt;„ComplexType: AddressingIdentificationType“ &lt;/Institution&gt; [1..1] [ END BASE TYPE ] &lt;WithDelegate&gt;„ComplexType: EmptyType“ &lt;/WithDelegate&gt; [0..1] &lt;/...&gt;</pre> |

#### ComplexType: DeclareIncidentConsumerRequestType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;DeclareIncidentDistributorRequestContext&gt;„ComplexType: DeclareIncidentDistributorRequestContextType“ [1..1] &lt;DeclareIncident&gt;„ComplexType: DeclareIncidentRequestType“ &lt;/DeclareIncident&gt; [1..1] &lt;/...&gt;</pre> |

#### ComplexType: DeclareIncidentConsumerResponseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;DeclareIncidentResponse&gt;„ComplexType: DeclareIncidentAddresseeSuccessJobStateType“ &lt;/DeclareIncidentResponse&gt; [1..1] &lt;/...&gt;</pre> |

#### ComplexType: DeclareIncidentDistributorRequestContextType

|          |    |
|----------|----|
| Abstract | no |
|----------|----|

|                             |   |
|-----------------------------|---|
| Parent type                 | <a href="#">„ComplexType: DistributorRequestContextType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: DistributorRequestContextType“ ] &lt;UserAgent&gt;„ComplexType: UserAgentType“&lt;/UserAgent&gt; [1..1] &lt;TransmissionDate&gt;xs:dateTime&lt;/TransmissionDate&gt; [1..1] &lt;DistributorRequestID&gt;„SimpleType: IDType“&lt;/DistributorRequestID&gt; [1..1] &lt;VersionMappingFrom&gt;„ComplexType: VersionMappingFromType“&lt;/VersionMapping\ From&gt; [0..1] &lt;ProducerSecurityTokens&gt;„ComplexType: ProducerSecurityTokensType“&lt;/ProducerSecu\ rityTokens&gt; [1..1] &lt;SignatureCertificateUID-BFS&gt;„ComplexType: SignatureCertificateUID-BFSType“&lt;/ SignatureCertificateUID-BFS&gt; [0..1] &lt;ProducerResponseNotifications&gt;„ComplexType: FeedbackNotificationsType“&lt;/Produ\ cerResponseNotifications&gt; [1..1] &lt;ConsumerNotifications&gt;„ComplexType: FeedbackNotificationsType“&lt;/ConsumerNotifi\ cations&gt; [0..1] [ END BASE TYPE ] &lt;IncidentCaseID&gt;„SimpleType: IDType“&lt;/IncidentCaseID&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: RegisterOrganizationAuthenticationAddresseeType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: SynchronizeRequestType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: SynchronizeRequestType“ ] [ BASE TYPE „ComplexType: RequestType“ ] &lt;RequestContext&gt;„ComplexType: RequestContextType“&lt;/RequestContext&gt; [1..1] [ END BASE TYPE ] &lt;Sender&gt;„ComplexType: CompanyUIDType“&lt;/Sender&gt; [1..1] [ END BASE TYPE ] &lt;Addressee&gt;„ComplexType: AddressingAddresseeType“&lt;/Addressee&gt; [1..1] &lt;Case&gt;„ComplexType: RegisterOrganizationAuthenticationAddresseeRequestCaseType“&lt;/ Case&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: RegisterOrganizationAuthenticationConsumerRequestType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;RegisterOrganizationAuthenticationDistributorRequestContext&gt;„ComplexType: RegisterOrganizationAuthen\ ticationDistributorRequestContextType“&lt;/RegisterOrganizationAuthenticationDistributorRequestContext&gt; [1..1] &lt;RegisterOrganizationAuthentication&gt;„ComplexType: RegisterOrganizationAuthenticationConsumerType“&lt;/ RegisterOrganizationAuthentication&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: RegisterOrganizationAuthenticationConsumerType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: RequestType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: RequestType“ ] &lt;RequestContext&gt;„ComplexType: RequestContextType“&lt;/RequestContext&gt; [1..1] [ END BASE TYPE ] &lt;Job&gt;„ComplexType: RegisterOrganizationAuthenticationJobType“&lt;/Job&gt; [1..1] &lt;Organization&gt;„ComplexType: CompanyConsumerRequestType“&lt;/Organization&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: RegisterOrganizationAuthenticationDistributorRequestContextType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: DistributorRequestContextType“</a>   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: DistributorRequestContextType“ ] &lt;UserAgent&gt;„ComplexType: UserAgentType“&lt;/UserAgent&gt; [1..1] &lt;TransmissionDate&gt;xs:dateTime&lt;/TransmissionDate&gt; [1..1] &lt;DistributorRequestID&gt;„SimpleType: IDType“&lt;/DistributorRequestID&gt; [1..1] &lt;VersionMappingFrom&gt;„ComplexType: VersionMappingFromType“&lt;/VersionMapping\ From&gt; [0..1] &lt;ProducerSecurityTokens&gt;„ComplexType: ProducerSecurityTokensType“&lt;/ProducerSecu\ rityTokens&gt; [1..1] &lt;SignatureCertificateUID-BFS&gt;„ComplexType: SignatureCertificateUID-BFSType“&lt;/ SignatureCertificateUID-BFS&gt; [0..1] &lt;ProducerResponseNotifications&gt;„ComplexType: FeedbackNotificationsType“&lt;/Produ\ cerResponseNotifications&gt; [1..1] </pre> |

|  |   |
|--|---|
|  | <pre> &lt;ConsumerNotifications&gt; „ComplexType: FeedbackNotificationsType“ &lt;/ConsumerNotifi\ cations&gt; [0..1] [ END BASE TYPE ] &lt;CertificateRequestID&gt; „SimpleType: IDType“ &lt;/CertificateRequestID&gt; [1..1] &lt;/...&gt; </pre> |
|--|---|

## ComplexType: RegisterOrganizationAuthenticationJobType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | Job Steuerung  |
| Technical description       | Steuerungselement für die Verarbeitung der Meldung.  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Addressee&gt; „ComplexType: AddresseeJobType“ &lt;/Addressee&gt; [1..1] &lt;TestCase&gt; „ComplexType: EmptyType“ &lt;/TestCase&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: SubscribeOrganizationConsumerRequestType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;SubscribeOrganizationDistributorRequestContext&gt; „ComplexType: SubscribeOrganizationDistributorRequestContextType“ &lt;/SubscribeOrganizationDistributorRequestContext&gt; [1..1] &lt;SubscribeOrganization&gt; „ComplexType: SubscribeOrganizationRequestType“ &lt;/SubscribeOrganization&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: SubscribeOrganizationConsumerResponseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;SubscribeOrganizationResponse&gt; „ComplexType: SubscribeOrganizationAddresseeSuccessJobStateType“ &lt;/SubscribeOrganizationResponse&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: SubscribeOrganizationDistributorRequestContextType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | „ComplexType: DistributorRequestContextType“   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: DistributorRequestContextType“ ] &lt;UserAgent&gt; „ComplexType: UserAgentType“ &lt;/UserAgent&gt; [1..1] &lt;TransmissionDate&gt;xs:dateTime&lt;/TransmissionDate&gt; [1..1] &lt;DistributorRequestID&gt; „SimpleType: IDType“ &lt;/DistributorRequestID&gt; [1..1] &lt;VersionMappingFrom&gt; „ComplexType: VersionMappingFromType“ &lt;/VersionMappingFrom&gt; [0..1] &lt;ProducerSecurityTokens&gt; „ComplexType: ProducerSecurityTokensType“ &lt;/ProducerSecurityTokens&gt; [1..1] &lt;SignatureCertificateUID-BFS&gt; „ComplexType: SignatureCertificateUID-BFSType“ &lt;/SignatureCertificateUID-BFS&gt; [0..1] &lt;ProducerResponseNotifications&gt; „ComplexType: FeedbackNotificationsType“ &lt;/ProducerResponseNotifications&gt; [1..1] &lt;ConsumerNotifications&gt; „ComplexType: FeedbackNotificationsType“ &lt;/ConsumerNotifications&gt; [0..1] [ END BASE TYPE ] &lt;SubscriptionID&gt; „SimpleType: IDType“ &lt;/SubscriptionID&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: SynchronizeDeclareIncidentConsumerRequestType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;DistributorRequestContext&gt; „ComplexType: DistributorRequestContextType“ &lt;/DistributorRequestContext&gt; [1..1] &lt;SynchronizeDeclareIncident&gt; „ComplexType: SynchronizeDeclareIncidentRequestType“ &lt;/SynchronizeDeclareIncident&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: SynchronizeDeclareIncidentConsumerResponseType

|          |    |
|----------|----|
| Abstract | no |
|----------|----|

|                             |   |
|-----------------------------|---|
| XML Instance Representation | <pre>&lt;...&gt; &lt;SynchronizeDeclareIncidentResponse&gt; „ComplexType: SynchronizeDeclareIncidentConsumerType“ &lt;/SynchronizeDeclareIncidentResponse&gt; [1..1] &lt;/...&gt;</pre> |
|-----------------------------|---|

### ComplexType: SynchronizeRegisterOrganizationAuthenticationConsumer

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | „ComplexType: SynchronizeRequestType“  |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: SynchronizeRequestType“ ] [ BASE TYPE „ComplexType: RequestType“ ] &lt;RequestContext&gt; „ComplexType: RequestContextType“ &lt;/RequestContext&gt; [1..1] [ END BASE TYPE ] &lt;Sender&gt; „ComplexType: CompanyUIDType“ &lt;/Sender&gt; [1..1] [ END BASE TYPE ] &lt;/...&gt;</pre> |

### ComplexType: SynchronizeRegisterOrganizationAuthenticationConsumerRequest-Type

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;RegisterOrganizationAuthenticationDistributorRequestContext&gt; „ComplexType: RegisterOrganizationAuthenticationDistributorRequestContextType“ &lt;/RegisterOrganizationAuthenticationDistributorRequestContext&gt; [1..1] &lt;RegisterOrganizationAuthentication&gt; „ComplexType: RegisterOrganizationAuthenticationAddresseeType“ &lt;/RegisterOrganizationAuthentication&gt; [1..1] &lt;/...&gt;</pre> |

### ComplexType: SynchronizeSubscribeOrganizationConsumerRequestType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;DistributorRequestContext&gt; „ComplexType: DistributorRequestContextType“ &lt;/DistributorRequestContext&gt; [1..1] &lt;SynchronizeSubscribeOrganization&gt; „ComplexType: SynchronizeSubscribeOrganizationRequestType“ &lt;/SynchronizeSubscribeOrganization&gt; [1..1] &lt;/...&gt;</pre> |

### ComplexType: SynchronizeSubscribeOrganizationConsumerResponseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;SynchronizeSubscribeOrganizationResponse&gt; „ComplexType: SynchronizeSubscribeOrganizationConsumerType“ &lt;/SynchronizeSubscribeOrganizationResponse&gt; [1..1] &lt;/...&gt;</pre> |

## G.3 Schema Dokumentation IncidentDeclarationContainer.xsd

|                     |   |
|---------------------|---|
| Target Namespace    | urn:ch:swissdec:kle:v2:20260306:incidentdeclaration:container   |
| Declared Namespaces | <ul style="list-style-type: none"> <li>c : urn:ch:swissdec:common:v3:20260306</li> <li>idc : urn:ch:swissdec:kle:v2:20260306:incidentdeclaration:container</li> <li>ep : urn:ch:swissdec:basis:v1:20260306:components</li> <li>id : urn:ch:swissdec:kle:v2:20260306:incidentdeclaration</li> <li>xs : http://www.w3.org/2001/XMLSchema</li> </ul> |
| Version             | 0.0   |

### ComplexType: AvailableType

|                             |                                 |
|-----------------------------|---------------------------------|
| Abstract                    | no                              |
| Technical description       | Informationen zur Verfügbarkeit |
| XML Instance Representation | <...>                           |



|  |   |
|--|---|
|  | <pre> &lt;DeclareIncident&gt; „ComplexType: DeclareIncidentAvailableType“ &lt;/DeclareIncident&gt; [0..1] &lt;/...&gt; </pre> |
|--|---|

## ComplexType: CompanyRequestType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: AddressingType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: AddressingType“ ] &lt;Institution&gt; „ComplexType: AddressingIdentificationType“ &lt;/Institution&gt; [1..1] [ END BASE TYPE ] &lt;CompanyDescription&gt; „ComplexType: CompanyDescriptionBaseType“ &lt;/CompanyDescription&gt; [1..1] &lt;Contact&gt; „ComplexType: ContactType“ &lt;/Contact&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: DeclareIncidentAddresseeContextType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: AddresseeResponseContextType“</a>  |
| Technical description       | Kontextangaben zum Adressaten  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: AddresseeResponseContextType“ ] [ BASE TYPE „ComplexType: ResponseContextType“ ] &lt;UserAgent&gt; „ComplexType: UserAgentType“ &lt;/UserAgent&gt; [1..1] &lt;InstitutionName&gt;xs:string&lt;/InstitutionName&gt; [1..1] &lt;TransmissionDate&gt;xs:dateTime&lt;/TransmissionDate&gt; [1..1] &lt;ResponseID&gt; „SimpleType: IDType“ &lt;/ResponseID&gt; [1..1] &lt;RequestID&gt; „SimpleType: IDType“ &lt;/RequestID&gt; [1..1] [ END BASE TYPE ] &lt;ProducerResponseNotifications&gt; „ComplexType: FeedbackNotificationsType“ &lt;/ProducerResponseNotifications&gt; [1..1] &lt;Warning&gt; „ComplexType: NotificationsType“ &lt;/Warning&gt; [0..1] &lt;Info&gt; „ComplexType: NotificationsType“ &lt;/Info&gt; [0..1] [ END BASE TYPE ] &lt;IncidentCaseID&gt; „SimpleType: IDType“ &lt;/IncidentCaseID&gt; [1..1] &lt;InsuranceCaseID&gt; „SimpleType: IDType“ &lt;/InsuranceCaseID&gt; [1..1] &lt;CompanyCaseID&gt; „SimpleType: IDType“ &lt;/CompanyCaseID&gt; [1..1] &lt;TestCase&gt; „ComplexType: EmptyType“ &lt;/TestCase&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: DeclareIncidentAddresseeJobStateType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: AddresseeJobType“</a>  |
| Translation                 | OpenCase Status  |
| Short description           | Initialer Status des Falls.  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: AddresseeJobType“ ] [ BASE TYPE „ComplexType: AddresseeType“ ] &lt;AddresseeIdentification&gt; „SimpleType: IDType“ &lt;/AddresseeIdentification&gt; [1..1] [ END BASE TYPE ] &lt;ProcessByDistributor&gt; „SimpleType: SimpleBooleanType“ &lt;/ProcessByDistributor&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;Success&gt; „ComplexType: DeclareIncidentAddresseeSuccessJobStateType“ &lt;/Success&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |

## ComplexType: DeclareIncidentAddresseeSuccessJobStateType

|                             |                             |
|-----------------------------|-----------------------------|
| Abstract                    | no                          |
| Translation                 | OpenCase Status             |
| Short description           | Initialer Status des Falls. |
| XML Instance Representation | <pre> &lt;...&gt; </pre>    |

|  |  |
|--|--|
|  | <pre> &lt;AddresseeContext&gt; <a href="#">„ComplexType: DeclareIncidentAddresseeContextType“</a> &lt;/AddresseeContext&gt; [1..1] &lt;/...&gt; </pre> |
|--|--|

## ComplexType: DeclareIncidentAddresseesType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Technical description       | Angaben zum Adressaten  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Addressee&gt; <a href="#">„ComplexType: AddresseeJobType“</a> &lt;/Addressee&gt; [0..unbounded] &lt;/...&gt; </pre> |

## ComplexType: DeclareIncidentAvailableType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Technical description       | Informationen zur Verfügbarkeit  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;IncidentCaseID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/IncidentCaseID&gt; [1..unbounded] &lt;/...&gt; </pre> |

## ComplexType: DeclareIncidentCaseContextType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: CaseContextBaseType“</a>  |
| Translation                 | Kontext des Falls   |
| Technical description       | Dieses Element enthält Informationen und Identifikatoren zum Fall.  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE <a href="#">„ComplexType: CaseContextBaseType“</a> ] &lt;ReceivedStoryIDs&gt; <a href="#">„ComplexType: ReceivedStoriesType“</a> &lt;/ReceivedStoryIDs&gt; [0..1] &lt;SuppressedSenderStoryIDs&gt; <a href="#">„ComplexType: ReceivedStoriesType“</a> &lt;/SuppressedSenderStoryIDs&gt; [0..1] &lt;SuppressedInstitutionStoryIDs&gt; <a href="#">„ComplexType: ReceivedStoriesType“</a> &lt;/SuppressedInstitutionStoryIDs&gt; [0..1] &lt;Warning&gt; <a href="#">„ComplexType: StoryNotificationsType“</a> &lt;/Warning&gt; [0..1] &lt;Info&gt; <a href="#">„ComplexType: StoryNotificationsType“</a> &lt;/Info&gt; [0..1] [ END BASE TYPE ] &lt;IncidentCaseID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/IncidentCaseID&gt; [1..1] &lt;InsuranceCaseID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/InsuranceCaseID&gt; [1..1] &lt;CompanyCaseID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/CompanyCaseID&gt; [1..1] &lt;TestCase&gt; <a href="#">„ComplexType: EmptyType“</a> &lt;/TestCase&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: DeclareIncidentCaseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | Fall   |
| Technical description       | Informationen zum Fall   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;CaseContext&gt; <a href="#">„ComplexType: DeclareIncidentCaseContextType“</a> &lt;/CaseContext&gt; [1..1] &lt;ReceivedState&gt; <a href="#">„ComplexType: DeclareIncidentStateType“</a> &lt;/ReceivedState&gt; [0..1] &lt;DialogMessage&gt; <a href="#">„ComplexType: DialogMessageType“</a> &lt;/DialogMessage&gt; [0..unbounded] &lt;/...&gt; </pre> |

## ComplexType: DeclareIncidentConsumerCaseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | Fall   |
| Technical description       | Informationen zum Fall   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;CaseContext&gt; <a href="#">„ComplexType: DeclareIncidentCaseContextType“</a> &lt;/CaseContext&gt; [1..1] &lt;State&gt; <a href="#">„ComplexType: DeclareIncidentStateType“</a> &lt;/State&gt; [1..1] &lt;DialogMessage&gt; <a href="#">„ComplexType: DialogMessageType“</a> &lt;/DialogMessage&gt; [0..unbounded] &lt;/...&gt; </pre> |

## ComplexType: DeclareIncidentJobStateType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | OpenCase Status  |
| Short description           | Initialer Status des Falls.  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Addressee&gt; „ComplexType: DeclareIncidentAddresseeJobStateType“ &lt;/     Addressee&gt; [0..unbounded] &lt;/...&gt;</pre> |

## ComplexType: DeclareIncidentJobType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Translation                 | Job Steuerung   |
| Technical description       | Steuerungselement für die Verarbeitung der Meldung.   |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Addressees&gt; „ComplexType: DeclareIncidentAddresseesType“ &lt;/Addressees&gt; [1..1]   &lt;Testcase&gt; „ComplexType: EmptyType“ &lt;/Testcase&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: DeclareIncidentRequestType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: RequestType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt;   [ BASE TYPE „ComplexType: RequestType“ ]   &lt;RequestContext&gt; „ComplexType: RequestContextType“ &lt;/RequestContext&gt; [1..1]   [ END BASE TYPE ]   &lt;Job&gt; „ComplexType: DeclareIncidentJobType“ &lt;/Job&gt; [1..1]   &lt;null&gt;idc:blubber&lt;/null&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: DeclareIncidentResponseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: ResponseType“</a>  |
| XML Instance Representation | <pre>&lt;...&gt;   [ BASE TYPE „ComplexType: ResponseType“ ]   &lt;ResponseContext&gt; „ComplexType: RequestContextType“ &lt;/ResponseContext&gt; [1..1]   [ END BASE TYPE ]   &lt;Addressees&gt; „ComplexType: DeclareIncidentJobStateType“ &lt;/Addressees&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: DeclareIncidentStateType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | OpenCase Status  |
| Short description           | Initialer Status des Falls.  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Process&gt; „SimpleType: DeclareIncidentStateEnumType“ &lt;/Process&gt; [1..1]   &lt;Coverage&gt; „SimpleType: CoverageType“ &lt;/Coverage&gt; [1..1]   &lt;SmallRemark&gt; „SimpleType: NotEmptyStringType“ &lt;/SmallRemark&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: RegisterOrganizationAuthenticationJobType

|                       |   |
|-----------------------|---|
| Abstract              | no  |
| Translation           | Job Steuerung                                       |
| Technical description | Steuerungselement für die Verarbeitung der Meldung. |

|                             |  |
|-----------------------------|--|
| XML Instance Representation | <pre> &lt;...&gt;   &lt;Addressee&gt;„ComplexType: AddresseeJobType“&lt;/Addressee&gt; [1..1]   &lt;TestCase&gt;„ComplexType: EmptyType“&lt;/TestCase&gt; [0..1] &lt;/...&gt; </pre> |
|-----------------------------|--|

## ComplexType: RegisterOrganizationAuthenticationRequestType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: RequestType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: RequestType“ ]   &lt;RequestContext&gt;„ComplexType: RequestContextType“&lt;/RequestContext&gt; [1..1] [ END BASE TYPE ]   &lt;Job&gt;„ComplexType: RegisterOrganizationAuthenticationJobType“&lt;/Job&gt; [1..1]   &lt;RegisterOrganization&gt;„ComplexType: CompanyRequestType“&lt;/RegisterOrganizati\ on&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: SubscribeOrganizationAddresseeContextType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: AddresseeResponseContextType“</a>   |
| Technical description       | Kontextangaben zum Adressaten   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: AddresseeResponseContextType“ ] [ BASE TYPE „ComplexType: ResponseContextType“ ]   &lt;UserAgent&gt;„ComplexType: UserAgentType“&lt;/UserAgent&gt; [1..1]   &lt;InstitutionName&gt;xs:string&lt;/InstitutionName&gt; [1..1]   &lt;TransmissionDate&gt;xs:dateTime&lt;/TransmissionDate&gt; [1..1]   &lt;ResponseID&gt;„SimpleType: IDType“&lt;/ResponseID&gt; [1..1]   &lt;RequestID&gt;„SimpleType: IDType“&lt;/RequestID&gt; [1..1] [ END BASE TYPE ]   &lt;ProducerResponseNotifications&gt;„ComplexType: FeedbackNotificationsType“&lt;/Produ\ cerResponseNotifications&gt; [1..1]   &lt;Warning&gt;„ComplexType: NotificationsType“&lt;/Warning&gt; [0..1]   &lt;Info&gt;„ComplexType: NotificationsType“&lt;/Info&gt; [0..1] [ END BASE TYPE ]   &lt;SubscriptionID&gt;„SimpleType: IDType“&lt;/SubscriptionID&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: SubscribeOrganizationAddresseeJobStateType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: AddresseeJobType“</a>  |
| Translation                 | OpenCase Status  |
| Short description           | Initialer Status des Falls.  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: AddresseeJobType“ ] [ BASE TYPE „ComplexType: AddresseeType“ ]   &lt;AddresseeIdentification&gt;„SimpleType: IDType“&lt;/AddresseeIdentificati\ on&gt; [1..1] [ END BASE TYPE ]   &lt;ProcessByDistributor&gt;„SimpleType: SimpleBooleanType“&lt;/ProcessByDistribu\ tor&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ]   &lt;Success&gt;„ComplexType: SubscribeOrganizationAddresseeSuccessJobStateType“&lt;/Suc\ cess&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |

## ComplexType: SubscribeOrganizationAddresseeSuccessJobStateType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | OpenCase Status  |
| Short description           | Initialer Status des Falls.  |
| XML Instance Representation | <pre> &lt;...&gt;   &lt;AddresseeContext&gt;„ComplexType: SubscribeOrganizationAddresseeContextType“&lt;/ AddresseeContext&gt; [1..1] </pre> |

|  |        |
|--|--------|
|  | </...> |
|--|--------|

## ComplexType: SubscribeOrganizationCaseContextType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: CaseContextBaseType“</a>   |
| Translation                 | Kontext des Falls  |
| Technical description       | Dieses Element enthält Informationen und Identifikatoren zum Fall.   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: CaseContextBaseType“ ] &lt;ReceivedStoryIDs&gt; „ComplexType: ReceivedStoriesType“ &lt;/ReceivedStoryIDs&gt; [0..1] &lt;SuppressedSenderStoryIDs&gt; „ComplexType: ReceivedStoriesType“ &lt;/SuppressedSenderStoryIDs&gt; [0..1] &lt;SuppressedInstitutionStoryIDs&gt; „ComplexType: ReceivedStoriesType“ &lt;/SuppressedInstitutionStoryIDs&gt; [0..1] &lt;Warning&gt; „ComplexType: StoryNotificationsType“ &lt;/Warning&gt; [0..1] &lt;Info&gt; „ComplexType: StoryNotificationsType“ &lt;/Info&gt; [0..1] [ END BASE TYPE ] &lt;SubscriptionID&gt; „SimpleType: IDType“ &lt;/SubscriptionID&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: SubscribeOrganizationCaseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | Fall   |
| Technical description       | Informationen zum Fall   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;CaseContext&gt; „ComplexType: SubscribeOrganizationCaseContextType“ &lt;/CaseContext&gt; [1..1] &lt;ReceivedState&gt; „SimpleType: SubscribeOrganizationStateType“ &lt;/ReceivedState&gt; [0..1] &lt;DialogMessage&gt; „ComplexType: DialogMessageType“ &lt;/DialogMessage&gt; [0..unbounded] &lt;Unsubscribe&gt; „ComplexType: EmptyType“ &lt;/Unsubscribe&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: SubscribeOrganizationConsumerCaseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Translation                 | Fall  |
| Technical description       | Informationen zum Fall  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;CaseContext&gt; „ComplexType: SubscribeOrganizationCaseContextType“ &lt;/CaseContext&gt; [1..1] &lt;State&gt; „SimpleType: SubscribeOrganizationStateType“ &lt;/State&gt; [1..1] &lt;DialogMessage&gt; „ComplexType: DialogMessageType“ &lt;/DialogMessage&gt; [0..unbounded] &lt;/...&gt;</pre> |

## ComplexType: SubscribeOrganizationJobStateType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Translation                 | OpenCase Status   |
| Short description           | Initialer Status des Falls.   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Addressee&gt; „ComplexType: SubscribeOrganizationAddresseeJobStateType“ &lt;/Addressee&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: SubscribeOrganizationJobType

|                       |   |
|-----------------------|---|
| Abstract              | no  |
| Translation           | Job Steuerung                                       |
| Technical description | Steuerungselement für die Verarbeitung der Meldung. |

|                             |   |
|-----------------------------|---|
| XML Instance Representation | <pre>&lt;...&gt; &lt;Addressee&gt;„ComplexType: AddresseeJobType“&lt;/Addressee&gt; [1..1] &lt;/...&gt;</pre> |
|-----------------------------|---|

## ComplexType: SubscribeOrganizationRequestType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: RequestType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: RequestType“ ] &lt;RequestContext&gt;„ComplexType: RequestContextType“&lt;/RequestContext&gt; [1..1] [ END BASE TYPE ] &lt;Job&gt;„ComplexType: SubscribeOrganizationJobType“&lt;/Job&gt; [1..1] &lt;Company&gt;„ComplexType: CompanyRequestType“&lt;/Company&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: SubscribeOrganizationResponseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: ResponseType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: ResponseType“ ] &lt;ResponseContext&gt;„ComplexType: ResponseContextType“&lt;/ResponseContext&gt; [1..1] [ END BASE TYPE ] &lt;Addressees&gt;„ComplexType: SubscribeOrganizationJobStateType“&lt;/Addressees&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: SynchronizeDeclareIncidentConsumerType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: AddresseeResponseType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: AddresseeResponseType“ ] &lt;AddresseeContext&gt;„ComplexType: AddresseeResponseContextType“&lt;/AddresseeCon\ text&gt; [1..1] [ END BASE TYPE ] &lt;Addressee&gt;„ComplexType: AddresseeType“&lt;/Addressee&gt; [1..1] &lt;Case&gt;„ComplexType: DeclareIncidentConsumerCaseType“&lt;/Case&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: SynchronizeDeclareIncidentRequestType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: SynchronizeRequestType“</a>  |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: SynchronizeRequestType“ ] [ BASE TYPE „ComplexType: RequestType“ ] &lt;RequestContext&gt;„ComplexType: RequestContextType“&lt;/RequestContext&gt; [1..1] [ END BASE TYPE ] &lt;Sender&gt;„ComplexType: CompanyUIDType“&lt;/Sender&gt; [1..1] [ END BASE TYPE ] &lt;Addressee&gt;„ComplexType: AddressingAddresseeType“&lt;/Addressee&gt; [1..1] &lt;Case&gt;„ComplexType: DeclareIncidentCaseType“&lt;/Case&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: SynchronizeDeclareIncidentResponseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: ResponseType“</a>  |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: ResponseType“ ] &lt;ResponseContext&gt;„ComplexType: ResponseContextType“&lt;/ResponseContext&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;Error&gt;„ComplexType: ErrorResponseType“&lt;/Error&gt; [1..1] &lt;SynchronizeDeclareIncidentConsumer&gt;„ComplexType: SynchronizeDeclareIncidentConsumerType“&lt;/ SynchronizeDeclareIncidentConsumer&gt; [1..1] [ END CHOICE ] &lt;/...&gt;</pre> |

## ComplexType: SynchronizeRegisterOrganizationAuthenticationRequestType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: SynchronizeRequestType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: SynchronizeRequestType“ ] [ BASE TYPE „ComplexType: RequestType“ ] &lt;RequestContext&gt;„ComplexType: RequestContextType“&lt;/RequestContext&gt; [1..1] [ END BASE TYPE ] &lt;Sender&gt;„ComplexType: CompanyUIDType“&lt;/Sender&gt; [1..1] [ END BASE TYPE ] &lt;Addressee&gt;„ComplexType: AddressingAddresseeType“&lt;/Addressee&gt; [1..1] &lt;Case&gt;„ComplexType: RegisterOrganizationAuthenticationSenderRequestCaseType“&lt;/Case&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: SynchronizeSubscribeOrganizationConsumerType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: AddresseeResponseType“</a>  |
| Technical description       | Dieses Element synchronisiert die Anmeldung der Organisation.   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: AddresseeResponseType“ ] &lt;AddresseeContext&gt;„ComplexType: AddresseeResponseContextType“&lt;/AddresseeCon\ text&gt; [1..1] [ END BASE TYPE ] &lt;Addressee&gt;„ComplexType: AddresseeType“&lt;/Addressee&gt; [1..1] &lt;Case&gt;„ComplexType: SubscribeOrganizationConsumerCaseType“&lt;/Case&gt; [1..1] &lt;Available&gt;„ComplexType: AvailableType“&lt;/Available&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: SynchronizeSubscribeOrganizationRequestType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: SynchronizeRequestType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: SynchronizeRequestType“ ] [ BASE TYPE „ComplexType: RequestType“ ] &lt;RequestContext&gt;„ComplexType: RequestContextType“&lt;/RequestContext&gt; [1..1] [ END BASE TYPE ] &lt;Sender&gt;„ComplexType: CompanyUIDType“&lt;/Sender&gt; [1..1] [ END BASE TYPE ] &lt;Addressee&gt;„ComplexType: AddressingAddresseeType“&lt;/Addressee&gt; [1..1] &lt;Case&gt;„ComplexType: SubscribeOrganizationCaseType“&lt;/Case&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: SynchronizeSubscribeOrganizationResponseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: ResponseType“</a>   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: ResponseType“ ] &lt;ResponseContext&gt;„ComplexType: ResponseContextType“&lt;/ResponseContext&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;Error&gt;„ComplexType: ErrorResponseType“&lt;/Error&gt; [1..1] &lt;SynchronizeSubscribeOrganizationConsumer&gt;„ComplexType: SynchronizeSubscribeOrganizationConsumerType“&lt;/SynchronizeSubscribeOrganizationConsumer&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |

## SimpleType: DeclareIncidentStateEnumType

|              |   |
|--------------|---|
| Content type | Restriction   |
| Base type    | xs:string   |
| Facets       | <ul style="list-style-type: none"> <li>Enum: RegistrationReceived</li> <li>Enum: CheckingDeclaration</li> </ul> |

|  |   |
|--|---|
|  | <ul style="list-style-type: none"> <li>• Enum: VerifyingClaim</li> <li>• Enum: PerformingEntitlement</li> <li>• Enum: CommunicationStopped</li> </ul> |
|--|---|

### SimpleType: SubscribeOrganizationStateType

|              |  |
|--------------|--|
| Content type | Restriction  |
| Base type    | xs:string  |
| Facets       | <ul style="list-style-type: none"> <li>• Enum: subscribed</li> <li>• Enum: closed</li> </ul> |

## G.4 Schema Dokumentation IncidentDeclaration.xsd

|                     |  |
|---------------------|--|
| Target Namespace    | urn:ch:swissdec:kle:v2:20260306:incidentdeclaration  |
| Declared Namespaces | <ul style="list-style-type: none"> <li>• c : urn:ch:swissdec:common:v3:20260306</li> <li>• ep : urn:ch:swissdec:basis:v1:20260306:components</li> <li>• id : urn:ch:swissdec:kle:v2:20260306:incidentdeclaration</li> <li>• xs : http://www.w3.org/2001/XMLSchema</li> </ul> |
| Version             | 0.0  |

### ComplexType: AbsenceType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Date&gt;xs:date&lt;/Date&gt;    [1..1]   &lt;Hours&gt;„SimpleType: HoursOrLessonsType“ &lt;/Hours&gt;    [1..1] &lt;/...&gt;</pre> |

### ComplexType: AbsencesType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | „ComplexType: StoryBaseType“  |
| XML Instance Representation | <pre>&lt;...&gt;   [ BASE TYPE „ComplexType: StoryBaseType“ ]   &lt;Creation&gt;xs:dateTime&lt;/Creation&gt;    [1..1]   &lt;StoryID&gt;„SimpleType: IDType“ &lt;/StoryID&gt;    [1..1]   [ END BASE TYPE ]   &lt;Absence&gt;„ComplexType: AbsenceType“ &lt;/Absence&gt;    [1..unbounded] &lt;/...&gt;</pre> |

### ComplexType: AccidentAtLeisureType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;LastWorkingDay&gt;xs:date&lt;/LastWorkingDay&gt;    [1..1]   &lt;ActivityAtTimeOfAccident&gt;„SimpleType: ActivityAtTimeOfAccidentType“ &lt;/ActivityAt\ TimeOfAccident&gt;    [1..1] &lt;/...&gt;</pre> |

### ComplexType: AccidentAtWorkType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;InvolvedObjects&gt;„SimpleType: NotEmptyStringType“ &lt;/InvolvedObjects&gt;    [0..1] &lt;/...&gt;</pre> |

### ComplexType: AccidentDescriptionType

|          |    |
|----------|----|
| Abstract | no |
|----------|----|



|                             |  |
|-----------------------------|--|
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt;„SimpleType: IDType“&lt;/StoryID&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;Accident&gt;„ComplexType: ClassOfInsuranceType“&lt;/Accident&gt; [1..1] &lt;OccupationalDisease&gt;„ComplexType: EmptyType“&lt;/OccupationalDisease&gt; [1..1] &lt;CaseOfDeath&gt;„ComplexType: CaseOfDeathType“&lt;/CaseOfDeath&gt; [1..1] &lt;Relapse&gt;„ComplexType: StoryRelapseType“&lt;/Relapse&gt; [1..1] [ END CHOICE ] &lt;IncidentDate&gt;„ComplexType: StoryIncidentDateTimeType“&lt;/IncidentDate&gt; [1..1] &lt;IncidentScene&gt;xs:string&lt;/IncidentScene&gt; [1..1] &lt;IncidentDescription&gt;xs:string&lt;/IncidentDescription&gt; [1..1] &lt;PoliceReportExists&gt;„SimpleType: YesNoUnknownType“&lt;/PoliceReportExists&gt; [1..1] &lt;Injury&gt;„ComplexType: InjuryType“&lt;/Injury&gt; [1..1] &lt;ExistInvolvedPersons&gt;„SimpleType: YesNoUnknownType“&lt;/ExistInvolvedPer\ sons&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: AccidentRegistrationType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;CompanyCaseID&gt;„SimpleType: IDType“&lt;/CompanyCaseID&gt; [1..1] [ START CHOICE ] &lt;Incident&gt;„ComplexType: IncidentAccidentRegistrationType“&lt;/Incident&gt; [1..1] &lt;Relapse&gt;„ComplexType: RelapseAccidentRegistrationType“&lt;/Relapse&gt; [1..1] [ END CHOICE ] &lt;CaseOfIncapacitiesToWork&gt;„ComplexType: EmptyType“&lt;/CaseOfIncapacitiesTo\ Work&gt; [0..1] [ START CHOICE ] &lt;InsuranceCode&gt;„SimpleType: AssuranceCategoryCodeType“&lt;/ InsuranceCode&gt; [1..unbounded] &lt;WithoutContract&gt;„ComplexType: EmptyType“&lt;/WithoutContract&gt; [1..1] [ END CHOICE ] &lt;HRLanguageCode&gt;„SimpleType: LanguageCodeType“&lt;/HRLanguageCode&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: AdditionsPersonType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt;„SimpleType: IDType“&lt;/StoryID&gt; [1..1] [ END BASE TYPE ] &lt;Position&gt;„SimpleType: PositionType“&lt;/Position&gt; [1..1] &lt;Contract&gt;„ComplexType: ContractType“&lt;/Contract&gt; [1..1] &lt;ContractFixedUntil&gt;xs:date&lt;/ContractFixedUntil&gt; [0..1] &lt;ContractDissolveFrom&gt;xs:date&lt;/ContractDissolveFrom&gt; [0..1] &lt;JobTitle&gt;„SimpleType: IDType“&lt;/JobTitle&gt; [1..1] &lt;TaxAtSource&gt;„ComplexType: TaxAtSourceType“&lt;/TaxAtSource&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: AddressingAddresseeType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: AddresseeType“</a>   |
| Short description           | Informationen zum Adressaten   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: AddresseeType“ ] &lt;AddresseeIdentification&gt;„SimpleType: IDType“&lt;/AddresseeIdentification&gt; [1..1] [ END BASE TYPE ] &lt;Domain&gt;„SimpleType: AddressingDomainType“&lt;/Domain&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: AddressingIdentificationType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; [ START CHOICE ] </pre> |

|  |  |
|--|--|
|  | <pre> &lt;UVG-LAA&gt;_ComplexType: UVG-LAA-IdentificationType" _&lt;/UVG-LAA&gt; [1..1] &lt;UVGZ-LAAC&gt;_ComplexType: IdentificationBaseType" _&lt;/UVGZ-LAAC&gt; [1..1] &lt;KTG-AMC&gt;_ComplexType: IdentificationBaseType" _&lt;/KTG-AMC&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |
|--|--|

## ComplexType: AddressingType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Institution&gt;_ComplexType: AddressingIdentificationType" _&lt;/Institution&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: AgreementType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">_ComplexType: StoryBaseType"</a>   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE _ComplexType: StoryBaseType" ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt;_SimpleType: IDType" _&lt;/StoryID&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;OtherInsurancePredefined&gt;_SimpleType: OtherInsurancePredefinedType" _&lt;/OtherIn\ surancePredefined&gt; [1..1] &lt;OtherInsurance&gt;_SimpleType: NotEmptyStringType" _&lt;/OtherInsurance&gt; [1..1] &lt;NoOtherInsurance&gt;_ComplexType: EmptyType" _&lt;/NoOtherInsurance&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |

## ComplexType: AllowancesPastType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Salary&gt;_ComplexType: InsuredSalaryPastType" _&lt;/Salary&gt; [1..12] &lt;AllowancesAnnualTotal&gt;_SimpleType: SalaryAmountType" _&lt;/AllowancesAnnualTo\ tal&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: AnnualSalaryType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">_ComplexType: StoryBaseType"</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE _ComplexType: StoryBaseType" ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt;_SimpleType: IDType" _&lt;/StoryID&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;IncidentMonth&gt;xs:gYearMonth&lt;/IncidentMonth&gt; [1..1] &lt;Present&gt;_ComplexType: KindOfSalariesTimedType" _&lt;/Present&gt; [0..1] &lt;Past&gt;_ComplexType: MonthSalaryType" _&lt;/Past&gt; [0..1] &lt;ManualAdditions&gt;_ComplexType: KindOfSalaryType" _&lt;/ ManualAdditions&gt; [0..unbounded] &lt;AnnualSalaryTotal&gt;_SimpleType: SalaryAmountType" _&lt;/AnnualSalaryTotal&gt; [1..1] &lt;Statistic&gt;_ComplexType: SalaryBasesType" _&lt;/Statistic&gt; [1..1] &lt;NoIncapacityToWork&gt;_ComplexType: EmptyType" _&lt;/NoIncapacityToWork&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |

## ComplexType: AnnualWorkingTimeType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">_ComplexType: StoryBaseType"</a>   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE _ComplexType: StoryBaseType" ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt;_SimpleType: IDType" _&lt;/StoryID&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;MonthlyWorkingTimes&gt;_ComplexType: MonthlyWorkingTimesType" _&lt;/MonthlyWorkingTi\ mes&gt; [1..1] </pre> |

|  |   |
|--|---|
|  | <pre> &lt;NoData&gt;„ComplexType: EmptyType“ &lt;/NoData&gt;    [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |
|--|---|

## ComplexType: AttachmentType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:date&lt;/Creation&gt;    [1..1] &lt;StoryID&gt;„SimpleType: IDType“ &lt;/StoryID&gt;    [1..1] [ END BASE TYPE ] &lt;Filename&gt;„SimpleType: NotEmptyStringType“ &lt;/Filename&gt;    [1..1] &lt;ContentType&gt;„SimpleType: ContentTypeType“ &lt;/ContentType&gt;    [1..1] &lt;Data&gt;xs:base64Binary&lt;/Data&gt;    [1..1] &lt;/...&gt; </pre> |

## ComplexType: AwaitPartControlType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;ExpirationDate&gt;xs:date&lt;/ExpirationDate&gt;    [0..1] &lt;/...&gt; </pre> |

## ComplexType: AwaitPartType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;IncapacitiesToWork&gt;„ComplexType: AwaitPartControlType“ &lt;/IncapacitiesTo\ Work&gt;    [0..1] &lt;TakeWorkUpFully&gt;„ComplexType: AwaitPartControlType“ &lt;/TakeWorkUpFully&gt;    [0..1] &lt;Treatments&gt;„ComplexType: AwaitPartControlType“ &lt;/Treatments&gt;    [0..1] &lt;Agreement&gt;„ComplexType: AwaitPartControlType“ &lt;/Agreement&gt;    [0..1] &lt;AnnualSalary&gt;„ComplexType: AwaitPartControlType“ &lt;/AnnualSalary&gt;    [0..1] &lt;AnnualWorkingTime&gt;„ComplexType: AwaitPartControlType“ &lt;/AnnualWorking\ Time&gt;    [0..1] &lt;Payment&gt;„ComplexType: AwaitPartControlType“ &lt;/Payment&gt;    [0..1] &lt;AdditionsPerson&gt;„ComplexType: AwaitPartControlType“ &lt;/AdditionsPerson&gt;    [0..1] &lt;Absence&gt;„ComplexType: AwaitPartControlType“ &lt;/Absence&gt;    [0..1] &lt;SpecialCode&gt;„ComplexType: AwaitPartControlType“ &lt;/SpecialCode&gt;    [0..1] &lt;AccidentDescription&gt;„ComplexType: AwaitPartControlType“ &lt;/AccidentDescripti\ on&gt;    [0..1] &lt;/...&gt; </pre> |

## ComplexType: BenefitPeriodType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: TimePeriodType“</a>   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: TimePeriodType“ ] &lt;from&gt;xs:date&lt;/from&gt;    [1..1] &lt;until&gt;xs:date&lt;/until&gt;    [1..1] [ END BASE TYPE ] &lt;IncapacityToWork&gt;„SimpleType: PercentType“ &lt;/IncapacityToWork&gt;    [0..1] &lt;/...&gt; </pre> |

## ComplexType: CaseOfDeathType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;DateOfDeath&gt;xs:date&lt;/DateOfDeath&gt;    [1..1] &lt;ClassOfInsurance&gt;„ComplexType: ClassOfInsuranceType“ &lt;/ClassOfInsurance&gt;    [1..1] &lt;/...&gt; </pre> |

## ComplexType: ChangelIdentitiesType

|                             |                          |
|-----------------------------|--------------------------|
| Abstract                    | no                       |
| XML Instance Representation | <pre> &lt;...&gt; </pre> |

|  |   |
|--|---|
|  | <pre> &lt;ValidAsOf&gt;xs:date&lt;/ValidAsOf&gt; [1..1] &lt;Company&gt; „ComplexType: CompanyIdentitiesType“ &lt;/Company&gt; [0..1] &lt;Institution&gt; „ComplexType: InstitutionIdentitiesType“ &lt;/Institution&gt; [0..1] &lt;/...&gt; </pre> |
|--|---|

## ComplexType: ClassOfInsuranceType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; [ START CHOICE ] &lt;AccidentAtWork&gt; „ComplexType: AccidentAtWorkType“ &lt;/AccidentAtWork&gt; [1..1] &lt;AccidentAtLeisure&gt; „ComplexType: AccidentAtLeisureType“ &lt;/AccidentAtLeisu\ re&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |

## ComplexType: CompanyIdentitiesType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Old&gt; „ComplexType: CompanyUIDType“ &lt;/Old&gt; [1..1] &lt;New&gt; „ComplexType: CompanyUIDType“ &lt;/New&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: CompanyUIDType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;UID-BFS&gt; „ComplexType: UID-BFS-UnknownType“ &lt;/UID-BFS&gt; [1..1] &lt;DelegateUID-BFS&gt; „SimpleType: UID-BFSType“ &lt;/DelegateUID-BFS&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: ContractIndemnificationType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;ContractIdentity&gt; „SimpleType: NotEmptyStringType“ &lt;/ContractIdentity&gt; [1..1] &lt;InsuranceCode&gt; „SimpleType: AssuranceCategoryCodeType“ &lt;/ InsuranceCode&gt; [0..unbounded] &lt;/...&gt; </pre> |

## ComplexType: ContractType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | Vertrag  |
| Short description           | Art des Arbeitsvertrags  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;KindOf&gt; „SimpleType: EmploymentContractType“ &lt;/KindOf&gt; [1..1] &lt;ValidAsOf&gt;xs:date&lt;/ValidAsOf&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: CoveredDailyAllowanceType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | „ComplexType: StoryBaseType“   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:date&lt;/Creation&gt; [1..1] &lt;StoryID&gt; „SimpleType: IDType“ &lt;/StoryID&gt; [1..1] [ END BASE TYPE ] &lt;Allowance&gt; „SimpleType: SalaryAmountType“ &lt;/Allowance&gt; [1..1] &lt;ValidAsOf&gt;xs:date&lt;/ValidAsOf&gt; [0..1] &lt;ValidUntil&gt;xs:date&lt;/ValidUntil&gt; [0..1] &lt;Provisional&gt; „ComplexType: EmptyType“ &lt;/Provisional&gt; [0..1] &lt;Description&gt; „SimpleType: NotEmptyStringType“ &lt;/Description&gt; [0..1] &lt;Reduction&gt;id:blubber&lt;/Reduction&gt; [0..1] &lt;HospitalCostDeduction&gt; „SimpleType: HospitalCostDeductionEnumType“ &lt;/HospitalCost\ Deduction&gt; [0..1] </pre> |

|  |  |
|--|--|
|  | <code>&lt;NetAllowance&gt; <a href="#">„SimpleType: SalaryAmountType“</a> &lt;/NetAllowance&gt; [0..1]</code><br><code>&lt;/...&gt;</code> |
|--|--|

## ComplexType: CustomerIdentificationType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;UVG-LAA&gt; <a href="#">„ComplexType: InstitutionType“</a> &lt;/UVG-LAA&gt; [0..1]</code><br><code>&lt;UVGZ-LAAC&gt; <a href="#">„ComplexType: IdentificationBaseType“</a> &lt;/UVGZ-LAAC&gt; [0..unbounded]</code><br><code>&lt;KTG-AMC&gt; <a href="#">„ComplexType: IdentificationBaseType“</a> &lt;/KTG-AMC&gt; [0..1]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: FavourCompanyType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;Payment&gt; <a href="#">„SimpleType: SalaryAmountAbsoluteType“</a> &lt;/Payment&gt; [1..1]</code><br><code>&lt;PaymentThird&gt; <a href="#">„SimpleType: SalaryAmountAbsoluteType“</a> &lt;/PaymentThird&gt; [0..1]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: FavourInstitutionType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;Repayment&gt; <a href="#">„SimpleType: SalaryAmountAbsoluteType“</a> &lt;/Repayment&gt; [1..1]</code><br><code>&lt;RepaymentThird&gt; <a href="#">„SimpleType: SalaryAmountAbsoluteType“</a> &lt;/RepaymentThird&gt; [0..1]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: HourlyPayType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;HourlySalary&gt; <a href="#">„SimpleType: SalaryAmountType“</a> &lt;/HourlySalary&gt; [1..1]</code><br><code>&lt;Earnings13th&gt; <a href="#">„SimpleType: PercentType“</a> &lt;/Earnings13th&gt; [0..1]</code><br><code>&lt;VacationAndPublicHolidayCompensation&gt; <a href="#">„SimpleType: PercentType“</a> &lt;/VacationAndPublicHolidayCompensation&gt; [0..1]</code><br><code>&lt;StatisticCode&gt; <a href="#">„SimpleType: SalaryBaseCodeType“</a> &lt;/StatisticCode&gt; [1..1]</code><br><code>&lt;AnnualHourSalary&gt; <a href="#">„SimpleType: SalaryAmountType“</a> &lt;/AnnualHourSalary&gt; [1..1]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: HourlyWagePastType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;Salary&gt; <a href="#">„ComplexType: InsuredSalaryHourlyPastType“</a> &lt;/Salary&gt; [1..12]</code><br><code>&lt;HourlyWageAnnualTotal&gt; <a href="#">„SimpleType: SalaryAmountType“</a> &lt;/HourlyWageAnnualTotal&gt; [1..1]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: IllnessRegistrationType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;CompanyCaseID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/CompanyCaseID&gt; [1..1]</code><br><code>&lt;IncapacitiesToWork&gt;id:blubber&lt;/IncapacitiesToWork&gt; [0..1]</code><br><code>&lt;Incident&gt; <a href="#">„ComplexType: IncidentAccidentRegistrationBaseType“</a> &lt;/Incident&gt; [0..1]</code><br><code>&lt;DateOfBirth&gt;xs:date&lt;/DateOfBirth&gt; [0..1]</code><br><code>&lt;InsuranceCaseID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/InsuranceCaseID&gt; [0..1]</code><br><code>&lt;InsuranceCode&gt; <a href="#">„SimpleType: AssuranceCategoryCodeType“</a> &lt;/InsuranceCode&gt; [1..unbounded]</code><br><code>&lt;HRLanguageCode&gt; <a href="#">„SimpleType: LanguageCodeType“</a> &lt;/HRLanguageCode&gt; [0..1]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: IncapacitiesToWorkType

|             |  |
|-------------|--|
| Abstract    | no   |
| Parent type | <a href="#">„ComplexType: StoryBaseType“</a> |

|                             |   |
|-----------------------------|---|
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt;„SimpleType: IDType“ &lt;/StoryID&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;IncapacityToWork&gt;„ComplexType: IncapacityToWorkType“ &lt;/ IncapacityToWork&gt; [1..unbounded] &lt;NoIncapacityToWork&gt; „ComplexType: EmptyType“ &lt;/NoIncapacityToWork&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |
|-----------------------------|---|

## ComplexType: IncapacityToWorkType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Percentage&gt;„SimpleType: Max100PercentType“ &lt;/Percentage&gt; [1..1] &lt;EffectiveFrom&gt;xs:date&lt;/EffectiveFrom&gt; [1..1] &lt;Valid&gt;„ComplexType: ValidToType“ &lt;/Valid&gt; [1..1] &lt;Provisional&gt;„ComplexType: EmptyType“ &lt;/Provisional&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: IncidentAccidentRegistrationBaseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;IncidentDate&gt;„ComplexType: RegistrationIncidentDateType“ &lt;/IncidentDate&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: IncidentAccidentRegistrationType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: IncidentAccidentRegistrationBaseType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: IncidentAccidentRegistrationBaseType“ ] &lt;IncidentDate&gt;„ComplexType: RegistrationIncidentDateType“ &lt;/IncidentDate&gt; [1..1] [ END BASE TYPE ] &lt;InsuranceCaseID&gt;„SimpleType: IDType“ &lt;/InsuranceCaseID&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: IncidentCaseIDType

|                             |                                       |
|-----------------------------|---------------------------------------|
| Abstract                    | no                                    |
| Translation                 | IncidentCaseID                        |
| XML Instance Representation | <pre> &lt;...&gt; &lt;/...&gt; </pre> |

## ComplexType: IncidentContextRequestType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;ReceivedState&gt;„ComplexType: StateType“ &lt;/ReceivedState&gt; [0..1] &lt;Info&gt;„ComplexType: StoryNotificationRequestType“ &lt;/Info&gt; [0..unbounded] &lt;Warning&gt;„ComplexType: StoryNotificationRequestType“ &lt;/Warning&gt; [0..unbounded] &lt;Error&gt;„ComplexType: StoryNotificationRequestType“ &lt;/Error&gt; [0..unbounded] &lt;/...&gt; </pre> |

## ComplexType: IncidentDeclarationType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Technical description       | Fachliche Datenstruktur für die Ereignismeldung.   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;CompanyDescription&gt;„ComplexType: CompanyDescriptionType“ &lt;/CompanyDescripti\ on&gt; [1..1] &lt;Person&gt;„ComplexType: PersonIncidentType“ &lt;/Person&gt; [1..1] &lt;Institutions&gt;„ComplexType: CustomerIdentificationType“ &lt;/Institutions&gt; [1..1] &lt;Contact&gt;„ComplexType: ContactType“ &lt;/Contact&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: IncidentForCompanyClosedType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; [ START CHOICE ] &lt;Normal&gt; „ComplexType: EmptyType“ &lt;/Normal&gt; [1..1] &lt;Cancel&gt;xs:string&lt;/Cancel&gt; [1..1] [ END CHOICE ] &lt;/...&gt;</pre> |

## ComplexType: IncidentStoriesRequestType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;IncidentContext&gt; „ComplexType: IncidentContextRequestType“ &lt;/IncidentCon\ text&gt; [1..1] &lt;IncapacitiesToWork&gt; „ComplexType: IncapacitiesToWorkType“ &lt;/IncapacitiesTo\ Work&gt; [0..1] &lt;TakeWorkUpFully&gt; „ComplexType: TakeWorkUpFullyType“ &lt;/TakeWorkUpFully&gt; [0..1] &lt;Treatments&gt; „ComplexType: TreatmentsType“ &lt;/Treatments&gt; [0..1] &lt;Agreement&gt; „ComplexType: AgreementType“ &lt;/Agreement&gt; [0..1] &lt;AnnualSalary&gt; „ComplexType: AnnualSalaryType“ &lt;/AnnualSalary&gt; [0..1] &lt;Payment&gt; „ComplexType: PaymentType“ &lt;/Payment&gt; [0..1] &lt;AdditionsPerson&gt; „ComplexType: AdditionsPersonType“ &lt;/AdditionsPerson&gt; [0..1] &lt;Absences&gt; „ComplexType: AbsencesType“ &lt;/Absences&gt; [0..1] &lt;SpecialCode&gt; „ComplexType: SpecialCodeType“ &lt;/SpecialCode&gt; [0..1] &lt;AccidentDescription&gt; „ComplexType: AccidentDescriptionType“ &lt;/AccidentDescripti\ on&gt; [0..1] &lt;SalaryRaise&gt; „ComplexType: SalaryRaiseType“ &lt;/SalaryRaise&gt; [0..1] &lt;PersonMutated&gt; „ComplexType: PersonMutatedType“ &lt;/PersonMutated&gt; [0..1] &lt;SpecialCoverage&gt; „ComplexType: SpecialCoverageType“ &lt;/SpecialCoverage&gt; [0..1] &lt;OtherEmployers&gt; „ComplexType: BooleanStoryType“ &lt;/OtherEmployers&gt; [0..1] &lt;ShortTimeWork&gt; „ComplexType: BooleanStoryType“ &lt;/ShortTimeWork&gt; [0..1] &lt;DialogMessage&gt; „ComplexType: DialogMessageType“ &lt;/DialogMessage&gt; [0..unbounded] &lt;Attachment&gt; „ComplexType: AttachmentType“ &lt;/Attachment&gt; [0..unbounded] &lt;ProcessCtrl&gt; „ComplexType: ProcessCtrlRequestType“ &lt;/ProcessCtrl&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: IndemnificationCorrectionType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | „ComplexType: IndemnificationType“   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: IndemnificationType“ ] &lt;IndemnificationID&gt; „SimpleType: IDType“ &lt;/IndemnificationID&gt; [1..1] &lt;ProductCategory&gt; „SimpleType: NotEmptyStringType“ &lt;/ProductCategory&gt; [0..1] &lt;WaitingDays&gt; „SimpleType: DaysType“ &lt;/WaitingDays&gt; [0..1] &lt;CoveringDays&gt; „SimpleType: DaysType“ &lt;/CoveringDays&gt; [0..1] &lt;IncapacityToWork&gt; „SimpleType: PercentType“ &lt;/IncapacityToWork&gt; [0..1] &lt;QuantityOfDays&gt;xs:short&lt;/QuantityOfDays&gt; [0..1] &lt;Period&gt; „ComplexType: TimePeriodType“ &lt;/Period&gt; [0..1] &lt;EffectiveDailyAllowance&gt; „SimpleType: SalaryAmountType“ &lt;/EffectiveDailyAllowan\ ce&gt; [0..1] &lt;Reduction&gt; „ComplexType: ReductionType“ &lt;/Reduction&gt; [0..unbounded] &lt;Balance&gt; „ComplexType: BalanceType“ &lt;/Balance&gt; [1..1] &lt;Contract&gt; „ComplexType: ContractIndemnificationType“ &lt;/Contract&gt; [0..1] &lt;Comment&gt; „ComplexType: NotificationsType“ &lt;/Comment&gt; [0..1] [ END BASE TYPE ] &lt;LinkToOldIndemnification&gt; „ComplexType: LinkToOldIndemnificationType“ &lt;/LinkToOl\ dIndemnification&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: IndemnificationType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;IndemnificationID&gt; „SimpleType: IDType“ &lt;/IndemnificationID&gt; [1..1] &lt;ProductCategory&gt; „SimpleType: NotEmptyStringType“ &lt;/ProductCategory&gt; [0..1] &lt;WaitingDays&gt; „SimpleType: DaysType“ &lt;/WaitingDays&gt; [0..1] &lt;CoveringDays&gt; „SimpleType: DaysType“ &lt;/CoveringDays&gt; [0..1] &lt;IncapacityToWork&gt; „SimpleType: PercentType“ &lt;/IncapacityToWork&gt; [0..1] &lt;QuantityOfDays&gt;xs:short&lt;/QuantityOfDays&gt; [0..1] &lt;Period&gt; „ComplexType: TimePeriodType“ &lt;/Period&gt; [0..1] &lt;EffectiveDailyAllowance&gt; „SimpleType: SalaryAmountType“ &lt;/EffectiveDailyAllowan\ ce&gt; [0..1] &lt;Reduction&gt; „ComplexType: ReductionType“ &lt;/Reduction&gt; [0..unbounded] &lt;Balance&gt; „ComplexType: BalanceType“ &lt;/Balance&gt; [1..1] &lt;Contract&gt; „ComplexType: ContractIndemnificationType“ &lt;/Contract&gt; [0..1]</pre> |

|  |  |
|--|--|
|  | <code>&lt;Comment&gt;_ComplexType: NotificationsType_&lt;/Comment&gt; [0..1]</code><br><code>&lt;/...&gt;</code> |
|--|--|

## ComplexType: IndemnificationsType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;Normal&gt;_ComplexType: IndemnificationType_&lt;/Normal&gt; [0..unbounded]</code><br><code>&lt;Correction&gt;_ComplexType: IndemnificationCorrectionType_&lt;/</code><br><code>Correction&gt; [0..unbounded]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: InjuredPartOfBodyType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;PartOfBody&gt;_ComplexType: PartOfBodyType_&lt;/PartOfBody&gt; [1..1]</code><br><code>[ START CHOICE ]</code><br><code>&lt;TypeOfInjury&gt;_SimpleType: TypeOfInjuryType_&lt;/TypeOfInjury&gt; [1..1]</code><br><code>&lt;OtherTypeOfInjury&gt;xs:string&lt;/OtherTypeOfInjury&gt; [1..1]</code><br><code>[ END CHOICE ]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: InjuryType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;InjuredPartOfBody&gt;_ComplexType: InjuredPartOfBodyType_&lt;/InjuredPartOfBo\</code><br><code>dy&gt; [1..3]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: InstitutionChangeType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;CustomerId&gt;_SimpleType: NotEmptyStringType_&lt;/CustomerId&gt; [1..1]</code><br><code>&lt;ContractId&gt;_SimpleType: NotEmptyStringType_&lt;/ContractId&gt; [0..1]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: InstitutionIdentitiesType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;Old&gt;_ComplexType: InstitutionChangeType_&lt;/Old&gt; [1..1]</code><br><code>&lt;New&gt;_ComplexType: InstitutionChangeType_&lt;/New&gt; [1..1]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: InstitutionIllnessType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;InsuranceCompanyName&gt;xs:string&lt;/InsuranceCompanyName&gt; [1..1]</code><br><code>&lt;CustomerId&gt;_SimpleType: NotEmptyStringType_&lt;/CustomerId&gt; [1..1]</code><br><code>&lt;ContractId&gt;_SimpleType: NotEmptyStringType_&lt;/ContractId&gt; [1..1]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: InstitutionType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;InsuranceCompanyName&gt;xs:string&lt;/InsuranceCompanyName&gt; [1..1]</code><br><code>&lt;CustomerId&gt;_SimpleType: NotEmptyStringType_&lt;/CustomerId&gt; [1..1]</code><br><code>[ START CHOICE ]</code><br><code>&lt;ContractId&gt;_SimpleType: NotEmptyStringType_&lt;/ContractId&gt; [1..1]</code><br><code>&lt;NoContract&gt;_ComplexType: EmptyType_&lt;/NoContract&gt; [1..1]</code><br><code>[ END CHOICE ]</code><br><code>&lt;/...&gt;</code> |



## ComplexType: InsuredSalaryHourlyPastType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Month&gt;xs:gYearMonth&lt;/Month&gt; [1..1] &lt;KindOfSalaries&gt;„ComplexType: KindOfSalariesPastType“ &lt;/KindOfSalaries&gt; [1..1] &lt;Total&gt;„SimpleType: SalaryAmountType“ &lt;/Total&gt; [1..1] &lt;HourlySalary&gt;„SimpleType: SalaryAmountType“ &lt;/HourlySalary&gt; [0..1] &lt;HoursPerMonth&gt;„SimpleType: HoursAndIndustryMinutesType“ &lt;/HoursPerMonth&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: InsuredSalaryLessonPastType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Month&gt;xs:gYearMonth&lt;/Month&gt; [1..1] &lt;KindOfSalaries&gt;„ComplexType: KindOfSalariesPastType“ &lt;/KindOfSalaries&gt; [1..1] &lt;Total&gt;„SimpleType: SalaryAmountType“ &lt;/Total&gt; [1..1] &lt;LessonSalary&gt;„SimpleType: SalaryAmountType“ &lt;/LessonSalary&gt; [0..1] &lt;LessonsPerMonth&gt;„SimpleType: HoursAndIndustryMinutesType“ &lt;/LessonsPerMonth&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: InsuredSalaryPastType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Month&gt;xs:gYearMonth&lt;/Month&gt; [1..1] &lt;KindOfSalaries&gt;„ComplexType: KindOfSalariesPastType“ &lt;/KindOfSalaries&gt; [1..1] &lt;Total&gt;„SimpleType: SalaryAmountType“ &lt;/Total&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: KindOfSalariesPastType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Detail&gt;„ComplexType: KindOfSalaryType“ &lt;/Detail&gt; [1..unbounded] &lt;/...&gt;</pre> |

## ComplexType: KindOfSalariesTimedPastType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Detail&gt;„ComplexType: KindOfSalaryTimedType“ &lt;/Detail&gt; [1..unbounded] &lt;/...&gt;</pre> |

## ComplexType: KindOfSalariesTimedType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;HourlyPay&gt;„ComplexType: HourlyPayType“ &lt;/HourlyPay&gt; [0..1] &lt;LessonPay&gt;„ComplexType: LessonPayType“ &lt;/LessonPay&gt; [0..1] &lt;Detail&gt;„ComplexType: KindOfSalaryType“ &lt;/Detail&gt; [0..unbounded] &lt;/...&gt;</pre> |

## ComplexType: KindOfSalaryTimedType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | „ComplexType: KindOfSalaryType“   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: KindOfSalaryType“ ] &lt;KindOfSalary&gt;xs:string&lt;/KindOfSalary&gt; [1..1] &lt;Amount&gt;„SimpleType: SalaryAmountType“ &lt;/Amount&gt; [1..1] &lt;StatisticCode&gt;„SimpleType: SalaryBaseCodeType“ &lt;/StatisticCode&gt; [1..1] [ END BASE TYPE ] &lt;BenefitPeriod&gt;„ComplexType: BenefitPeriodType“ &lt;/BenefitPeriod&gt; [0..unbounded] &lt;/...&gt;</pre> |

## ComplexType: KindOfSalaryType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;KindOfSalary&gt;xs:string&lt;/KindOfSalary&gt; [1..1] &lt;Amount&gt; „SimpleType: SalaryAmountType“ &lt;/Amount&gt; [1..1] &lt;StatisticCode&gt; „SimpleType: SalaryBaseCodeType“ &lt;/StatisticCode&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: LessonPayType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;LessonSalary&gt; „SimpleType: SalaryAmountType“ &lt;/LessonSalary&gt; [1..1] &lt;Earnings13th&gt; „SimpleType: PercentType“ &lt;/Earnings13th&gt; [0..1] &lt;VacationAndPublicHolidayCompensation&gt; „SimpleType: PercentType“ &lt;/VacationAndPublicHolidayCompensation&gt; [0..1] &lt;StatisticCode&gt; „SimpleType: SalaryBaseCodeType“ &lt;/StatisticCode&gt; [1..1] &lt;AnnualLessonSalary&gt; „SimpleType: SalaryAmountType“ &lt;/AnnualLessonSalary&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: LessonWagePastType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Salary&gt; „ComplexType: InsuredSalaryLessonPastType“ &lt;/Salary&gt; [1..12] &lt;LessonWageAnnualTotal&gt; „SimpleType: SalaryAmountType“ &lt;/LessonWageAnnualTotal&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: LinkToOldIndemnificationType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;SettlementID&gt; „SimpleType: IDType“ &lt;/SettlementID&gt; [1..1] &lt;IndemnificationID&gt; „SimpleType: IDType“ &lt;/IndemnificationID&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: MonthSalaryType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;HourlyWage&gt; „ComplexType: HourlyWagePastType“ &lt;/HourlyWage&gt; [0..1] &lt;LessonWage&gt; „ComplexType: LessonWagePastType“ &lt;/LessonWage&gt; [0..1] &lt;Allowances&gt; „ComplexType: AllowancesPastType“ &lt;/Allowances&gt; [0..1] &lt;WageReplacementBenefits&gt; „ComplexType: WageReplacementBenefitsPastType“ &lt;/WageReplacementBenefits&gt; [0..1] &lt;PastAnnualTotal&gt; „SimpleType: SalaryAmountType“ &lt;/PastAnnualTotal&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: MonthlyWorkingTimeType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Month&gt;xs:gYearMonth&lt;/Month&gt; [1..1] &lt;HoursPerMonth&gt; „SimpleType: HoursAndIndustryMinutesType“ &lt;/HoursPerMonth&gt; [0..1] &lt;LessonsPerMonth&gt; „SimpleType: HoursAndIndustryMinutesType“ &lt;/LessonsPerMonth&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: MonthlyWorkingTimesType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;MonthlyWorkingTime&gt; „ComplexType: MonthlyWorkingTimeType“ &lt;/MonthlyWorkingTime&gt; [1..12] &lt;/...&gt;</pre> |

## ComplexType: OldNewIDType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Old&gt;„SimpleType: IDType“&lt;/Old&gt;    [1..1] &lt;New&gt;„SimpleType: IDType“&lt;/New&gt;    [1..1] &lt;/...&gt;</pre> |

## ComplexType: PartOfBodyType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; [ START CHOICE ] &lt;UnassignedPart&gt;„SimpleType: BodyPartType“&lt;/UnassignedPart&gt;    [1..1] &lt;Part&gt;„SimpleType: BodyOrientedPartType“&lt;/Part&gt;    [1..1] &lt;Position&gt;„SimpleType: PositionPartType“&lt;/Position&gt;    [1..1] [ END CHOICE ] &lt;/...&gt;</pre> |

## ComplexType: PaymentAddressBaseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;AccountOwner&gt;xs:string&lt;/AccountOwner&gt;    [1..1] &lt;AccountOwnerSpecialAddress&gt;„ComplexType: NameAndAddressType“&lt;/AccountOwnerSpe\ cialAddress&gt;    [0..1] &lt;IBAN&gt;„SimpleType: IBANType“&lt;/IBAN&gt;    [1..1] &lt;FinancialInstitution&gt;„ComplexType: NameAndAddressType“&lt;/FinancialInstituti\ on&gt;    [0..1] &lt;/...&gt;</pre> |

## ComplexType: PaymentAddressType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: PaymentAddressBaseType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: PaymentAddressBaseType“ ] &lt;AccountOwner&gt;xs:string&lt;/AccountOwner&gt;    [1..1] &lt;AccountOwnerSpecialAddress&gt;„ComplexType: NameAndAddressType“&lt;/AccountOwnerSpe\ cialAddress&gt;    [0..1] &lt;IBAN&gt;„SimpleType: IBANType“&lt;/IBAN&gt;    [1..1] &lt;FinancialInstitution&gt;„ComplexType: NameAndAddressType“&lt;/FinancialInstituti\ on&gt;    [0..1] [ END BASE TYPE ] &lt;ValidAsOf&gt;xs:date&lt;/ValidAsOf&gt;    [1..1] &lt;/...&gt;</pre> |

## ComplexType: PaymentType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>  |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt;    [1..1] &lt;StoryID&gt;„SimpleType: IDType“&lt;/StoryID&gt;    [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;Company&gt;„ComplexType: PaymentAddressBaseType“&lt;/Company&gt;    [1..1] &lt;Person&gt;„ComplexType: PaymentAddressType“&lt;/Person&gt;    [1..1] [ END CHOICE ] &lt;/...&gt;</pre> |

## ComplexType: PersonIncidentType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: PersonBaseType“</a>                      |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: PersonBaseType“ ]</pre> |

|  |  |
|--|--|
|  | <pre> &lt;Particulars&gt; „ComplexType: ParticularsType“ &lt;/Particulars&gt; [1..1] &lt;Work&gt; „ComplexType: WorkType“ &lt;/Work&gt; [1..unbounded] [ END BASE TYPE ] &lt;UVG-LAA-Registration&gt; „ComplexType: AccidentRegistrationType“ &lt;/UVG-LAA-Registrati\ on&gt; [0..1] &lt;UVGZ-LAAC-Registration&gt; „ComplexType: AccidentRegistrationType“ &lt;/UVGZ-LAAC- Registration&gt; [0..unbounded] &lt;KTG-AMC-Registration&gt; „ComplexType: IllnessRegistrationType“ &lt;/KTG-AMC-Registrati\ on&gt; [0..1] &lt;/...&gt; </pre> |
|--|--|

## ComplexType: PersonMutatedType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | „ComplexType: StoryBaseType“   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt; „SimpleType: IDType“ &lt;/StoryID&gt; [1..1] [ END BASE TYPE ] &lt;ValidAsOf&gt;xs:date&lt;/ValidAsOf&gt; [1..1] &lt;Lastname&gt;xs:string&lt;/Lastname&gt; [0..1] &lt;Firstname&gt;xs:string&lt;/Firstname&gt; [0..1] &lt;WithdrawalDate&gt;xs:date&lt;/WithdrawalDate&gt; [0..1] &lt;Address&gt; „ComplexType: AddressType“ &lt;/Address&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: ProcessCtrlRequestType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | „ComplexType: StoryBaseType“   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt; „SimpleType: IDType“ &lt;/StoryID&gt; [1..1] [ END BASE TYPE ] &lt;IncidentForCompanyClosed&gt; „ComplexType: IncidentForCompanyClosedType“ &lt;/Incident\ ForCompanyClosed&gt; [0..1] &lt;ChangeIdentities&gt; „ComplexType: ChangeIdentitiesType“ &lt;/ChangeIdentities&gt; [0..1] &lt;Comment&gt; „ComplexType: NotificationsType“ &lt;/Comment&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: ProcessCtrlResponseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | „ComplexType: StoryBaseType“   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt; „SimpleType: IDType“ &lt;/StoryID&gt; [1..1] [ END BASE TYPE ] &lt;ChangeInsuranceCaseID&gt; „ComplexType: OldNewIDType“ &lt;/ChangeInsuranceCa\ seID&gt; [0..1] &lt;InstitutionAddress&gt; „ComplexType: NameAndAddressType“ &lt;/InstitutionAd\ dress&gt; [0..1] &lt;Contact&gt; „ComplexType: ContactType“ &lt;/Contact&gt; [0..1] &lt;AwaitStory&gt; „ComplexType: AwaitPartType“ &lt;/AwaitStory&gt; [0..1] &lt;Comment&gt; „ComplexType: NotificationsType“ &lt;/Comment&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: ReceivedStoriesType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;StoryID&gt; „SimpleType: IDType“ &lt;/StoryID&gt; [0..unbounded] &lt;/...&gt; </pre> |

## ComplexType: ReductionType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;ReasonPartialRelease&gt;xs:string&lt;/ReasonPartialRelease&gt; [0..1] </pre> |

|  |   |
|--|---|
|  | <pre> &lt;ReasonReduction&gt;xs:string&lt;/ReasonReduction&gt;    [0..1] &lt;ReasonOnCompensation&gt;xs:string&lt;/ReasonOnCompensation&gt;    [0..1] &lt;ReasonHospitalAliment&gt;xs:string&lt;/ReasonHospitalAliment&gt;    [0..1] &lt;Percentage&gt;„SimpleType: PercentType“&lt;/Percentage&gt;    [0..1] &lt;CoveringDays&gt;„SimpleType: DaysType“&lt;/CoveringDays&gt;    [0..1] &lt;QuantityOfDays&gt;xs:short&lt;/QuantityOfDays&gt;    [0..1] &lt;Period&gt;„ComplexType: TimePeriodType“&lt;/Period&gt;    [0..1] &lt;ReductionAllowance&gt;„SimpleType: SalaryAmountAbsoluteType“&lt;/ReductionAllowan\ ce&gt;    [0..1] &lt;ReductionAmountFavourInstitution&gt;„SimpleType: SalaryAmountAbsoluteType“&lt;/Reduc\ tionAmountFavourInstitution&gt;    [0..1] &lt;AmountBeforeReductionFavourCompany&gt;„SimpleType: SalaryAmountAbsoluteType“&lt;/ AmountBeforeReductionFavourCompany&gt;    [0..1] &lt;/...&gt; </pre> |
|--|---|

## ComplexType: RegistrationIncidentDateType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; [ START CHOICE ] &lt;Exact&gt;xs:date&lt;/Exact&gt;    [1..1] &lt;Approximate&gt;xs:date&lt;/Approximate&gt;    [1..1] &lt;Clarification&gt;„ComplexType: EmptyType“&lt;/Clarification&gt;    [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |

## ComplexType: RelapseAccidentRegistrationType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;RelapseDate&gt;xs:date&lt;/RelapseDate&gt;    [1..1] &lt;IncidentDate&gt;„ComplexType: RegistrationIncidentDateType“&lt;/IncidentDate&gt;    [1..1] &lt;InsuranceCaseID&gt;„SimpleType: IDType“&lt;/InsuranceCaseID&gt;    [1..1] &lt;/...&gt; </pre> |

## ComplexType: RepaymentType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt;    [1..1] &lt;StoryID&gt;„SimpleType: IDType“&lt;/StoryID&gt;    [1..1] [ END BASE TYPE ] &lt;RepaymentID&gt;„SimpleType: IDType“&lt;/RepaymentID&gt;    [1..1] &lt;LinkToSettlementID&gt;„SimpleType: IDType“&lt;/LinkToSettlementID&gt;    [1..1] &lt;ExpiryDate&gt;xs:dateTime&lt;/ExpiryDate&gt;    [0..1] &lt;Transfer&gt;„ComplexType: TransferType“&lt;/Transfer&gt;    [1..1] &lt;Contact&gt;„ComplexType: ContactType“&lt;/Contact&gt;    [0..1] &lt;/...&gt; </pre> |

## ComplexType: SalaryBasesType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;BasicWages&gt;„SimpleType: SalaryAmountType“&lt;/BasicWages&gt;    [0..1] &lt;FamilyIncomeSupplement&gt;„SimpleType: SalaryAmountType“&lt;/FamilyIncomeSupple\ ment&gt;    [0..1] &lt;VacationAndPublicHolidayCompensation&gt;„SimpleType: SalaryAmountType“&lt;/VacationAnd\ PublicHolidayCompensation&gt;    [0..1] &lt;OtherSupplements&gt;„SimpleType: SalaryAmountType“&lt;/OtherSupplements&gt;    [0..1] &lt;Gratuity&gt;„SimpleType: SalaryAmountType“&lt;/Gratuity&gt;    [0..1] &lt;/...&gt; </pre> |

## ComplexType: SalaryRaiseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt;    [1..1] &lt;StoryID&gt;„SimpleType: IDType“&lt;/StoryID&gt;    [1..1] [ END BASE TYPE ] </pre> |

|  |  |
|--|--|
|  | <pre> &lt;ValidAsOf&gt;xs:date&lt;/ValidAsOf&gt;    [1..1] &lt;Present&gt; „ComplexType: KindOfSalariesTimedType“ &lt;/Present&gt;    [1..1] &lt;ManualAdditions&gt; „ComplexType: KindOfSalaryType“ &lt;/ ManualAdditions&gt;    [0..unbounded] &lt;/...&gt; </pre> |
|--|--|

## ComplexType: SettlementType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt;    [1..1] &lt;StoryID&gt; „SimpleType: IDType“ &lt;/StoryID&gt;    [1..1] [ END BASE TYPE ] &lt;SettlementID&gt; „SimpleType: IDType“ &lt;/SettlementID&gt;    [1..1] &lt;SettlementDate&gt;xs:dateTime&lt;/SettlementDate&gt;    [1..1] &lt;TotalBalance&gt; „ComplexType: TotalBalanceType“ &lt;/TotalBalance&gt;    [1..1] &lt;Coverage&gt; „SimpleType: NotEmptyStringType“ &lt;/Coverage&gt;    [0..1] &lt;AnnualEarnings&gt; „SimpleType: SalaryAmountType“ &lt;/AnnualEarnings&gt;    [0..1] &lt;CoveredDailyAllowance&gt; „SimpleType: SalaryAmountType“ &lt;/CoveredDailyAllowan\ ce&gt;    [0..1] &lt;Contact&gt; „ComplexType: ContactType“ &lt;/Contact&gt;    [0..1] &lt;Indemnifications&gt; „ComplexType: IndemnificationsType“ &lt;/Indemnifications&gt;    [0..1] &lt;Comment&gt; „ComplexType: NotificationsType“ &lt;/Comment&gt;    [0..1] &lt;/...&gt; </pre> |

## ComplexType: SpecialCodeType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt;    [1..1] &lt;StoryID&gt; „SimpleType: IDType“ &lt;/StoryID&gt;    [1..1] [ END BASE TYPE ] &lt;Code&gt; „SimpleType: NotEmptyStringType“ &lt;/Code&gt;    [1..1] &lt;/...&gt; </pre> |

## ComplexType: SpecialCoverageType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>  |
| Translation                 | Spez. Übernahme   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt;    [1..1] &lt;StoryID&gt; „SimpleType: IDType“ &lt;/StoryID&gt;    [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;FamilyMember&gt; „ComplexType: EmptyType“ &lt;/FamilyMember&gt;    [1..1] &lt;OptionalEnterpriserInsurance&gt; „ComplexType: EmptyType“ &lt;/OptionalEnterpriserIn\ surance&gt;    [1..1] &lt;NegotiatedInsurance&gt; „ComplexType: EmptyType“ &lt;/NegotiatedInsurance&gt;    [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |

## ComplexType: StateType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Process&gt; „SimpleType: MainStateType“ &lt;/Process&gt;    [1..1] &lt;Coverage&gt; „SimpleType: CoverageType“ &lt;/Coverage&gt;    [1..1] &lt;SmallRemark&gt; „SimpleType: NotEmptyStringType“ &lt;/SmallRemark&gt;    [0..1] &lt;CrossChannelLinkIDWaitForUse&gt; „SimpleType: IDType“ &lt;/ CrossChannelLinkIDWaitForUse&gt;    [0..unbounded] &lt;/...&gt; </pre> |

## ComplexType: StoriesWithoutDeclarationType

|          |    |
|----------|----|
| Abstract | no |
|----------|----|

|                             |  |
|-----------------------------|--|
| XML Instance Representation | <pre>&lt;...&gt; &lt;DialogMessage&gt;„ComplexType: DialogMessageType“&lt;/DialogMessage&gt; [0..unbounded] &lt;/...&gt;</pre> |
|-----------------------------|--|

## ComplexType: StoryIncidentDateTimeType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; [ START CHOICE ] &lt;ExactDateTime&gt;xs:dateTime&lt;/ExactDateTime&gt; [1..1] &lt;Approximate&gt;xs:date&lt;/Approximate&gt; [1..1] [ END CHOICE ] &lt;/...&gt;</pre> |

## ComplexType: StoryNotificationRequestType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;ResponseStoryID&gt;„SimpleType: IDType“&lt;/ResponseStoryID&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: StoryRelapseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;InsuranceCaseID&gt;„SimpleType: IDType“&lt;/InsuranceCaseID&gt; [1..1] &lt;RelapseDate&gt;xs:date&lt;/RelapseDate&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: TakeWorkUpFullyType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>  |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt;„SimpleType: IDType“&lt;/StoryID&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;TakeWorkUpFullyDate&gt;xs:date&lt;/TakeWorkUpFullyDate&gt; [1..1] &lt;ExpectedLastDayOfIncapacityToWork&gt;xs:date&lt;/ExpectedLastDayOfIncapacityTo\ Work&gt; [1..1] &lt;ExpectedIncapacityToWork&gt;„SimpleType: ExpectedIncapacityToWorkType“&lt;/ExpectedIn\ capacityToWork&gt; [1..1] &lt;NoIncapacityToWork&gt;„ComplexType: EmptyType“&lt;/NoIncapacityToWork&gt; [1..1] [ END CHOICE ] &lt;/...&gt;</pre> |

## ComplexType: TaxAtSourceType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;TaxAtSourceCanton&gt;„SimpleType: CantonAddressType“&lt;/TaxAtSourceCanton&gt; [1..1] &lt;TaxAtSourceMunicipalityID&gt;„SimpleType: MunicipalityIDType“&lt;/TaxAtSourceMunicipa\ lityID&gt; [1..1] &lt;TaxAtSourceCategory&gt;„ComplexType: TaxAtSourceCategoryType“&lt;/TaxAtSourceCatego\ ry&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: TotalBalanceType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; [ START CHOICE ] &lt;FavourCompany&gt;„ComplexType: FavourCompanyType“&lt;/FavourCompany&gt; [1..1] &lt;FavourInstitution&gt;„ComplexType: FavourInstitutionType“&lt;/FavourInstituti\ on&gt; [1..1] &lt;FavourPerson&gt;„SimpleType: SalaryAmountAbsoluteType“&lt;/FavourPerson&gt; [1..1] [ END CHOICE ]</pre> |

|  |        |
|--|--------|
|  | </...> |
|--|--------|

## ComplexType: TransferType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: PaymentAddressBaseType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: PaymentAddressBaseType“ ] &lt;AccountOwner&gt;xs:string&lt;/AccountOwner&gt; [1..1] &lt;AccountOwnerSpecialAddress&gt; „ComplexType: NameAndAddressType“ &lt;/AccountOwnerSpe\ cialAddress&gt; [0..1] &lt;IBAN&gt; „SimpleType: IBANType“ &lt;/IBAN&gt; [1..1] &lt;FinancialInstitution&gt; „ComplexType: NameAndAddressType“ &lt;/FinancialInstituti\ on&gt; [0..1] [ END BASE TYPE ] &lt;Balance&gt; „SimpleType: SalaryAmountType“ &lt;/Balance&gt; [1..1] &lt;Reference&gt;xs:string&lt;/Reference&gt; [1..1] &lt;AccountNumber&gt;xs:string&lt;/AccountNumber&gt; [0..1] &lt;Line&gt;xs:string&lt;/Line&gt; [0..unbounded] &lt;/...&gt;</pre> |

## ComplexType: TreatmentAddressExtensionType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;ZSRNumber&gt;xs:string&lt;/ZSRNumber&gt; [0..1] &lt;UID-BFS&gt; „SimpleType: UID-BFSType“ &lt;/UID-BFS&gt; [0..1] &lt;GLNNumber&gt;xs:string&lt;/GLNNumber&gt; [0..1] &lt;TreatmentInstitution&gt;xs:string&lt;/TreatmentInstitution&gt; [1..1] &lt;Address&gt; „ComplexType: AddressType“ &lt;/Address&gt; [1..1] &lt;EmailAddress&gt; „SimpleType: EmailAddressType“ &lt;/EmailAddress&gt; [0..1] &lt;PhoneNumber&gt;xs:string&lt;/PhoneNumber&gt; [0..1] &lt;MobilePhoneNumber&gt;xs:string&lt;/MobilePhoneNumber&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: TreatmentsType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt; „SimpleType: IDType“ &lt;/StoryID&gt; [1..1] [ END BASE TYPE ] &lt;InitialTreatment&gt; „ComplexType: TreatmentAddressExtensionType“ &lt;/InitialTreat\ ment&gt; [1..1] &lt;FurtherTreatment&gt; „ComplexType: TreatmentAddressExtensionType“ &lt;/ FurtherTreatment&gt; [0..unbounded] &lt;/...&gt;</pre> |

## ComplexType: UVG-LAA-IdentificationType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;InsuranceCompanyName&gt;xs:string&lt;/InsuranceCompanyName&gt; [1..1] [ START CHOICE ] &lt;CustomerIdentity&gt; „SimpleType: NotEmptyStringType“ &lt;/CustomerIdentity&gt; [1..1] [ START CHOICE ] &lt;ContractIdentity&gt; „SimpleType: NotEmptyStringType“ &lt;/ContractIdentity&gt; [1..1] &lt;NoContract&gt; „ComplexType: EmptyType“ &lt;/NoContract&gt; [1..1] [ END CHOICE ] &lt;OrderCustomerIdentity&gt; „ComplexType: EmptyType“ &lt;/OrderCustomerIdentity&gt; [1..1] [ END CHOICE ] &lt;/...&gt;</pre> |

## ComplexType: VacationsType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a> |
| XML Instance Representation | <...>  |



|  |   |
|--|---|
|  | <pre>[ BASE TYPE <a href="#">„ComplexType: StoryBaseType“</a> ] &lt;Creation&gt;xs:date&lt;/Creation&gt; [1..1] &lt;StoryID&gt;<a href="#">„SimpleType: IDType“</a>&lt;/StoryID&gt; [1..1] [ END BASE TYPE ] &lt;Vacation&gt;<a href="#">„ComplexType: TimePeriodType“</a>&lt;/Vacation&gt; [1..unbounded] &lt;/...&gt;</pre> |
|--|---|

## ComplexType: ValidToType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; [ START CHOICE ] &lt;To&gt;xs:date&lt;/To&gt; [1..1] &lt;TakeWorkUpFullyDate&gt;xs:date&lt;/TakeWorkUpFullyDate&gt; [0..1] &lt;Unknown&gt;<a href="#">„ComplexType: EmptyType“</a>&lt;/Unknown&gt; [1..1] [ END CHOICE ] &lt;/...&gt;</pre> |

## ComplexType: WageReplacementBenefitsPastType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Salary&gt;<a href="#">„ComplexType: WageReplacementBenefitsSalaryPastType“</a>&lt;/Salary&gt; [1..12] &lt;WageReplacementBenefitsAnnualTotal&gt;<a href="#">„SimpleType: SalaryAmountType“</a>&lt;/WageReplace\ mentBenefitsAnnualTotal&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: WageReplacementBenefitsSalaryPastType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Month&gt;xs:gYearMonth&lt;/Month&gt; [1..1] &lt;KindOfSalaries&gt;<a href="#">„ComplexType: KindOfSalariesTimedPastType“</a>&lt;/KindOfSala\ ries&gt; [1..1] &lt;Total&gt;<a href="#">„SimpleType: SalaryAmountType“</a>&lt;/Total&gt; [1..1] &lt;/...&gt;</pre> |

## SimpleType: ActivityAtTimeOfAccidentType

|              |  |
|--------------|--|
| Translation  | Tätigkeit  |
| Content type | Restriction  |
| Base type    | xs:NMTOKEN   |
| Facets       | <ul style="list-style-type: none"> <li>• Enum: skiing</li> <li>• Enum: snowboarding</li> <li>• Enum: footballOrsoccer</li> <li>• Enum: otherBallGames</li> <li>• Enum: jogging</li> <li>• Enum: fitnessTraining</li> <li>• Enum: bathingOrSwimming</li> <li>• Enum: hiking</li> <li>• Enum: walking</li> <li>• Enum: eBiking</li> <li>• Enum: bicycleOrMBT</li> <li>• Enum: carDriving</li> <li>• Enum: motorbiking</li> <li>• Enum: housekeepingWork</li> <li>• Enum: doltYourselfOrMaintenanceWork</li> <li>• Enum: gardeningOrForestAndAgricultureWork</li> <li>• Enum: brawlingOrFightingOrViolence</li> <li>• Enum: otherSport</li> <li>• Enum: otherActivitiesAtHome</li> <li>• Enum: otherActivities</li> </ul> |

### SimpleType: AddressingDomainType

|                   |   |
|-------------------|---|
| Translation       | Domäne  |
| Short description | Domäne  |
| Content type      | Restriction   |
| Base type         | xs:NMTOKEN  |
| Facets            | <ul style="list-style-type: none"> <li>• Enum: UVG-LAA</li> <li>• Enum: UVGZ-LAAC</li> <li>• Enum: KTG-AMC</li> </ul> |

### SimpleType: BodyOrientedPartType

|              |   |
|--------------|---|
| Content type | Restriction   |
| Base type    | xs:NMTOKEN  |
| Facets       | <ul style="list-style-type: none"> <li>• Enum: eye_12</li> <li>• Enum: pelvis_40</li> <li>• Enum: elbow_32</li> <li>• Enum: finger_36</li> <li>• Enum: metatarsusWithoutToes_54</li> <li>• Enum: ankle_53</li> <li>• Enum: face_11</li> <li>• Enum: metacarpusWithoutFingers_35</li> <li>• Enum: wrist_34</li> <li>• Enum: hipJoint_41</li> <li>• Enum: jaw_16</li> <li>• Enum: knee_51</li> <li>• Enum: groin_43</li> <li>• Enum: lungsInclusiveAirways_62</li> <li>• Enum: multipleAreasUpperExtremities_37</li> <li>• Enum: multipleAreasLowerExtremities_56</li> <li>• Enum: kidney_65</li> <li>• Enum: upperArm_31</li> <li>• Enum: thigh_50</li> <li>• Enum: earHearing_14</li> <li>• Enum: rear_21</li> <li>• Enum: shoulder_30</li> <li>• Enum: thorax_23</li> <li>• Enum: lowerArm_33</li> <li>• Enum: lowerThigh_52</li> <li>• Enum: teeth_15</li> <li>• Enum: toes_55</li> </ul> |

### SimpleType: BodyPartType

|              |  |
|--------------|--|
| Content type | Restriction  |
| Base type    | xs:NMTOKEN   |
| Facets       | <ul style="list-style-type: none"> <li>• Enum: abdomen_459</li> <li>• Enum: bladder_669</li> <li>• Enum: thoracicSpine_259</li> <li>• Enum: entireBodySystemicEffect_819</li> <li>• Enum: genitals_449</li> <li>• Enum: neck_209</li> <li>• Enum: cervicalSpine_249</li> </ul> |

|  |  |
|--|--|
|  | <ul style="list-style-type: none"> <li>• Enum: heart_619</li> <li>• Enum: otherInternalInjuries_609</li> <li>• Enum: liver_639</li> <li>• Enum: lumbarSpine_269</li> <li>• Enum: spleen_649</li> <li>• Enum: nose_139</li> <li>• Enum: multipleInjuries_709</li> <li>• Enum: skullBrain_109</li> <li>• Enum: shockPsych_809</li> <li>• Enum: coccyxButtocks_429</li> </ul> |
|--|--|

### SimpleType: ContentTypeType

|                   |   |
|-------------------|---|
| Translation       | Attachment Typ  |
| Short description | Arbeitsunfähigkeitsbescheinigung, Korrespondenz, Rechnung und Übriges   |
| Content type      | Restriction   |
| Base type         | xs:NMTOKEN  |
| Facets            | <ul style="list-style-type: none"> <li>• Enum: certificateOfIncapacityToWork</li> <li>• Enum: correspondence</li> <li>• Enum: invoice</li> <li>• Enum: other</li> </ul> |

### SimpleType: CoverageType

|                   |  |
|-------------------|--|
| Translation       | Zustand der Deckung  |
| Short description | Zustand der Deckung  |
| Content type      | Restriction  |
| Base type         | xs:NMTOKEN   |
| Facets            | <ul style="list-style-type: none"> <li>• Enum: yes</li> <li>• Enum: no</li> <li>• Enum: annulled</li> <li>• Enum: unknown</li> </ul> |

### SimpleType: DaysType

|              |   |
|--------------|---|
| Content type | Restriction   |
| Base type    | xs:decimal  |
| Facets       | <ul style="list-style-type: none"> <li>• MinInclusive: -9999.99</li> <li>• MaxInclusive: 9999.99</li> </ul> |

### SimpleType: DigitalizationScopeType

|              |   |
|--------------|---|
| Content type | Restriction   |
| Base type    | xs:NMTOKEN  |
| Facets       | <ul style="list-style-type: none"> <li>• Enum: Basic</li> <li>• Enum: BasicCoverage</li> <li>• Enum: BasicCoverageSettlement</li> <li>• Enum: BasicDialog</li> <li>• Enum: BasicCoverageDialog</li> </ul> |

|  |                                       |
|--|---------------------------------------|
|  | • Enum: BasicCoverageSettlementDialog |
|--|---------------------------------------|

### SimpleType: ExpectedIncapacityToWorkType

|              |   |
|--------------|---|
| Content type | Restriction   |
| Base type    | xs:NMTOKEN  |
| Facets       | <ul style="list-style-type: none"> <li>• Enum: to30Days</li> <li>• Enum: moreThan30Days</li> <li>• Enum: unknown</li> </ul> |

### SimpleType: HospitalCostDeductionEnumType

|                   |  |
|-------------------|--|
| Translation       | Attachment Typ   |
| Short description | Arbeitsunfähigkeitsbescheinigung, Korrespondenz, Rechnung und Übriges  |
| Content type      | Restriction  |
| Base type         | xs:NMTOKEN   |
| Facets            | <ul style="list-style-type: none"> <li>• Enum: 10PercentageOr10CHF</li> <li>• Enum: 20PercentageOr20CHF</li> </ul> |

### SimpleType: HoursAndIndustryMinutesType

|              |   |
|--------------|---|
| Content type | Restriction   |
| Base type    | xs:decimal  |
| Facets       | <ul style="list-style-type: none"> <li>• Pattern: [0-9]+\.[0-9]{2}</li> </ul> |

### SimpleType: IBANType

|                       |   |
|-----------------------|---|
| Translation           | IBAN Nummer   |
| Short description     | IBAN Nummer   |
| Technical description | <p>Regex Pattern Sample<br/> <a href="https://www.six-interbank-clearing.com/de/home/standardization/iso-payments/customer-bank/implementation-guidelines.html">https://www.six-interbank-clearing.com/de/home/standardization/iso-payments/customer-bank/implementation-guidelines.html</a><br/>           implementation-guidelines-ct-examples.zip - simpleType<br/>           name="IBAN2007Identifier" / string - pattern<br/>           value="[A-Z]{2,2}[0-9]{2,2}[a-zA-Z0-9]{1,30}"<br/> <a href="https://www.w3.org/TR/xmlschema-2/">https://www.w3.org/TR/xmlschema-2/</a> - F Regular Expressions - S{n,m}<br/>           = S{n}<br/>           and n=m - All strings in L(S{n,n}). ( All sequences of exactly n strings from L(S) )</p> |
| Content type          | Restriction   |
| Base type             | xs:string   |
| Facets                | <ul style="list-style-type: none"> <li>• Pattern: [A-Z]{2,2}[0-9]{2,2}[a-zA-Z0-9]{1,30}</li> </ul>  |

### SimpleType: MainStateType

|              |  |
|--------------|--|
| Content type | Restriction  |
| Base type    | xs:NMTOKEN   |
| Facets       | <ul style="list-style-type: none"> <li>• Enum: registrationReceived</li> </ul> |

|  |  |
|--|--|
|  | <ul style="list-style-type: none"> <li>• Enum: checkingDeclaration</li> <li>• Enum: verifyingClaim</li> <li>• Enum: performingEntitlement</li> <li>• Enum: communicationStopped</li> </ul> |
|--|--|

### SimpleType: OtherInsurancePredefinedType

|              |   |
|--------------|---|
| Translation  | Definierte andere Versicherer   |
| Content type | Restriction   |
| Base type    | xs:NMTOKEN  |
| Facets       | <ul style="list-style-type: none"> <li>• Enum: suva</li> <li>• Enum: otherCompulsoryAccidentInsurance</li> <li>• Enum: KTG-AMC</li> <li>• Enum: IV-AI</li> <li>• Enum: AHV-AVS</li> <li>• Enum: occupationalPreventionEstablishment</li> <li>• Enum: militaryInsurance</li> <li>• Enum: unemploymentInsurance</li> <li>• Enum: maternetyldemnity</li> </ul> |

### SimpleType: PositionPartType

|              |   |
|--------------|---|
| Content type | Restriction   |
| Base type    | xs:NMTOKEN  |
| Facets       | <ul style="list-style-type: none"> <li>• Enum: left_1</li> <li>• Enum: right_2</li> <li>• Enum: both_3</li> <li>• Enum: unassigned_9</li> </ul> |

### SimpleType: ReductionReasonEnumType

|                   |  |
|-------------------|--|
| Translation       | Attachment Typ   |
| Short description | Arbeitsunfähigkeitsbescheinigung, Korrespondenz, Rechnung und Übriges  |
| Content type      | Restriction  |
| Base type         | xs:NMTOKEN   |
| Facets            | <ul style="list-style-type: none"> <li>• Enum: provisional</li> <li>• Enum: partialApproval</li> <li>• Enum: legalReduction</li> <li>• Enum: contractualReduction</li> <li>• Enum: overcompensation</li> </ul> |

### SimpleType: SalaryBaseCodeType

|              |  |
|--------------|--|
| Content type | Restriction  |
| Base type    | xs:NMTOKEN   |
| Facets       | <ul style="list-style-type: none"> <li>• Enum: BasicWages</li> <li>• Enum: FamilyIncomeSupplement</li> <li>• Enum: VacationAndPublicHolidayCompensation</li> <li>• Enum: OtherSupplements</li> <li>• Enum: Gratuity</li> </ul> |

|  |   |
|--|---|
|  | <ul style="list-style-type: none"> <li>Enum: WageReplacement</li> </ul> |
|--|---|

### SimpleType: SupportedMinorSchemaVersionAttributeType

|                       |   |
|-----------------------|---|
| Translation           | Unterstützte Minor Schema Versionen   |
| Short description     | Hier werden die kompatiblen Minor Versionen aufgelistet   |
| Technical description | pattern mehrere Vorkommastellen ermöglichen und führenden Nullen ausschliessen [0-9]\.[0-9] = Major eine Stelle, Punkt, Minor eine Stelle oder [1-9]\d*\.[0-9] = beliebige Major ohne führende Nullen, Punkt, eine Stelle Minor |
| Content type          | Restriction   |
| Base type             | xs:decimal  |
| Facets                | <ul style="list-style-type: none"> <li>Pattern: [0-9]\.[0-9]  [1-9]\d*\.[0-9]</li> <li>Enum: 0.0</li> </ul>   |

### SimpleType: TypeOfInjuryType

|              |   |
|--------------|---|
| Content type | Restriction   |
| Base type    | xs:NMTOKEN  |
| Facets       | <ul style="list-style-type: none"> <li>Enum: bite_01</li> <li>Enum: fracture_02</li> <li>Enum: inflammation_03</li> <li>Enum: contusion_04</li> <li>Enum: bruise_05</li> <li>Enum: chap_06</li> <li>Enum: cut_07</li> <li>Enum: shot_08</li> <li>Enum: swelling_09</li> <li>Enum: graze_10</li> <li>Enum: compressionSpine_11</li> <li>Enum: sting_12</li> <li>Enum: tearFissure_13</li> <li>Enum: burn_14</li> <li>Enum: sprainTwist_15</li> <li>Enum: poisoning_16</li> <li>Enum: causticity_17</li> <li>Enum: pulledMuscleLigament_18</li> <li>Enum: foreignObject_20</li> <li>Enum: dislocation_21</li> </ul> |

## G.5 Schema Dokumentation SwissdecComponents.xsd

|                     |  |
|---------------------|--|
| Target Namespace    | urn:ch:swissdec:basis:v1:20260306:components   |
| Declared Namespaces | <ul style="list-style-type: none"> <li>ep : urn:ch:swissdec:basis:v1:20260306:components</li> <li>xs : http://www.w3.org/2001/XMLSchema</li> </ul> |
| Version             | 0.0  |

### ComplexType: AccessInformationType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Url&gt;xs:string&lt;/Url&gt;    [1..1]   &lt;ExpiryDate&gt;xs:dateTime&lt;/ExpiryDate&gt;    [1..1]</pre> |

|  |        |
|--|--------|
|  | </...> |
|--|--------|

## ComplexType: AddresseeJobType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: AddresseeType“</a>  |
| Translation                 | Adressierung der Endempfänger   |
| Short description           | Adressierung der Endempfänger   |
| Technical description       | Auflistung der Endempfänger, die die übermittelten Daten erhalten sollen.   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE <a href="#">„ComplexType: AddresseeType“</a> ] &lt;AddresseeIdentification&gt; <a href="#">„SimpleType: IDType“</a> &lt;/AddresseeIdentification&gt; [1..1] [ END BASE TYPE ] &lt;ProcessByDistributor&gt; <a href="#">„SimpleType: SimpleBooleanType“</a> &lt;/ProcessByDistributor&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: AddresseeResponseContextType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: ResponseContextType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE <a href="#">„ComplexType: ResponseContextType“</a> ] &lt;UserAgent&gt; <a href="#">„ComplexType: UserAgentType“</a> &lt;/UserAgent&gt; [1..1] &lt;InstitutionName&gt;xs:string&lt;/InstitutionName&gt; [1..1] &lt;TransmissionDate&gt;xs:dateTime&lt;/TransmissionDate&gt; [1..1] &lt;ResponseID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/ResponseID&gt; [1..1] &lt;RequestID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/RequestID&gt; [1..1] [ END BASE TYPE ] &lt;ProducerResponseNotifications&gt; <a href="#">„ComplexType: FeedbackNotificationsType“</a> &lt;/ProducerResponseNotifications&gt; [1..1] &lt;Warning&gt; <a href="#">„ComplexType: NotificationsType“</a> &lt;/Warning&gt; [0..1] &lt;Info&gt; <a href="#">„ComplexType: NotificationsType“</a> &lt;/Info&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: AddresseeResponseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;AddresseeContext&gt; <a href="#">„ComplexType: AddresseeResponseContextType“</a> &lt;/AddresseeContext&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: AddresseeType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Short description           | Informationen zum Adressaten   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;AddresseeIdentification&gt; <a href="#">„SimpleType: IDType“</a> &lt;/AddresseeIdentification&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: AnswerAmountType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Default&gt; <a href="#">„SimpleType: SalaryAmountType“</a> &lt;/Default&gt; [0..1] &lt;Value&gt; <a href="#">„SimpleType: SalaryAmountType“</a> &lt;/Value&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: AnswerBooleanType

|          |    |
|----------|----|
| Abstract | no |
|----------|----|

|                             |   |
|-----------------------------|---|
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Default&gt;xs:boolean&lt;/Default&gt;    [0..1]   &lt;Value&gt;xs:boolean&lt;/Value&gt;      [0..1] &lt;/...&gt;</pre> |
|-----------------------------|---|

### ComplexType: AnswerDateTimeType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Default&gt;xs:dateTime&lt;/Default&gt;    [0..1]   &lt;Value&gt;xs:dateTime&lt;/Value&gt;      [0..1] &lt;/...&gt;</pre> |

### ComplexType: AnswerDateType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Default&gt;xs:date&lt;/Default&gt;        [0..1]   &lt;Value&gt;xs:date&lt;/Value&gt;          [0..1] &lt;/...&gt;</pre> |

### ComplexType: AnswerDoubleType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Default&gt;xs:double&lt;/Default&gt;      [0..1]   &lt;Value&gt;xs:double&lt;/Value&gt;        [0..1] &lt;/...&gt;</pre> |

### ComplexType: AnswerIntegerType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Default&gt;xs:integer&lt;/Default&gt;    [0..1]   &lt;Value&gt;xs:integer&lt;/Value&gt;      [0..1] &lt;/...&gt;</pre> |

### ComplexType: AnswerStringType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Default&gt;xs:string&lt;/Default&gt;     [0..1]   &lt;Value&gt;xs:string&lt;/Value&gt;        [0..1] &lt;/...&gt;</pre> |

### ComplexType: AnswerYesNoUnknownType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Default&gt;_SimpleType: YesNoUnknownType_&lt;/Default&gt;    [0..1]   &lt;Value&gt;_SimpleType: YesNoUnknownType_&lt;/Value&gt;        [0..1] &lt;/...&gt;</pre> |

### ComplexType: AwaitPartControlType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;ExpirationDate&gt;xs:dateTime&lt;/ExpirationDate&gt;    [0..1] &lt;/...&gt;</pre> |

### ComplexType: BooleanStoryType

|          |    |
|----------|----|
| Abstract | no |
|----------|----|



|                             |  |
|-----------------------------|--|
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt;„SimpleType: IDType“&lt;/StoryID&gt; [1..1] [ END BASE TYPE ] &lt;Valid&gt;xs:boolean&lt;/Valid&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: CaseContextBaseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;ReceivedStoryIDs&gt;„ComplexType: ReceivedStoriesType“&lt;/ReceivedStoryIDs&gt; [0..1] &lt;SuppressedSenderStoryIDs&gt;„ComplexType: ReceivedStoriesType“&lt;/SuppressedSenderSto\ ryIDs&gt; [0..1] &lt;SuppressedInstitutionStoryIDs&gt;„ComplexType: ReceivedStoriesType“&lt;/SuppressedIn\ stitutionStoryIDs&gt; [0..1] &lt;Warning&gt;„ComplexType: StoryNotificationsType“&lt;/Warning&gt; [0..1] &lt;Info&gt;„ComplexType: StoryNotificationsType“&lt;/Info&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: CaseContextType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: CaseContextBaseType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: CaseContextBaseType“ ] &lt;ReceivedStoryIDs&gt;„ComplexType: ReceivedStoriesType“&lt;/ReceivedStoryIDs&gt; [0..1] &lt;SuppressedSenderStoryIDs&gt;„ComplexType: ReceivedStoriesType“&lt;/SuppressedSenderS\ toryIDs&gt; [0..1] &lt;SuppressedInstitutionStoryIDs&gt;„ComplexType: ReceivedStoriesType“&lt;/SuppressedIn\ stitutionStoryIDs&gt; [0..1] &lt;Warning&gt;„ComplexType: StoryNotificationsType“&lt;/Warning&gt; [0..1] &lt;Info&gt;„ComplexType: StoryNotificationsType“&lt;/Info&gt; [0..1] [ END BASE TYPE ] &lt;Credentials&gt;„ComplexType: CredentialsType“&lt;/Credentials&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: CheckInteroperabilityRequestType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;UserAgent&gt;„ComplexType: UserAgentType“&lt;/UserAgent&gt; [1..1] &lt;UmlautString&gt;xs:string&lt;/UmlautString&gt; [1..1] &lt;FirstOperand&gt;„SimpleType: SalaryAmountType“&lt;/FirstOperand&gt; [1..1] &lt;SecondOperand&gt;„SimpleType: SalaryAmountType“&lt;/SecondOperand&gt; [1..1] &lt;SystemDateTime&gt;xs:dateTime&lt;/SystemDateTime&gt; [1..1] &lt;MonitoringID&gt;„SimpleType: MonitoringIDType“&lt;/MonitoringID&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: CheckInteroperabilityResponseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;UserAgent&gt;„ComplexType: UserAgentType“&lt;/UserAgent&gt; [1..1] &lt;UmlautStringIsCorrect&gt;„SimpleType: SimpleBooleanType“&lt;/UmlautStringIsCor\ rect&gt; [1..1] &lt;FirstOperandIsCorrect&gt;„SimpleType: SimpleBooleanType“&lt;/FirstOperandIsCor\ rect&gt; [1..1] &lt;UmlautString&gt;xs:string&lt;/UmlautString&gt; [1..1] &lt;AdditionResult&gt;„SimpleType: SalaryAmountType“&lt;/AdditionResult&gt; [1..1] &lt;SubtractionResult&gt;„SimpleType: SalaryAmountType“&lt;/SubtractionResult&gt; [1..1] &lt;SystemDateTime&gt;xs:dateTime&lt;/SystemDateTime&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: CompanyUIDType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;UID-BFS&gt;„ComplexType: UID-BFS-UnknownType“&lt;/UID-BFS&gt; [1..1] &lt;DelegateUID-BFS&gt;„SimpleType: UID-BFSType“&lt;/DelegateUID-BFS&gt; [0..1] &lt;/...&gt;</pre> |

|  |        |
|--|--------|
|  | </...> |
|--|--------|

## ComplexType: CompletionType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>   |
| Short description           | Informationen zur Freigabe der Daten.  |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE <a href="#">„ComplexType: StoryBaseType“</a> ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt;<a href="#">„SimpleType: IDType“</a>&lt;/StoryID&gt; [1..1] [ END BASE TYPE ] &lt;Completion&gt;<a href="#">„ComplexType: AccessInformationType“</a>&lt;/Completion&gt; [1..1] &lt;Credentials&gt;<a href="#">„ComplexType: ResultType“</a>&lt;/Credentials&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: ConsumerFaultType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: ResponseType“</a>  |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE <a href="#">„ComplexType: ResponseType“</a> ] &lt;ResponseContext&gt;<a href="#">„ComplexType: ResponseContextType“</a>&lt;/ResponseContext&gt; [1..1] [ END BASE TYPE ] &lt;FaultState&gt;<a href="#">„ComplexType: FaultStateType“</a>&lt;/FaultState&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: CredentialsType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Translation                 | Berechtigungsnachweis   |
| Short description           | Berechtigungsnachweis   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Key&gt;xs:string&lt;/Key&gt; [1..1] &lt;Password&gt;xs:string&lt;/Password&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: DialogMessageType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>   |
| Technical description       | Semistrukturierte Daten für Dialog zwischen den Fall-Teilnehmern.  |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE <a href="#">„ComplexType: StoryBaseType“</a> ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt;<a href="#">„SimpleType: IDType“</a>&lt;/StoryID&gt; [1..1] [ END BASE TYPE ] &lt;StandardDialogID&gt;<a href="#">„SimpleType: StandardFormIDType“</a>&lt;/StandardDialogID&gt; [1..1] &lt;Previous&gt;<a href="#">„ComplexType: PreviousType“</a>&lt;/Previous&gt; [0..1] &lt;Title&gt;<a href="#">„SimpleType: IDType“</a>&lt;/Title&gt; [0..1] &lt;Description&gt;xs:token&lt;/Description&gt; [0..1] &lt;Paragraph&gt;<a href="#">„ComplexType: ParagraphType“</a>&lt;/Paragraph&gt; [1..unbounded] &lt;Section&gt;<a href="#">„ComplexType: SectionType“</a>&lt;/Section&gt; [0..unbounded] &lt;/...&gt;</pre> |

## ComplexType: DistributorRequestContextType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;UserAgent&gt;<a href="#">„ComplexType: UserAgentType“</a>&lt;/UserAgent&gt; [1..1] &lt;TransmissionDate&gt;xs:dateTime&lt;/TransmissionDate&gt; [1..1] &lt;DistributorRequestID&gt;<a href="#">„SimpleType: IDType“</a>&lt;/DistributorRequestID&gt; [1..1] &lt;VersionMappingFrom&gt;<a href="#">„ComplexType: VersionMappingFromType“</a>&lt;/VersionMappingFrom&gt; [0..1] &lt;ProducerSecurityTokens&gt;<a href="#">„ComplexType: ProducerSecurityTokensType“</a>&lt;/ProducerSecurityTokens&gt; [1..1]</pre> |

|  |   |
|--|---|
|  | <pre> &lt;SignatureCertificateUID-BFS&gt; <a href="#">„ComplexType: SignatureCertificateUID-BFSType“</a> &lt;/ SignatureCertificateUID-BFS&gt; [0..1] &lt;ProducerResponseNotifications&gt; <a href="#">„ComplexType: FeedbackNotificationsType“</a> &lt;/Producer\ ResponseNotifications&gt; [1..1] &lt;ConsumerNotifications&gt; <a href="#">„ComplexType: FeedbackNotificationsType“</a> &lt;/ConsumerNotifica\ tions&gt; [0..1] &lt;/...&gt; </pre> |
|--|---|

## ComplexType: DuplicateType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;LastDistributorRequestID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/LastDistributorRequestID&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: EmptyType

|                             |                                       |
|-----------------------------|---------------------------------------|
| Abstract                    | no                                    |
| XML Instance Representation | <pre> &lt;...&gt; &lt;/...&gt; </pre> |

## ComplexType: ErrorResponseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | Fehler   |
| Short description           | Die Verarbeitung war nicht erfolgreich und es sind Fehler aufgetreten.   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;EndUserInformation&gt;xs:string&lt;/EndUserInformation&gt; [1..1] &lt;DetailInformation&gt;xs:string&lt;/DetailInformation&gt; [0..1] [ START CHOICE ] &lt;FaultInformation&gt; <a href="#">„ComplexType: ConsumerFaultType“</a> &lt;/FaultInformation&gt; [1..1] &lt;PlannedMaintenance&gt; <a href="#">„ComplexType: PlannedMaintenanceWindowType“</a> &lt;/PlannedMainten\ ance&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |

## ComplexType: FaultStateType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | Fehler Status  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Code&gt; <a href="#">„SimpleType: FaultCodeType“</a> &lt;/Code&gt; [1..1] &lt;Error&gt; <a href="#">„ComplexType: NotificationsType“</a> &lt;/Error&gt; [0..1] &lt;Warning&gt; <a href="#">„ComplexType: NotificationsType“</a> &lt;/Warning&gt; [0..1] &lt;Info&gt; <a href="#">„ComplexType: NotificationsType“</a> &lt;/Info&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: FaultType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: ResponseType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE <a href="#">„ComplexType: ResponseType“</a> ] &lt;ResponseContext&gt; <a href="#">„ComplexType: ResponseContextType“</a> &lt;/ResponseContext&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;Fault&gt; <a href="#">„ComplexType: FaultStateType“</a> &lt;/Fault&gt; [1..1] &lt;PlannedMaintenance&gt; <a href="#">„ComplexType: PlannedMaintenanceWindowType“</a> &lt;/PlannedMainten\ ance&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |

## ComplexType: FeedbackNotificationsType

|                             |                          |
|-----------------------------|--------------------------|
| Abstract                    | no                       |
| XML Instance Representation | <pre> &lt;...&gt; </pre> |

|  |   |
|--|---|
|  | <pre>&lt;Warning&gt;„ComplexType: NotificationType“ &lt;/Warning&gt; [0..unbounded] &lt;Info&gt;„ComplexType: NotificationType“ &lt;/Info&gt; [0..unbounded] &lt;/...&gt;</pre> |
|--|---|

## ComplexType: GetStatusRequestType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: RequestType“</a>  |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: RequestType“ ] &lt;RequestContext&gt;„ComplexType: RequestContextType“ &lt;/RequestContext&gt; [1..1] [ END BASE TYPE ] &lt;JobKey&gt;„SimpleType: JobKeyType“ &lt;/JobKey&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: GetStatusResponseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: ResponseType“</a>  |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: ResponseType“ ] &lt;ResponseContext&gt;„ComplexType: ResponseContextType“ &lt;/ResponseContext&gt; [1..1] [ END BASE TYPE ] &lt;JobFinished&gt;„SimpleType: SimpleBooleanType“ &lt;/JobFinished&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: InitialResponseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: ResponseType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: ResponseType“ ] &lt;ResponseContext&gt;„ComplexType: ResponseContextType“ &lt;/ResponseContext&gt; [1..1] [ END BASE TYPE ] &lt;JobKey&gt;„SimpleType: JobKeyType“ &lt;/JobKey&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: MessageType

|                             |                                     |
|-----------------------------|-------------------------------------|
| Abstract                    | no                                  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;/...&gt;</pre> |

## ComplexType: NotificationType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | Meldung  |
| Short description           | Meldung  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;QualityLevel&gt;„SimpleType: QualityLevelType“ &lt;/QualityLevel&gt; [1..1] &lt;DescriptionCode&gt;„SimpleType: DescriptionCodeType“ &lt;/DescriptionCode&gt; [1..1] &lt;Description&gt;xs:string&lt;/Description&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: NotificationsStoryType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>  |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt;„SimpleType: IDType“ &lt;/StoryID&gt; [1..1] [ END BASE TYPE ]</pre> |

|  |  |
|--|--|
|  | <code>&lt;Notifications&gt; <a href="#">„ComplexType: NotificationsType“</a> &lt;/Notifications&gt; [1..1]</code><br><code>&lt;/...&gt;</code> |
|--|--|

## ComplexType: NotificationsType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;Notification&gt; <a href="#">„ComplexType: NotificationType“</a> &lt;/Notification&gt; [1..unbounded]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: ParagraphAnswerType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>[ START CHOICE ]</code><br><code>&lt;String&gt; <a href="#">„ComplexType: AnswerStringType“</a> &lt;/String&gt; [1..1]</code><br><code>&lt;Integer&gt; <a href="#">„ComplexType: AnswerIntegerType“</a> &lt;/Integer&gt; [1..1]</code><br><code>&lt;Double&gt; <a href="#">„ComplexType: AnswerDoubleType“</a> &lt;/Double&gt; [1..1]</code><br><code>&lt;Boolean&gt; <a href="#">„ComplexType: AnswerBooleanType“</a> &lt;/Boolean&gt; [1..1]</code><br><code>&lt;Date&gt; <a href="#">„ComplexType: AnswerDateType“</a> &lt;/Date&gt; [1..1]</code><br><code>&lt;DateTime&gt; <a href="#">„ComplexType: AnswerDateTimeType“</a> &lt;/DateTime&gt; [1..1]</code><br><code>&lt;YesNoUnknown&gt; <a href="#">„ComplexType: AnswerYesNoUnknownType“</a> &lt;/YesNoUnknown&gt; [1..1]</code><br><code>&lt;Amount&gt; <a href="#">„ComplexType: AnswerAmountType“</a> &lt;/Amount&gt; [1..1]</code><br><code>[ END CHOICE ]</code><br><code>&lt;Problem&gt; <a href="#">„SimpleType: NotEmptyStringType“</a> &lt;/Problem&gt; [0..1]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: ParagraphType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;ID&gt;xs:short&lt;/ID&gt; [1..1]</code><br><code>&lt;Label&gt;xs:token&lt;/Label&gt; [1..1]</code><br><code>[ START CHOICE ]</code><br><code>&lt;Value&gt; <a href="#">„ComplexType: ParagraphValueType“</a> &lt;/Value&gt; [0..1]</code><br><code>&lt;Answer&gt; <a href="#">„ComplexType: ParagraphAnswerType“</a> &lt;/Answer&gt; [0..1]</code><br><code>[ END CHOICE ]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: ParagraphValueType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>[ START CHOICE ]</code><br><code>&lt;String&gt;xs:string&lt;/String&gt; [1..1]</code><br><code>&lt;Integer&gt;xs:integer&lt;/Integer&gt; [1..1]</code><br><code>&lt;Double&gt;xs:double&lt;/Double&gt; [1..1]</code><br><code>&lt;Boolean&gt;xs:boolean&lt;/Boolean&gt; [1..1]</code><br><code>&lt;Date&gt;xs:date&lt;/Date&gt; [1..1]</code><br><code>&lt;DateTime&gt;xs:dateTime&lt;/DateTime&gt; [1..1]</code><br><code>&lt;YesNoUnknown&gt; <a href="#">„SimpleType: YesNoUnknownType“</a> &lt;/YesNoUnknown&gt; [1..1]</code><br><code>&lt;Amount&gt; <a href="#">„SimpleType: SalaryAmountType“</a> &lt;/Amount&gt; [1..1]</code><br><code>[ END CHOICE ]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: PingConsumerRequestType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <code>&lt;...&gt;</code><br><code>&lt;UserAgent&gt; <a href="#">„ComplexType: UserAgentType“</a> &lt;/UserAgent&gt; [1..1]</code><br><code>&lt;Timestamp&gt;xs:dateTime&lt;/Timestamp&gt; [1..1]</code><br><code>&lt;NextCheck&gt;xs:dateTime&lt;/NextCheck&gt; [1..1]</code><br><code>&lt;RegisteredMaintenance&gt; <a href="#">„ComplexType: RegisteredMaintenanceType“</a> &lt;/RegisteredMain\</code><br><code>tenance&gt; [0..1]</code><br><code>&lt;/...&gt;</code> |

## ComplexType: PingConsumerResponseType

|                             |                          |
|-----------------------------|--------------------------|
| Abstract                    | no                       |
| XML Instance Representation | <code>&lt;...&gt;</code> |

|  |   |
|--|---|
|  | <pre> &lt;UserAgent&gt; „ComplexType: UserAgentType“ &lt;/UserAgent&gt;    [1..1] &lt;Timestamp&gt;xs:dateTime&lt;/Timestamp&gt;    [1..1] [ START CHOICE ]   &lt;PlannedMaintenance&gt; „ComplexType: PlannedMaintenanceType“ &lt;/PlannedMaintenance&gt;    [1..1]   &lt;NoPlannedMaintenance&gt; „ComplexType: EmptyType“ &lt;/NoPlannedMaintenance&gt;    [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |
|--|---|

### ComplexType: PingRequestType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;UserAgent&gt; „ComplexType: UserAgentType“ &lt;/UserAgent&gt;    [1..1] &lt;SystemDateTime&gt;xs:dateTime&lt;/SystemDateTime&gt;    [1..1] &lt;MonitoringID&gt; „SimpleType: MonitoringIDType“ &lt;/MonitoringID&gt;    [0..1] &lt;/...&gt; </pre> |

### ComplexType: PingResponseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;UserAgent&gt; „ComplexType: UserAgentType“ &lt;/UserAgent&gt;    [1..1] &lt;SystemDateTime&gt;xs:dateTime&lt;/SystemDateTime&gt;    [1..1] &lt;/...&gt; </pre> |

### ComplexType: PlannedMaintenanceType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Start&gt;xs:dateTime&lt;/Start&gt;    [1..1] &lt;End&gt;xs:dateTime&lt;/End&gt;    [1..1] &lt;ProducerMessages&gt; „ComplexType: ProducerMessagesType“ &lt;/ProducerMessages&gt;    [1..1] &lt;/...&gt; </pre> |

### ComplexType: PlannedMaintenanceWindowType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Start&gt;xs:dateTime&lt;/Start&gt;    [1..1] &lt;End&gt;xs:dateTime&lt;/End&gt;    [1..1] &lt;Message&gt;xs:string&lt;/Message&gt;    [1..1] &lt;/...&gt; </pre> |

### ComplexType: PreviousType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; [ START CHOICE ]   &lt;RequestStoryID&gt; „SimpleType: IDType“ &lt;/RequestStoryID&gt;    [1..1]   &lt;ResponseStoryID&gt; „SimpleType: IDType“ &lt;/ResponseStoryID&gt;    [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |

### ComplexType: ProducerMessagesType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Message&gt; „ComplexType: MessageType“ &lt;/Message&gt;    [3..unbounded] &lt;/...&gt; </pre> |

### ComplexType: ProducerSecurityTokensType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Timestamp&gt;xs:dateTime&lt;/Timestamp&gt;    [1..unbounded] </pre> |

|  |  |
|--|--|
|  | <pre>&lt;X509Certificate&gt; <a href="#">„ComplexType: X509CertificateType“</a> &lt;/X509Certificate&gt; [1..unbounded] &lt;/...&gt;</pre> |
|--|--|

## ComplexType: ReceivedStoriesType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;StoryID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/StoryID&gt; [1..unbounded] &lt;/...&gt;</pre> |

## ComplexType: RegisteredMaintenanceType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Start&gt;xs:dateTime&lt;/Start&gt; [1..1] &lt;End&gt;xs:dateTime&lt;/End&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: RequestContextType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | Kontext der Anfrage  |
| Short description           | Der RequestContext enthält Informationen zum Zeitpunkt der Übermittlung, dem Absender und der Sprache.   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;UserAgent&gt; <a href="#">„ComplexType: UserAgentType“</a> &lt;/UserAgent&gt; [1..1] &lt;CompanyName&gt; <a href="#">„SimpleType: NotEmptyStringType“</a> &lt;/CompanyName&gt; [1..1] &lt;TransmissionDate&gt;xs:dateTime&lt;/TransmissionDate&gt; [1..1] &lt;RequestID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/RequestID&gt; [1..1] &lt;LanguageCode&gt; <a href="#">„SimpleType: LanguageCodeType“</a> &lt;/LanguageCode&gt; [1..1] &lt;MonitoringID&gt; <a href="#">„SimpleType: MonitoringIDType“</a> &lt;/MonitoringID&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: RequestType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;RequestContext&gt; <a href="#">„ComplexType: RequestContextType“</a> &lt;/RequestContext&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: ResponseContextType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | Kontext der Antwort  |
| Short description           | Der ResponseContext enthält Informationen zum Zeitpunkt der Übermittlung, dem Absender und der Sprache.  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;UserAgent&gt; <a href="#">„ComplexType: UserAgentType“</a> &lt;/UserAgent&gt; [1..1] &lt;InstitutionName&gt;xs:string&lt;/InstitutionName&gt; [1..1] &lt;TransmissionDate&gt;xs:dateTime&lt;/TransmissionDate&gt; [1..1] &lt;ResponseID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/ResponseID&gt; [1..1] &lt;RequestID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/RequestID&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: ResponseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;ResponseContext&gt; <a href="#">„ComplexType: ResponseContextType“</a> &lt;/ResponseContext&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: ResultType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Credentials&gt;„ComplexType: CredentialsType“&lt;/Credentials&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: SectionType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Heading&gt;„SimpleType: IDType“&lt;/Heading&gt; [0..1]   &lt;Description&gt;xs:token&lt;/Description&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: SignatureCertificateUID-BFSType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;CommonName&gt;xs:string&lt;/CommonName&gt; [1..1]   &lt;UID&gt;xs:string&lt;/UID&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: StoryBaseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1]   &lt;StoryID&gt;„SimpleType: IDType“&lt;/StoryID&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: StoryNotificationType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: NotificationType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt;   [ BASE TYPE „ComplexType: NotificationType“ ]   &lt;QualityLevel&gt;„SimpleType: QualityLevelType“&lt;/QualityLevel&gt; [1..1]   &lt;DescriptionCode&gt;„SimpleType: DescriptionCodeType“&lt;/DescriptionCode&gt; [1..1]   &lt;Description&gt;xs:string&lt;/Description&gt; [1..1]   [ END BASE TYPE ]   &lt;StoryID&gt;„SimpleType: IDType“&lt;/StoryID&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: StoryNotificationsType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Notification&gt;„ComplexType: StoryNotificationType“&lt;/Notification&gt; [1..unbounded] &lt;/...&gt;</pre> |

## ComplexType: SubstitutionMappingType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;MappedPredecessorDistributorRequestID&gt;„SimpleType: IDType“&lt;/MappedPredecessorDis\ tributorRequestID&gt; [1..1]   &lt;MappedPredecessorConsumerResponseID&gt;„SimpleType: IDType“&lt;/MappedPredecessorConsu\ merResponseID&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: SynchronizeRequestType

|          |    |
|----------|----|
| Abstract | no |
|----------|----|



|                             |  |
|-----------------------------|--|
| Parent type                 | <a href="#">„ComplexType: RequestType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE <a href="#">„ComplexType: RequestType“</a> ]   &lt;RequestContext&gt;<a href="#">„ComplexType: RequestContextType“</a>&lt;/RequestContext&gt; [1..1] [ END BASE TYPE ]   &lt;Sender&gt;<a href="#">„ComplexType: CompanyUIDType“</a>&lt;/Sender&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: TimePeriodType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;from&gt;xs:date&lt;/from&gt; [1..1]   &lt;until&gt;xs:date&lt;/until&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: UID-BFS-UnknownType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | Unternehmens UID-BFS   |
| Short description           | Unternehmens-Identifikationsnummer / UID-BFS   |
| XML Instance Representation | <pre>&lt;...&gt; [ START CHOICE ]   &lt;UID&gt;<a href="#">„SimpleType: UID-BFSType“</a>&lt;/UID&gt; [1..1]   &lt;Unknown&gt;<a href="#">„ComplexType: EmptyType“</a>&lt;/Unknown&gt; [1..1] [ END CHOICE ] &lt;/...&gt;</pre> |

## ComplexType: UserAgentType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | Applikationsinformation  |
| Short description           | Beschreibung der wesentlichen System-Identifikation-Daten  |
| Technical description       | Zur Qualitätssicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert  |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Producer&gt;<a href="#">„SimpleType: NotEmptyStringType“</a>&lt;/Producer&gt; [1..1]   &lt;Name&gt;<a href="#">„SimpleType: NotEmptyStringType“</a>&lt;/Name&gt; [1..1]   &lt;Version&gt;<a href="#">„SimpleType: NotEmptyStringType“</a>&lt;/Version&gt; [1..1]   &lt;StandardVersion&gt;xs:decimal&lt;/StandardVersion&gt; [1..1]   &lt;Certificate&gt;<a href="#">„SimpleType: NotEmptyStringType“</a>&lt;/Certificate&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: UserNotificationType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;Name&gt;xs:string&lt;/Name&gt; [1..1]   &lt;EmailAddress&gt;<a href="#">„SimpleType: EmailAddressType“</a>&lt;/EmailAddress&gt; [1..1]   &lt;PhoneNumber&gt;xs:string&lt;/PhoneNumber&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: VersionMappingFromType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt;   &lt;WSDLNamespace&gt;xs:anyURI&lt;/WSDLNamespace&gt; [1..1]   &lt;ServiceTypesNamespace&gt;xs:anyURI&lt;/ServiceTypesNamespace&gt; [1..1]   &lt;MinorVersion&gt;xs:decimal&lt;/MinorVersion&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: X509CertificateType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Usage&gt;_SimpleType: UsageType_&lt;/Usage&gt; [1..1] &lt;DistinguishedNameOwner&gt;xs:string&lt;/DistinguishedNameOwner&gt; [1..1] &lt;DistinguishedNameIssuer&gt;xs:string&lt;/DistinguishedNameIssuer&gt; [1..1] &lt;SecuredElement&gt;xs:string&lt;/SecuredElement&gt; [1..unbounded] &lt;PEM&gt;xs:base64Binary&lt;/PEM&gt; [1..1] &lt;/...&gt; </pre> |

## SimpleType: DescriptionCodeType

|                       |   |
|-----------------------|---|
| Translation           | Beschreibungscode   |
| Technical description | Severity: Information Code 9999 ist immer für Freitext reserviert. Die restlichen Codes sind dynamisch in AcknowledgementNotification spezifiziert. |
| Content type          | Restriction   |
| Base type             | xs:nonNegativeInteger   |
| Facets                | <ul style="list-style-type: none"> <li>TotalDigits: 7</li> </ul>  |

## SimpleType: EmailAddressType

|                       |   |
|-----------------------|---|
| Translation           | Email Adresse   |
| Short description     | Email Adresse   |
| Technical description | Email Adresse   |
| Content type          | Restriction   |
| Base type             | xs:string   |
| Facets                | <ul style="list-style-type: none"> <li>Pattern: [^@]+@[^\.]+\..+</li> </ul> |

## SimpleType: FaultCodeType

|              |  |
|--------------|--|
| Translation  | Fehler-Code  |
| Content type | Restriction  |
| Base type    | xs:NMTOKEN   |
| Facets       | <ul style="list-style-type: none"> <li>Enum: NOT_accepted</li> <li>Enum: NOT_plausible</li> <li>Enum: NOT_valid</li> </ul> |

## SimpleType: IDType

|              |  |
|--------------|--|
| Content type | Restriction  |
| Base type    | xs:string  |
| Facets       | <ul style="list-style-type: none"> <li>MinLength: 1</li> <li>MaxLength: 255</li> </ul> |

## SimpleType: InstanceRefIDType

|              |                      |
|--------------|----------------------|
| Translation  | Instance Referenz ID |
| Content type | Restriction          |
| Base type    | xs:string            |

|        |  |
|--------|--|
| Facets | <ul style="list-style-type: none"> <li>• Pattern: #.*</li> </ul> |
|--------|--|

### SimpleType: JobKeyType

|                   |   |
|-------------------|---|
| Translation       | Eindeutiger Bezeichner für den Job                                |
| Short description | Eindeutiger Bezeichner für den Job                                |
| Content type      | Restriction   |
| Base type         | xs:string   |
| Facets            | <ul style="list-style-type: none"> <li>• MinLength: 10</li> </ul> |

### SimpleType: LanguageCodeType

|                   |  |
|-------------------|--|
| Translation       | Sprachcode   |
| Short description | Sprachcode   |
| Content type      | Restriction  |
| Base type         | xs:NMTOKEN   |
| Facets            | <ul style="list-style-type: none"> <li>• Enum: de</li> <li>• Enum: fr</li> <li>• Enum: it</li> <li>• Enum: en</li> </ul> |

### SimpleType: Max100PercentType

|              |   |
|--------------|---|
| Translation  | Prozent (maximal 100%)  |
| Content type | Restriction   |
| Base type    | xs:decimal  |
| Facets       | <ul style="list-style-type: none"> <li>• MaxInclusive: 100.00</li> <li>• Pattern: [0-9]+\.[0-9]{2}</li> </ul> |

### SimpleType: MonitoringIDType

|                       |  |
|-----------------------|--|
| Translation           | KontrollID   |
| Short description     | Die MonitoringID erlaubt es, auf den Testsystemen Übermittlungen den Benutzern zuzuweisen.<br>Sie ist in der Produktion überflüssig. |
| Technical description | Die KontrollID wird vor allem in der Referenzapplikation zur Aufteilung der Daten verwendet  |
| Content type          | Restriction  |
| Base type             | xs:string  |
| Facets                | <ul style="list-style-type: none"> <li>• MinLength: 1</li> <li>• MaxLength: 32</li> </ul>  |

### SimpleType: NotEmptyStringType

|              |  |
|--------------|--|
| Content type | Restriction  |
| Base type    | xs:string  |
| Facets       | <ul style="list-style-type: none"> <li>• MinLength: 1</li> </ul> |

### SimpleType: PercentType

|              |   |
|--------------|---|
| Translation  | Prozent   |
| Content type | Restriction   |
| Base type    | xs:decimal  |
| Facets       | <ul style="list-style-type: none"> <li>Pattern: [0-9]+\.[0-9]{2}</li> </ul> |

### SimpleType: QualityLevelType

|                       |   |
|-----------------------|---|
| Translation           | Qualität-Stufen   |
| Short description     | Qualität-Stufen der Übermittlung  |
| Technical description | Folgende Stufen gelten<br>Validity: erste Stufe;<br>Plausibility: zweite Stufe;<br>Acceptance: dritte Stufe;<br>Comment: vierte Stufe;        |
| Content type          | Restriction   |
| Base type             | xs:NMTOKEN  |
| Facets                | <ul style="list-style-type: none"> <li>Enum: Validity</li> <li>Enum: Plausibility</li> <li>Enum: Acceptance</li> <li>Enum: Comment</li> </ul> |

### SimpleType: SalaryAmountType

|                       |   |
|-----------------------|---|
| Translation           | LohnbetragsTyp  |
| Short description     | LohnbetragsTyp  |
| Technical description | LohnbetragsTyp  |
| Content type          | Restriction   |
| Base type             | xs:decimal  |
| Facets                | <ul style="list-style-type: none"> <li>Pattern: [-]?[0-9]+\.[0-9]{2}</li> </ul> |

### SimpleType: SimpleBooleanType

|                   |   |
|-------------------|---|
| Translation       | SimpleBoolean   |
| Short description | Boolean   |
| Content type      | Restriction   |
| Base type         | xs:boolean  |
| Facets            | <ul style="list-style-type: none"> <li>Pattern: true</li> <li>Pattern: false</li> </ul> |

### SimpleType: StandardFormIDType

|              |  |
|--------------|--|
| Translation  | Standard Formular ID   |
| Content type | Restriction  |
| Base type    | <a href="#">„SimpleType: IDType“</a>   |
| Facets       | <ul style="list-style-type: none"> <li>Pattern: notStandard ([0-9]{4}\.[0-9]{4}\.[0-9]{4}-[0-9]{3})</li> </ul> |

### SimpleType: UID-BFSType

|                   |   |
|-------------------|---|
| Translation       | Unternehmens UID-BFS Typ  |
| Short description | Unternehmens UID-BFS Typ  |
| Content type      | Restriction   |
| Base type         | xs:NMTOKEN  |
| Facets            | <ul style="list-style-type: none"> <li>Pattern: CHE-[0-9]{3}\.[0-9]{3}\.[0-9]{3}</li> </ul> |

### SimpleType: UsageType

|              |   |
|--------------|---|
| Content type | Restriction   |
| Base type    | xs:NMTOKEN  |
| Facets       | <ul style="list-style-type: none"> <li>Enum: Signature</li> <li>Enum: QualifiedSignature</li> <li>Enum: Encryption</li> </ul> |

### SimpleType: YesNoUnknownType

|              |  |
|--------------|--|
| Translation  | ja, nein oder unbekannt  |
| Content type | Restriction  |
| Base type    | xs:NMTOKEN   |
| Facets       | <ul style="list-style-type: none"> <li>Enum: yes</li> <li>Enum: no</li> <li>Enum: unknown</li> </ul> |

## G.6 Schema Dokumentation Common.xsd

|                     |  |
|---------------------|--|
| Target Namespace    | urn:ch:swissdec:common:v3:20260306   |
| Declared Namespaces | <ul style="list-style-type: none"> <li>c : urn:ch:swissdec:common:v3:20260306</li> <li>ep : urn:ch:swissdec:basis:v1:20260306:components</li> <li>xs : http://www.w3.org/2001/XMLSchema</li> </ul> |
| Version             | 0.0  |

### ComplexType: AddressExtensionType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: AddressWorkplaceType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: AddressWorkplaceType“ ] &lt;ComplementaryLine&gt;xs:string&lt;/ComplementaryLine&gt; [0..1] &lt;Street&gt;xs:string&lt;/Street&gt; [0..1] &lt;Locality&gt;xs:string&lt;/Locality&gt; [0..1] &lt;ZIP-Code&gt;„SimpleType: ZIP-CodeType“&lt;/ZIP-Code&gt; [1..1] &lt;City&gt;xs:string&lt;/City&gt; [1..1] &lt;Country&gt;xs:string&lt;/Country&gt; [0..1] [ END BASE TYPE ] &lt;Canton&gt;„SimpleType: CantonAddressType“&lt;/Canton&gt; [0..1] &lt;MunicipalityID&gt;„SimpleType: MunicipalityIDType“&lt;/MunicipalityID&gt; [0..1] &lt;/...&gt;</pre> |

### ComplexType: AddressType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;ComplementaryLine&gt;xs:string&lt;/ComplementaryLine&gt; [0..1] &lt;Street&gt;xs:string&lt;/Street&gt; [0..1] &lt;Postbox&gt;xs:string&lt;/Postbox&gt; [0..1]</pre> |

|  |   |
|--|---|
|  | <pre> &lt;Locality&gt;xs:string&lt;/Locality&gt;    [0..1] &lt;ZIP-Code&gt;„SimpleType: ZIP-CodeType“ &lt;/ZIP-Code&gt;    [1..1] &lt;City&gt;xs:string&lt;/City&gt;    [1..1] &lt;Country&gt;xs:string&lt;/Country&gt;    [0..1] &lt;/...&gt; </pre> |
|--|---|

## ComplexType: AddressWorkplaceType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;ComplementaryLine&gt;xs:string&lt;/ComplementaryLine&gt;    [0..1] &lt;Street&gt;xs:string&lt;/Street&gt;    [0..1] &lt;Locality&gt;xs:string&lt;/Locality&gt;    [0..1] &lt;ZIP-Code&gt;„SimpleType: ZIP-CodeType“ &lt;/ZIP-Code&gt;    [1..1] &lt;City&gt;xs:string&lt;/City&gt;    [1..1] &lt;Country&gt;xs:string&lt;/Country&gt;    [0..1] &lt;/...&gt; </pre> |

## ComplexType: BalanceType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; [ START CHOICE ] &lt;FavourCompany&gt;„SimpleType: SalaryAmountAbsoluteType“ &lt;/FavourCompany&gt;    [1..1] &lt;FavourInstitution&gt;„SimpleType: SalaryAmountAbsoluteType“ &lt;/FavourInstitution&gt;    [1..1] &lt;FavorPerson&gt;„SimpleType: SalaryAmountAbsoluteType“ &lt;/FavorPerson&gt;    [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |

## ComplexType: CertificateSignRequestBaseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt;    [1..1] &lt;StoryID&gt;„SimpleType: IDType“ &lt;/StoryID&gt;    [1..1] [ END BASE TYPE ] &lt;PEM&gt;xs:base64Binary&lt;/PEM&gt;    [1..1] &lt;/...&gt; </pre> |

## ComplexType: CertificateSignRequestType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: CertificateSignRequestBaseType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: CertificateSignRequestBaseType“ ] [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt;    [1..1] &lt;StoryID&gt;„SimpleType: IDType“ &lt;/StoryID&gt;    [1..1] [ END BASE TYPE ] &lt;PEM&gt;xs:base64Binary&lt;/PEM&gt;    [1..1] [ END BASE TYPE ] &lt;OneTimePassword&gt;„SimpleType: NotEmptyStringType“ &lt;/OneTimePassword&gt;    [1..1] &lt;/...&gt; </pre> |

## ComplexType: CertificateSignResponseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt;    [1..1] &lt;StoryID&gt;„SimpleType: IDType“ &lt;/StoryID&gt;    [1..1] [ END BASE TYPE ] &lt;SubjectDN&gt;„SimpleType: NotEmptyStringType“ &lt;/SubjectDN&gt;    [1..1] &lt;IssuerDN&gt;„SimpleType: NotEmptyStringType“ &lt;/IssuerDN&gt;    [1..1] &lt;NotBefore&gt;xs:dateTime&lt;/NotBefore&gt;    [1..1] &lt;NotAfter&gt;xs:dateTime&lt;/NotAfter&gt;    [1..1] &lt;PEM&gt;xs:base64Binary&lt;/PEM&gt;    [1..1] &lt;/...&gt; </pre> |

## ComplexType: CivilStatusAndDateType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Status&gt;„SimpleType: CivilStatusType“&lt;/Status&gt;    [1..1] &lt;ValidAsOf&gt;xs:date&lt;/ValidAsOf&gt;    [0..1] &lt;/...&gt;</pre> |

## ComplexType: CompanyDescriptionBaseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Name&gt;„ComplexType: CompanyNameType“&lt;/Name&gt;    [1..1] &lt;Owner&gt;„ComplexType: CompanyOwnerType“&lt;/Owner&gt;    [0..1] &lt;Address&gt;„ComplexType: AddressType“&lt;/Address&gt;    [1..1] &lt;UID-BFS&gt;„ComplexType: UID-BFS-UnknownType“&lt;/UID-BFS&gt;    [1..1] &lt;Delegate&gt;„ComplexType: DelegateType“&lt;/Delegate&gt;    [0..1] &lt;/...&gt;</pre> |

## ComplexType: CompanyDescriptionType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: CompanyDescriptionBaseType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: CompanyDescriptionBaseType“ ] &lt;Name&gt;„ComplexType: CompanyNameType“&lt;/Name&gt;    [1..1] &lt;Owner&gt;„ComplexType: CompanyOwnerType“&lt;/Owner&gt;    [0..1] &lt;Address&gt;„ComplexType: AddressType“&lt;/Address&gt;    [1..1] &lt;UID-BFS&gt;„ComplexType: UID-BFS-UnknownType“&lt;/UID-BFS&gt;    [1..1] &lt;Delegate&gt;„ComplexType: DelegateType“&lt;/Delegate&gt;    [0..1] [ END BASE TYPE ] &lt;Workplace&gt;„ComplexType: WorkplaceType“&lt;/Workplace&gt;    [1..unbounded] &lt;CompanyWorkingTime&gt;„ComplexType: CompanyWorkingTimeIDType“&lt;/ CompanyWorkingTime&gt;    [1..unbounded] &lt;/...&gt;</pre> |

## ComplexType: CompanyNameType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;HR-RC-Name&gt;xs:string&lt;/HR-RC-Name&gt;    [1..1] &lt;ComplementaryLine&gt;xs:string&lt;/ComplementaryLine&gt;    [0..2] &lt;/...&gt;</pre> |

## ComplexType: CompanyOwnerType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Firstname&gt;xs:string&lt;/Firstname&gt;    [1..1] &lt;Lastname&gt;xs:string&lt;/Lastname&gt;    [1..1] &lt;/...&gt;</pre> |

## ComplexType: CompanyWorkingTimeIDType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Description&gt;xs:string&lt;/Description&gt;    [0..1] [ START CHOICE ] &lt;WeeklyHours&gt;„SimpleType: HoursOrLessonsType“&lt;/WeeklyHours&gt;    [1..1] &lt;WeeklyLessons&gt;„SimpleType: HoursOrLessonsType“&lt;/WeeklyLessons&gt;    [1..1] &lt;WeeklyHoursAndLessons&gt;„ComplexType: WeeklyHoursAndLessonsType“&lt;/WeeklyHoursAnd\ Lessons&gt;    [1..1] [ END CHOICE ] &lt;/...&gt;</pre> |

## ComplexType: ContactRequestType

|          |    |
|----------|----|
| Abstract | no |
|----------|----|

|                             |   |
|-----------------------------|---|
| XML Instance Representation | <pre>&lt;...&gt; &lt;Contact&gt;__ComplexType: ContactType__&lt;/Contact&gt; [1..1] &lt;Info&gt;__ComplexType: NotificationsType__&lt;/Info&gt; [1..1] &lt;/...&gt;</pre> |
|-----------------------------|---|

## ComplexType: ContactStoryType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">__ComplexType: StoryBaseType__</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE __ComplexType: StoryBaseType__ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt;__SimpleType: IDType__&lt;/StoryID&gt; [1..1] [ END BASE TYPE ] &lt;Contact&gt;__ComplexType: ContactType__&lt;/Contact&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: ContactType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Name&gt;__SimpleType: NotEmptyStringType__&lt;/Name&gt; [1..1] &lt;EmailAddress&gt;__SimpleType: EmailAddressType__&lt;/EmailAddress&gt; [0..1] &lt;PhoneNumber&gt;xs:string&lt;/PhoneNumber&gt; [0..1] &lt;MobilePhoneNumber&gt;xs:string&lt;/MobilePhoneNumber&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: DelegateType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Name&gt;__ComplexType: CompanyNameType__&lt;/Name&gt; [1..1] &lt;Owner&gt;__ComplexType: CompanyOwnerType__&lt;/Owner&gt; [0..1] &lt;Address&gt;__ComplexType: AddressType__&lt;/Address&gt; [1..1] &lt;UID-BFS&gt;__SimpleType: UID-BFSType__&lt;/UID-BFS&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: IdentificationBaseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;InsuranceCompanyName&gt;xs:string&lt;/InsuranceCompanyName&gt; [1..1] &lt;CustomerIdentity&gt;__SimpleType: NotEmptyStringType__&lt;/CustomerIdentity&gt; [1..1] &lt;ContractIdentity&gt;__SimpleType: NotEmptyStringType__&lt;/ContractIdentity&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: InstitutionDescriptionBaseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Name&gt;__ComplexType: CompanyNameType__&lt;/Name&gt; [1..1] &lt;Address&gt;__ComplexType: AddressType__&lt;/Address&gt; [1..1] &lt;UID-BFS&gt;__ComplexType: UID-BFS-UnknownType__&lt;/UID-BFS&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: NameAndAddressType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;Name&gt;xs:string&lt;/Name&gt; [1..1] &lt;Address&gt;__ComplexType: AddressType__&lt;/Address&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: ParticularsAddressType

|          |    |
|----------|----|
| Abstract | no |
|----------|----|



|                             |  |
|-----------------------------|--|
| Parent type                 | <a href="#">„ComplexType: AddressType“</a>   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE <a href="#">„ComplexType: AddressType“</a> ] &lt;ComplementaryLine&gt;xs:string&lt;/ComplementaryLine&gt; [0..1] &lt;Street&gt;xs:string&lt;/Street&gt; [0..1] &lt;Postbox&gt;xs:string&lt;/Postbox&gt; [0..1] &lt;Locality&gt;xs:string&lt;/Locality&gt; [0..1] &lt;ZIP-Code&gt;<a href="#">„SimpleType: ZIP-CodeType“</a>&lt;/ZIP-Code&gt; [1..1] &lt;City&gt;xs:string&lt;/City&gt; [1..1] &lt;Country&gt;xs:string&lt;/Country&gt; [0..1] [ END BASE TYPE ] &lt;ResidenceCanton&gt;<a href="#">„SimpleType: CantonAndEXTYPE“</a>&lt;/ResidenceCanton&gt; [1..1] &lt;MunicipalityID&gt;<a href="#">„SimpleType: MunicipalityIDType“</a>&lt;/MunicipalityID&gt; [0..1] &lt;DepartureDate&gt;xs:date&lt;/DepartureDate&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: ParticularsAddressesType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Address&gt;<a href="#">„ComplexType: ParticularsAddressType“</a>&lt;/Address&gt; [0..unbounded] &lt;/...&gt; </pre> |

## ComplexType: ParticularsBaseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Social-InsuranceIdentification&gt;<a href="#">„ComplexType: Social-InsuranceIdentificationType“</a>&lt;/Social-InsuranceIdentification&gt; [1..1] &lt;EmployeeNumber&gt;xs:string&lt;/EmployeeNumber&gt; [1..1] &lt;Lastname&gt;xs:string&lt;/Lastname&gt; [1..1] &lt;Firstname&gt;xs:string&lt;/Firstname&gt; [1..1] &lt;Sex&gt;<a href="#">„SimpleType: SexType“</a>&lt;/Sex&gt; [1..1] &lt;DateOfBirth&gt;xs:date&lt;/DateOfBirth&gt; [1..1] &lt;Nationality&gt;<a href="#">„SimpleType: NationalityType“</a>&lt;/Nationality&gt; [1..1] &lt;CivilStatus&gt;<a href="#">„ComplexType: CivilStatusAndDateType“</a>&lt;/CivilStatus&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: ParticularsType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: ParticularsBaseType“</a>   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE <a href="#">„ComplexType: ParticularsBaseType“</a> ] &lt;Social-InsuranceIdentification&gt;<a href="#">„ComplexType: Social-InsuranceIdentificationType“</a>&lt;/Social-InsuranceIdentification&gt; [1..1] &lt;EmployeeNumber&gt;xs:string&lt;/EmployeeNumber&gt; [1..1] &lt;Lastname&gt;xs:string&lt;/Lastname&gt; [1..1] &lt;Firstname&gt;xs:string&lt;/Firstname&gt; [1..1] &lt;Sex&gt;<a href="#">„SimpleType: SexType“</a>&lt;/Sex&gt; [1..1] &lt;DateOfBirth&gt;xs:date&lt;/DateOfBirth&gt; [1..1] &lt;Nationality&gt;<a href="#">„SimpleType: NationalityType“</a>&lt;/Nationality&gt; [1..1] &lt;CivilStatus&gt;<a href="#">„ComplexType: CivilStatusAndDateType“</a>&lt;/CivilStatus&gt; [1..1] [ END BASE TYPE ] &lt;Addresses&gt;<a href="#">„ComplexType: ParticularsAddressesType“</a>&lt;/Addresses&gt; [1..1] &lt;EmailAddress&gt;<a href="#">„SimpleType: EmailAddressType“</a>&lt;/EmailAddress&gt; [0..1] &lt;PhoneNumber&gt;xs:string&lt;/PhoneNumber&gt; [0..1] &lt;MobilePhoneNumber&gt;xs:string&lt;/MobilePhoneNumber&gt; [0..1] &lt;ResidenceCategory&gt;<a href="#">„SimpleType: ResidenceCategoryType“</a>&lt;/ResidenceCategory&gt; [0..1] &lt;LanguageCode&gt;<a href="#">„SimpleType: LanguageCodeType“</a>&lt;/LanguageCode&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: PersonBaseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Particulars&gt;<a href="#">„ComplexType: ParticularsType“</a>&lt;/Particulars&gt; [1..1] &lt;Work&gt;<a href="#">„ComplexType: WorkType“</a>&lt;/Work&gt; [1..unbounded] &lt;/...&gt; </pre> |

## ComplexType: RegisterOrganizationAuthenticationAddresseeContextType

|          |    |
|----------|----|
| Abstract | no |
|----------|----|

|                             |  |
|-----------------------------|--|
| Parent type                 | <a href="#">„ComplexType: AddresseeResponseContextType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: AddresseeResponseContextType“ ] [ BASE TYPE „ComplexType: ResponseContextType“ ] &lt;UserAgent&gt;„ComplexType: UserAgentType“ &lt;/UserAgent&gt; [1..1] &lt;InstitutionName&gt;xs:string&lt;/InstitutionName&gt; [1..1] &lt;TransmissionDate&gt;xs:dateTime&lt;/TransmissionDate&gt; [1..1] &lt;ResponseID&gt;„SimpleType: IDType“ &lt;/ResponseID&gt; [1..1] &lt;RequestID&gt;„SimpleType: IDType“ &lt;/RequestID&gt; [1..1] [ END BASE TYPE ] &lt;ProducerResponseNotifications&gt;„ComplexType: FeedbackNotificationsType“ &lt;/Produ\ cerResponseNotifications&gt; [1..1] &lt;Warning&gt;„ComplexType: NotificationsType“ &lt;/Warning&gt; [0..1] &lt;Info&gt;„ComplexType: NotificationsType“ &lt;/Info&gt; [0..1] [ END BASE TYPE ] &lt;CertificateRequestID&gt;„SimpleType: IDType“ &lt;/CertificateRequestID&gt; [1..1] &lt;TestCase&gt;„ComplexType: EmptyType“ &lt;/TestCase&gt; [0..1] &lt;/...&gt; </pre> |

### ComplexType: RegisterOrganizationAuthenticationAddresseeJobStateType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: AddresseeJobType“</a>   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: AddresseeJobType“ ] [ BASE TYPE „ComplexType: AddresseeType“ ] &lt;AddresseeIdentification&gt;„SimpleType: IDType“ &lt;/AddresseeIdentificati\ on&gt; [1..1] [ END BASE TYPE ] &lt;ProcessByDistributor&gt;„SimpleType: SimpleBooleanType“ &lt;/ProcessByDistribu\ tor&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;Success&gt;„ComplexType: RegisterOrganizationAuthenticationAddresseeSuccessJobStateType“ &lt;/ Success&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre> |

### ComplexType: RegisterOrganizationAuthenticationAddresseeRequestCaseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;CaseContext&gt;„ComplexType: RegisterOrganizationAuthenticationCaseContextType“ &lt;/ CaseContext&gt; [1..1] &lt;ReceivedState&gt;„SimpleType: RegisterOrganizationAuthenticationStateType“ &lt;/Receive\ dState&gt; [0..1] &lt;/...&gt; </pre> |

### ComplexType: RegisterOrganizationAuthenticationAddresseeResponseCaseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;CaseContext&gt;„ComplexType: RegisterOrganizationAuthenticationCaseContextType“ &lt;/ CaseContext&gt; [1..1] &lt;State&gt;„SimpleType: RegisterOrganizationAuthenticationStateType“ &lt;/State&gt; [1..1] &lt;Success&gt;„ComplexType: RegisterOrganizationAuthenticationConsumerSuccessType“ &lt;/ Success&gt; [0..1] &lt;/...&gt; </pre> |

### ComplexType: RegisterOrganizationAuthenticationAddresseeSuccessJobStateType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;AddresseeContext&gt;„ComplexType: RegisterOrganizationAuthenticationAddresseeContextType“ &lt;/ AddresseeContext&gt; [1..1] &lt;Credentials&gt;„ComplexType: CredentialsType“ &lt;/Credentials&gt; [1..1] &lt;/...&gt; </pre> |

### ComplexType: RegisterOrganizationAuthenticationCaseContextType

|          |    |
|----------|----|
| Abstract | no |
|----------|----|

|                             |   |
|-----------------------------|---|
| Parent type                 | <a href="#">„ComplexType: CaseContextType“</a>  |
| Translation                 | Fallkontext   |
| Short description           | Fallkontext   |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: CaseContextType“ ] [ BASE TYPE „ComplexType: CaseContextBaseType“ ] &lt;ReceivedStoryIDs&gt; „ComplexType: ReceivedStoriesType“ &lt;/ReceivedSto\ ryIDs&gt; [0..1] &lt;SuppressedSenderStoryIDs&gt; „ComplexType: ReceivedStoriesType“ &lt;/SuppressedSen\ derStoryIDs&gt; [0..1] &lt;SuppressedInstitutionStoryIDs&gt; „ComplexType: ReceivedStoriesType“ &lt;/Suppresse\ dInstitutionStoryIDs&gt; [0..1] &lt;Warning&gt; „ComplexType: StoryNotificationsType“ &lt;/Warning&gt; [0..1] &lt;Info&gt; „ComplexType: StoryNotificationsType“ &lt;/Info&gt; [0..1] [ END BASE TYPE ] &lt;Credentials&gt; „ComplexType: CredentialsType“ &lt;/Credentials&gt; [1..1] [ END BASE TYPE ] &lt;CertificateRequestID&gt; „SimpleType: IDType“ &lt;/CertificateRequestID&gt; [1..1] &lt;TestCase&gt; „ComplexType: EmptyType“ &lt;/TestCase&gt; [0..1] &lt;/...&gt; </pre> |

## ComplexType: RegisterOrganizationAuthenticationConsumerResponseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; &lt;RegisterOrganizationAuthenticationResponse&gt; „ComplexType: RegisterOrganizationAuthenticationAddressseeJobStateType“ &lt;/RegisterOrganizationAuthenticationResponse&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: RegisterOrganizationAuthenticationConsumerSuccessType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt; „SimpleType: IDType“ &lt;/StoryID&gt; [1..1] [ END BASE TYPE ] &lt;CompanyName&gt; „ComplexType: CompanyNameType“ &lt;/CompanyName&gt; [1..1] &lt;CompanyAddress&gt; „ComplexType: AddressType“ &lt;/CompanyAddress&gt; [1..1] &lt;CompanyUID-BFS&gt; „SimpleType: UID-BFSType“ &lt;/CompanyUID-BFS&gt; [1..1] &lt;QualityLevel&gt;xs:int&lt;/QualityLevel&gt; [1..1] &lt;InstitutionContactPerson&gt; „ComplexType: ContactType“ &lt;/InstitutionContactPer\ son&gt; [1..1] &lt;Delegate&gt; „ComplexType: DelegateType“ &lt;/Delegate&gt; [0..unbounded] &lt;/...&gt; </pre> |

## ComplexType: RegisterOrganizationAuthenticationJobStateType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Translation                 | Job Status   |
| Technical description       | Antwort Struktur der Adressaten.   |
| XML Instance Representation | <pre> &lt;...&gt; &lt;Addressee&gt; „ComplexType: RegisterOrganizationAuthenticationAddresseeJobStateType“ &lt;/Addressee&gt; [1..1] &lt;/...&gt; </pre> |

## ComplexType: RegisterOrganizationAuthenticationQuittanceType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: StoryBaseType“</a>  |
| XML Instance Representation | <pre> &lt;...&gt; [ BASE TYPE „ComplexType: StoryBaseType“ ] &lt;Creation&gt;xs:dateTime&lt;/Creation&gt; [1..1] &lt;StoryID&gt; „SimpleType: IDType“ &lt;/StoryID&gt; [1..1] [ END BASE TYPE ] &lt;X509Subject&gt;c:blubber&lt;/X509Subject&gt; [1..1] &lt;CompanyUID-BFS&gt; „SimpleType: UID-BFSType“ &lt;/CompanyUID-BFS&gt; [1..1] &lt;AuthorizedForUID-BFS&gt; „SimpleType: UID-BFSType“ &lt;/AuthorizedForUID-BFS&gt; [0..1] &lt;Comment&gt; „ComplexType: NotificationsType“ &lt;/Comment&gt; [0..1] </pre> |

|  |        |
|--|--------|
|  | </...> |
|--|--------|

## ComplexType: RegisterOrganizationAuthenticationResponseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: ResponseType“</a>  |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: ResponseType“ ] &lt;ResponseContext&gt; „ComplexType: ResponseContextType“ &lt;/ResponseContext&gt; [1..1] [ END BASE TYPE ] &lt;Addressees&gt; „ComplexType: RegisterOrganizationAuthenticationJobStateType“ &lt;/Addres\ sees&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: RegisterOrganizationAuthenticationSenderRequestCaseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Technical description       | Fallcontext und Stories   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;CaseContext&gt; „ComplexType: RegisterOrganizationAuthenticationCaseContextType“ &lt;/ CaseContext&gt; [1..1] &lt;ReceivedState&gt; „SimpleType: RegisterOrganizationAuthenticationStateType“ &lt;/Receive\ dState&gt; [0..1] &lt;SignCertificate&gt; „ComplexType: CertificateSignRequestType“ &lt;/SignCertifica\ te&gt; [0..1] &lt;RenewCertificate&gt; „ComplexType: CertificateSignRequestBaseType“ &lt;/RenewCertifica\ te&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: RegisterOrganizationAuthenticationSenderResponseCaseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Translation                 | Fall  |
| Short description           | Informationen zum Fall  |
| Technical description       | Informationen zum Fall  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;CaseContext&gt; „ComplexType: RegisterOrganizationAuthenticationCaseContextType“ &lt;/ CaseContext&gt; [1..1] &lt;State&gt; „SimpleType: RegisterOrganizationAuthenticationStateType“ &lt;/State&gt; [1..1] &lt;Quittance&gt; „ComplexType: RegisterOrganizationAuthenticationQuittanceType“ &lt;/Quit\ tance&gt; [0..1] &lt;Certificate&gt; „ComplexType: CertificateSignResponseType“ &lt;/Certificate&gt; [0..1] &lt;/...&gt;</pre> |

## ComplexType: Social-InsuranceIdentificationType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; [ START CHOICE ] &lt;SV-AS-Number&gt; „SimpleType: SV-AS-NumberType“ &lt;/SV-AS-Number&gt; [1..1] &lt;unknown&gt; „ComplexType: EmptyType“ &lt;/unknown&gt; [1..1] [ END CHOICE ] &lt;/...&gt;</pre> |

## ComplexType: SynchronizeRegisterOrganizationAuthenticationAddresseeResponse

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: AddresseeResponseType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: AddresseeResponseType“ ] &lt;AddresseeContext&gt; „ComplexType: AddresseeResponseContextType“ &lt;/AddresseeCon\ text&gt; [1..1] [ END BASE TYPE ] &lt;Addressee&gt; „ComplexType: AddresseeType“ &lt;/Addressee&gt; [1..1] &lt;Case&gt; „ComplexType: RegisterOrganizationAuthenticationAddresseeResponseCaseType“ &lt;/ Case&gt; [1..1]</pre> |

|  |        |
|--|--------|
|  | </...> |
|--|--------|

## ComplexType: SynchronizeRegisterOrganizationAuthenticationConsumerResponseType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;SynchronizeRegisterOrganizationAuthentication&gt; „ComplexType: SynchronizeRegisterOrganizationAuthentic SynchronizeRegisterOrganizationAuthentication“ [1..1] &lt;/...&gt;</pre> |

## ComplexType: SynchronizeRegisterOrganizationAuthenticationResponseType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| Parent type                 | <a href="#">„ComplexType: ResponseType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: ResponseType“ ] &lt;ResponseContext&gt; „ComplexType: ResponseContextType“ &lt;/ResponseContext&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;Error&gt; „ComplexType: ErrorResponseType“ &lt;/Error&gt; [1..1] &lt;SynchronizeRegisterOrganizationAuthenticationConsumer&gt; „ComplexType: SynchronizeRegisterOrganizati SynchronizeRegisterOrganizationAuthenticationConsumer“ [1..1] [ END CHOICE ] &lt;/...&gt;</pre> |

## ComplexType: SynchronizeRegisterOrganizationAuthenticationSenderType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| Parent type                 | <a href="#">„ComplexType: AddresseeResponseType“</a>   |
| XML Instance Representation | <pre>&lt;...&gt; [ BASE TYPE „ComplexType: AddresseeResponseType“ ] &lt;AddresseeContext&gt; „ComplexType: AddresseeResponseContextType“ &lt;/AddresseeCon\ text&gt; [1..1] [ END BASE TYPE ] &lt;Addressee&gt; „ComplexType: AddresseeType“ &lt;/Addressee&gt; [1..1] &lt;Case&gt; „ComplexType: RegisterOrganizationAuthenticationSenderResponseCaseType“ &lt;/ Case&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: TaxAtSourceCategoryType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; [ START CHOICE ] &lt;TaxAtSourceCode&gt; „SimpleType: TaxAtSourceCodeType“ &lt;/TaxAtSourceCode&gt; [1..1] &lt;CategoryPredefined&gt; „SimpleType: CategoryPredefinedType“ &lt;/CategoryPredefi\ ned&gt; [1..1] &lt;CategoryOpen&gt; „SimpleType: IDType“ &lt;/CategoryOpen&gt; [1..1] [ END CHOICE ] &lt;/...&gt;</pre> |

## ComplexType: WeeklyHoursAndLessonsType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre>&lt;...&gt; &lt;WeeklyHours&gt; „SimpleType: HoursOrLessonsType“ &lt;/WeeklyHours&gt; [1..1] &lt;WeeklyLessons&gt; „SimpleType: HoursOrLessonsType“ &lt;/WeeklyLessons&gt; [1..1] &lt;/...&gt;</pre> |

## ComplexType: WorkType

|                             |  |
|-----------------------------|--|
| Abstract                    | no   |
| XML Instance Representation | <pre>&lt;...&gt; &lt;WorkingTime&gt;c:blubber&lt;/WorkingTime&gt; [1..1] &lt;EntryDate&gt;xs:date&lt;/EntryDate&gt; [1..1] &lt;WithdrawalDate&gt;xs:date&lt;/WithdrawalDate&gt; [0..1]</pre> |

|  |        |
|--|--------|
|  | </...> |
|--|--------|

## ComplexType: WorkplaceType

|                             |   |
|-----------------------------|---|
| Abstract                    | no  |
| XML Instance Representation | <pre> &lt;...&gt; [ START CHOICE ] &lt;BUR-REE-Number&gt; „SimpleType: BUR-REE-NumberType“ &lt;/BUR-REE-Number&gt; [1..1] &lt;InHouseID&gt; „SimpleType: IDType“ &lt;/InHouseID&gt; [1..1] [ END CHOICE ] &lt;AddressExtended&gt; „ComplexType: AddressExtensionType“ &lt;/AddressExtended&gt; [1..1] &lt;/...&gt; </pre> |

## SimpleType: AssuranceCategoryCodeType

|                   |  |
|-------------------|--|
| Translation       | Versicherungskategorie-Code Typ  |
| Short description | Versicherungskategorie-Code Typ  |
| Content type      | Restriction  |
| Base type         | xs:NMTOKEN   |
| Facets            | <ul style="list-style-type: none"> <li>Pattern: [A-Z0-9]{2}</li> </ul> |

## SimpleType: BUR-REE-NumberType

|              |  |
|--------------|--|
| Translation  | BUR-Nummer Typ   |
| Content type | Restriction  |
| Base type    | xs:NMTOKEN   |
| Facets       | <ul style="list-style-type: none"> <li>Pattern: [A-Z][0-9]{8}</li> </ul> |

## SimpleType: CantonAddressType

|                   |  |
|-------------------|--|
| Translation       | KantonsTyp   |
| Short description | KantonsTyp der Schweiz   |
| Content type      | Restriction  |
| Base type         | „SimpleType: CantonAndEXTType“   |
| Facets            | <ul style="list-style-type: none"> <li>Enum: AG</li> <li>Enum: AI</li> <li>Enum: AR</li> <li>Enum: BE</li> <li>Enum: BL</li> <li>Enum: BS</li> <li>Enum: FR</li> <li>Enum: GE</li> <li>Enum: GL</li> <li>Enum: GR</li> <li>Enum: JU</li> <li>Enum: LU</li> <li>Enum: NE</li> <li>Enum: NW</li> <li>Enum: OW</li> <li>Enum: SG</li> <li>Enum: SH</li> <li>Enum: SO</li> <li>Enum: SZ</li> <li>Enum: TG</li> </ul> |

|  |  |
|--|--|
|  | <ul style="list-style-type: none"> <li>• Enum: TI</li> <li>• Enum: UR</li> <li>• Enum: VD</li> <li>• Enum: VS</li> <li>• Enum: ZG</li> <li>• Enum: ZH</li> </ul> |
|--|--|

### SimpleType: CantonAndEXType

|              |  |
|--------------|--|
| Translation  | KantonsTyp (inkl. EX)  |
| Content type | Restriction  |
| Base type    | xs:NMTOKEN   |
| Facets       | <ul style="list-style-type: none"> <li>• Enum: AG</li> <li>• Enum: AI</li> <li>• Enum: AR</li> <li>• Enum: BE</li> <li>• Enum: BL</li> <li>• Enum: BS</li> <li>• Enum: FR</li> <li>• Enum: GE</li> <li>• Enum: GL</li> <li>• Enum: GR</li> <li>• Enum: JU</li> <li>• Enum: LU</li> <li>• Enum: NE</li> <li>• Enum: NW</li> <li>• Enum: OW</li> <li>• Enum: SG</li> <li>• Enum: SH</li> <li>• Enum: SO</li> <li>• Enum: SZ</li> <li>• Enum: TG</li> <li>• Enum: TI</li> <li>• Enum: UR</li> <li>• Enum: VD</li> <li>• Enum: VS</li> <li>• Enum: ZG</li> <li>• Enum: ZH</li> <li>• Enum: EX</li> </ul> |

### SimpleType: CategoryPredefinedType

|                       |  |
|-----------------------|--|
| Translation           | Vordefinierte Kategorien   |
| Short description     | <p>Mögliche Werte der vordefinierte Kategorien:</p> <ul style="list-style-type: none"> <li>- HEN, HEY: VR-Honorar an qsP mit Wohnsitz im Ausland</li> <li>- MEN, MEY: Leistungen aus exportierten Mitarbeiterbeteiligungen an qsP mit Wohnsitz im Ausland</li> <li>- NON, NOY: Nicht quellensteuerpflichtig</li> <li>- SFN: Sondervereinbarung mit Frankreich</li> </ul> |
| Technical description | 4 mögliche Werte der vordefinierten Kategorien.  |

|              |   |
|--------------|---|
| Content type | Restriction   |
| Base type    | xs:NMTOKEN  |
| Facets       | <ul style="list-style-type: none"> <li>• Enum: HEN</li> <li>• Enum: HEY</li> <li>• Enum: MEN</li> <li>• Enum: MEY</li> <li>• Enum: NON</li> <li>• Enum: NOY</li> <li>• Enum: SFN</li> </ul> |

### SimpleType: CivilStatusType

|                   |  |
|-------------------|--|
| Translation       | ZivilstandsTyp   |
| Short description | ZivilstandsTyp   |
| Content type      | Restriction  |
| Base type         | xs:NMTOKEN   |
| Facets            | <ul style="list-style-type: none"> <li>• Enum: unknown</li> <li>• Enum: single</li> <li>• Enum: married</li> <li>• Enum: widowed</li> <li>• Enum: divorced</li> <li>• Enum: separated</li> <li>• Enum: registeredPartnership</li> <li>• Enum: partnershipDissolvedByLaw</li> <li>• Enum: partnershipDissolvedByDeath</li> <li>• Enum: partnershipDissolvedByDeclarationOfLost</li> </ul> |

### SimpleType: EmploymentContractType

|                   |   |
|-------------------|---|
| Translation       | Arbeitsvertragsart  |
| Short description | <p>Arbeitsvertragsart ist als folgende mögliche Werte anzugeben:</p> <ul style="list-style-type: none"> <li>- indefiniteSalaryMth = unbefristeter Vertrag mit Monatslohn (indefinite duration with salary per month)</li> <li>- indefiniteSalaryMthAWT = unbefristeter Vertrag mit Monatslohn und Jahresarbeitszeit (indefinite duration with salary per month and annual working time model)</li> <li>- indefiniteSalaryHrs = unbefristeter Vertrag mit Stundenlohn (indefinite duration with salary per month)</li> <li>- indefiniteSalaryNoTimeConstraint = unbefristeter Vertrag mit Provision, Pauschale, Akkordlohn (indefinite duration with commission, lump sum, piece rate)</li> <li>- fixedSalaryMth = befristeter Vertrag mit Monatslohn (fixed/temporary duration salary per month)</li> </ul> |



|                       |  |
|-----------------------|--|
|                       | <ul style="list-style-type: none"> <li>- fixedSalaryHrs = befristeter Vertrag mit Stundenlohn (fixed/temporary duration salary per hour)</li> <li>- fixedSalaryNoTimeConstraint = befristeter Vertrag mit Provision, Pauschale, Akkordlohn (fixed/temporary duration with commission, lump sum, piece rate)</li> <li>- apprentice = Lehrvertrag</li> <li>- internshipContract = Praktikumsvertrag</li> </ul> |
| Technical description | Arbeitsvertragsart ist aus 11 möglichen Werte anzugeben.   |
| Content type          | Restriction  |
| Base type             | xs:NMTOKEN   |
| Facets                | <ul style="list-style-type: none"> <li>• Enum: indefiniteSalaryMth</li> <li>• Enum: indefiniteSalaryMthAWT</li> <li>• Enum: indefiniteSalaryHrs</li> <li>• Enum: indefiniteSalaryNoTimeConstraint</li> <li>• Enum: fixedSalaryMth</li> <li>• Enum: fixedSalaryHrs</li> <li>• Enum: fixedSalaryNoTimeConstraint</li> <li>• Enum: apprentice</li> <li>• Enum: internshipContract</li> </ul>                    |

### SimpleType: HoursOrLessonsType

|              |   |
|--------------|---|
| Translation  | Stunden oder Lektionen  |
| Content type | Restriction   |
| Base type    | xs:decimal  |
| Facets       | <ul style="list-style-type: none"> <li>• Pattern: [0-9]+\.[0-9]{2}</li> </ul> |

### SimpleType: MunicipalityIDType

|              |   |
|--------------|---|
| Translation  | Gemeindenummer  |
| Content type | Restriction   |
| Base type    | xs:int  |
| Facets       | <ul style="list-style-type: none"> <li>• MinInclusive: 1</li> <li>• MaxInclusive: 9999</li> <li>• TotalDigits: 4</li> </ul> |

### SimpleType: NationalityType

|                       |  |
|-----------------------|--|
| Translation           | Staatsangehörigkeit  |
| Short description     | Die Codes für eine korrekte Zuteilung der Staatsangehörigkeit sind bei der UNO hinterlegt (ISO 3166)   |
| Technical description | <p>Erweiterung 11= unbekannt 22= staatenlos: Die ISO-Codes wurden auf dem Prod-Distri noch nie geprüft: In der Qualitätsstufe Plausibilität kann evtl. gegen die ISO 3166 Codes geprüft und eine Warnung angezeigt werden.</p> <p>Achtung: Probleme mit bestehenden Datenbeständen (Änderungen der Ländernamen und Verwechslungen)</p> |

|              |  |
|--------------|--|
| Content type | Restriction  |
| Base type    | xs:string  |
| Facets       | <ul style="list-style-type: none"> <li>Pattern: [A-Z][A-Z]{11 22}</li> </ul> |

### SimpleType: PositionType

|                   |  |
|-------------------|--|
| Translation       | Berufliche Stellung  |
| Short description | <p>Berufliche Stellung des Arbeitnehmers:</p> <p>Mögliche Werte:</p> <p>highestCadre: Oberes Kader;<br/>middleCadre: Mittleres Kader;<br/>lowerCadre: Unteres Kader;<br/>lowestCadre: Verantwortlich für die Ausführung der Arbeiten;<br/>noCadre: ohne Kaderfunktion;</p> |
| Content type      | Restriction  |
| Base type         | xs:NMTOKEN   |
| Facets            | <ul style="list-style-type: none"> <li>Enum: highestCadre</li> <li>Enum: middleCadre</li> <li>Enum: lowerCadre</li> <li>Enum: lowestCadre</li> <li>Enum: noCadre</li> </ul>  |

### SimpleType: RegisterOrganizationAuthenticationStateType

|                   |   |
|-------------------|---|
| Translation       | Status des Falls  |
| Short description | Status des Falls  |
| Content type      | Restriction   |
| Base type         | xs:string   |
| Facets            | <ul style="list-style-type: none"> <li>Enum: processing</li> <li>Enum: registered</li> <li>Enum: rejected</li> <li>Enum: verified</li> <li>Enum: expired</li> </ul> |

### SimpleType: ResidenceCategoryType

|                   |   |
|-------------------|---|
| Translation       | Aufenthaltskategorien   |
| Short description | <p>Mögliche Werte der Aufenthaltskategorien für Ausländer:</p> <p>shortTerm-L = Kurzaufenthalter (L)</p> <p>annual-B = Jahresaufenthalter (B)</p> <p>settled-C = Niedergelassene (C)</p> <p>crossBorder-G = Grenzgänger (G)</p> <p>asylumSeeker-N = Asylsuchender (N)</p> |

|                       |   |
|-----------------------|---|
|                       | <p>needForProtection-S = Schutzbedürftige (S)</p> <p>NotificationProcedureForShorttermWork90Days = Meldeverfahren für kurzfristige Erwerbstätigkeit (bis 90 Tage, Schengenregelung)</p> <p>NotificationProcedureForShorttermWork120Days = Meldeverfahren für kurzfristige Erwerbstätigkeit bis 120 Tage</p> <p>ProvisionallyAdmittedForeigners (F) = Vorläufig aufgenommene Ausländer</p> <p>ResidentForeignNationalWithGainfulEmployment (Ci) = Aufenthaltsbewilligung mit Erwerbstätigkeit</p> <p>othersNotSwiss = Andere (nicht Schweizer)</p> |
| Technical description | Es gibt 11 mögliche Werte der Aufenthaltskategorien für Ausländer.  |
| Content type          | Restriction   |
| Base type             | xs:NMTOKEN  |
| Facets                | <ul style="list-style-type: none"> <li>• Enum: shortTerm-L</li> <li>• Enum: annual-B</li> <li>• Enum: settled-C</li> <li>• Enum: crossBorder-G</li> <li>• Enum: asylumSeeker-N</li> <li>• Enum: needForProtection-S</li> <li>• Enum: NotificationProcedureForShorttermWork90Days</li> <li>• Enum: NotificationProcedureForShorttermWork120Days</li> <li>• Enum: ProvisionallyAdmittedForeigners-F</li> <li>• Enum: ResidentForeignNationalWithGainfulEmployment-Ci</li> <li>• Enum: othersNotSwiss</li> </ul>                                     |

### SimpleType: SV-AS-NumberType

|              |  |
|--------------|--|
| Translation  | 13-stellige SV-Nr Typ  |
| Content type | Restriction  |
| Base type    | xs:NMTOKEN   |
| Facets       | <ul style="list-style-type: none"> <li>• Pattern: [0-9]{3}\.[0-9]{4}\.[0-9]{4}\.[0-9]{2}</li> <li>• MaxLength: 16</li> </ul> |

### SimpleType: SalaryAmountAbsoluteType

|                   |   |
|-------------------|---|
| Translation       | Absolut-LohnbetragsTyp  |
| Short description | Absolut-LohnbetragsTyp ohne Vorzeichen  |
| Content type      | Restriction   |
| Base type         | <a href="#">„SimpleType: SalaryAmountType“</a>                                |
| Facets            | <ul style="list-style-type: none"> <li>• Pattern: [0-9]+\.[0-9]{2}</li> </ul> |

### SimpleType: SexType

|             |            |
|-------------|------------|
| Translation | Geschlecht |
|-------------|------------|

|                       |  |
|-----------------------|--|
| Short description     | F weiblich; M männlich   |
| Technical description | In diesem Element wird das Geschlecht der Person angegeben.                |
| Content type          | Restriction  |
| Base type             | xs:NMTOKEN   |
| Facets                | <ul style="list-style-type: none"> <li>Enum: F</li> <li>Enum: M</li> </ul> |

### SimpleType: SupportedRegisterOrganizationAuthenticationSchemaVersionAttributeType

|              |  |
|--------------|--|
| Translation  | Unterstützte Minor Schema Versionen  |
| Content type | Restriction  |
| Base type    | xs:decimal   |
| Facets       | <ul style="list-style-type: none"> <li>Pattern: [0-9]\.[0-9][0-9][1-9]\d*\.[0-9]</li> <li>Enum: 0.0</li> </ul> |

### SimpleType: TaxAtSourceCodeType

|                       |  |
|-----------------------|--|
| Translation           | QST-Code   |
| Short description     | QST-Code   |
| Technical description | QST-Code   |
| Content type          | Restriction  |
| Base type             | xs:NMTOKEN   |
| Facets                | <ul style="list-style-type: none"> <li>Pattern: [A-Z][0-9][Y,N]</li> </ul> |

### SimpleType: ZIP-CodeType

|                   |   |
|-------------------|---|
| Translation       | PLZ Typ   |
| Short description | Postleitzahl  |
| Content type      | Restriction   |
| Base type         | xs:string   |
| Facets            | <ul style="list-style-type: none"> <li>MinLength: 1</li> <li>MaxLength: 15</li> </ul> |