

# Technische Spezifikation eBILANZ

Version 20260306

Anforderungen Transmitter

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 1.0	06.03.2026	mas
Swissdec eBILANZ 1.0.		

## 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. Geschäftsprozesse zur Einreichung der eBilanz und eCH-0276 .....	3
2.1. Übermittlung der E-Bilanz (XBRL) .....	5
2.2. Übermittlung des eCH-0276 .....	6
3. Use Cases und zugehörige Operationen .....	7
3.1. Übersicht .....	7
3.2. Grafische Darstellung der Geschäftsprozesse und ihren Use Cases .....	7
3.3. Uses Cases und ihre Operationen .....	9
4. Ping .....	10
5. CheckInteroperability .....	13
6. DeclareRawBalanceSheet .....	17
6.1. Übermittlung der E-Bilanz (XBRL) .....	17
6.2. Initiale synchrone Meldung .....	17
6.3. Ablauf / Protokoll .....	18
6.4. Datenstruktur der initialen Meldung .....	18
6.5. Datenstruktur der initialen Antwort .....	20
6.6. Datenstruktur der Synchronize-Meldung .....	21
6.7. Datenstruktur der Synchronize-Antwort .....	24
7. DeclareBalanceSheet .....	27
7.1. Übermittlung des eCH-0276 .....	27
7.2. Initiale asynchrone Meldung .....	27
7.3. Ablauf / Protokoll .....	29
7.4. Datenstruktur der initialen Meldung .....	30
7.5. Datenstruktur der initialen Antwort .....	32
7.6. Datenstruktur der GetStatus-Meldung .....	33
7.7. Datenstruktur der GetStatus-Antwort .....	34
7.8. Datenstruktur der Synchronize-Meldung .....	36
7.9. Datenstruktur der Synchronize-Antwort .....	39
8. Use Cases .....	42
8.1. UC001 Initiale Meldung senden .....	42
8.2. UC002 Status abholen .....	44
8.3. UC003 Testmeldung kennzeichnen .....	46
8.4. UC004 Ersatzmeldung kennzeichnen .....	47
8.5. UC005 Synchronisieren .....	48
8.6. UC006 Prozesskontrolle durchführen .....	49
8.7. UC007 Fall schliessen .....	50
8.8. UC008 Stories melden und quittieren .....	51
8.9. UC009 Stories abholen und verarbeiten .....	52
8.10. UC010 Datenflusskontrolle durchführen .....	53
8.11. UC011 Completion aufrufen .....	53
8.12. UC012 Dialog Message abwickeln .....	54
8.13. UC013 Anmeldeprozess (SubscribeOrganization) .....	55
8.14. UC014 Unternehmensauthentifizierung (RegisterOrganization) .....	56
8.15. UC015 Security anwenden .....	56
8.16. UC016 Transmitter konfigurieren .....	56
8.17. UC017 Supportinformationen anzeigen .....	57
8.18. UC018 Erreichbarkeit prüfen .....	58
8.19. UC019 Interoperabilität prüfen .....	59
A. Referenzierte Dokumente .....	62
B. Glossar .....	63
C. Detailspezifikation Swissdec Unternehmens-Authentifizierung SUA .....	65
C.1. Einleitung .....	65

C.2. SUA Prozess .....	65
C.3. Technische Anforderungen an die Zertifikate .....	67
D. Detailspezifikation DialogMessages .....	70
D.1. Einleitung .....	70
D.2. Die Verwendung von DialogMessages .....	70
D.3. Standard DialogMessages .....	76
E. Detailspezifikation Completion URL .....	77
E.1. Aufbau und Verarbeitung der Completion URL .....	77
F. Bausteine .....	80
F.1. AB-01 Konfiguration .....	80
F.2. AB-02 Erreichbarkeit .....	80
F.3. AB-03 Interoperabilität .....	80
F.4. AB-04 Anmeldung .....	81
F.5. AB-05 Explizite Adressierung (direkte Adressierung) .....	81
F.6. AB-06 Implizite Adressierung (Gruppenadressierung) .....	82
F.7. AB-07 Absichtserklärung .....	83
F.8. AB-08 Identifikation eines Geschäftsvorfalles .....	83
F.9. AB-09 Ersatzmeldung .....	84
F.10. AB-10 Request und Response IDs .....	84
F.11. AB-11 Testmeldung .....	85
F.12. AB-12 Plausibilisierung .....	85
F.13. AB-13 Filterung .....	86
F.14. AB-14 Versionsmapping .....	87
F.15. AB-15 Doublettenerkennung .....	87
F.16. AB-16 Anonymisierung .....	87
F.17. AB-17 Meldungsfragmentierung .....	88
F.18. AB-18 Swissdec Unternehmensauthentifizierung (SUA) .....	88
F.19. AB-19 Archivdateien .....	88
F.20. SB-01 Transportsicherheit (HTTPS / TLS) .....	89
F.21. SB-02 Transport Authentication (TLS Mutual Authentication) .....	89
F.22. SB-03 Nutzdatenverschlüsselung (WS-Encryption) .....	90
F.23. SB-04 Nutzdatensignatur (WS-Signature) .....	90
F.24. SB-05 Unternehmens-Authentifizierung SUA Signatur (WS-Signatur) .....	90
F.25. SB-06 Nichtabstreitbarkeit .....	91
F.26. PB-01 Synchrone Verteilung .....	91
F.27. PB-02 Asynchrone Verteilung .....	91
F.28. PB-03 Synchronisierung .....	92
F.29. PB-04 Dialogisierung .....	92
F.30. PB-05 Completion .....	93
F.31. PB-06 Prozessnavigation .....	93
F.32. PB-07 Digitalisierungsbereich .....	94
F.33. PB-08 Datenanfordern .....	94
G. Technische Dokumentationen .....	96
G.1. Schema Dokumentation BalanceSheetDeclarationServiceTypes.xsd .....	96
G.2. Schema Dokumentation BalanceSheetDeclarationContainer.xsd .....	96
G.3. Schema Dokumentation SwissdecComponents.xsd .....	103

## Abbildungsverzeichnis

2.1. Prozessübersicht eBilanz .....	4
2.2. Prozessübersicht DeclareRawBalanceSheet .....	5
2.3. Prozessübersicht DeclareBalanceSheet .....	6
3.1. Use Cases Initiale Meldung .....	7
3.2. Use Cases Synchronisierung .....	8
3.3. Andere Use Cases .....	8
4.1. Schemabild Ping .....	10
4.2. Schemabild PingResponse .....	11
5.1. Schemabild CheckInteroperability .....	13
5.2. Schemabild CheckInteroperabilityResponse .....	15
6.1. Synchroner Meldungsprozess .....	17
6.2. DeclareRawBalanceSheet Protokollstatus .....	18
6.3. Schemabild DeclareRawBalanceSheet .....	18
6.4. Schemabild Addressee .....	19
6.5. Schemabild DeclareRawBalanceSheetResponse .....	20
6.6. Schemabild SynchronizeDeclareRawBalanceSheet .....	21
6.7. Schemabild CaseContext .....	23
6.8. Schemabild SynchronizeDeclareRawBalanceSheetResponse .....	24
7.1. Asynchroner Meldungsprozess .....	28
7.2. DeclareBalanceSheet Protokollstatus .....	29
7.3. Schemabild DeclareBalanceSheet .....	30
7.4. Schemabild Addressee .....	31
7.5. Schemabild DeclareBalanceSheetResponse .....	32
7.6. Schemabild GetStatusFromDeclareBalanceSheet .....	33
7.7. Schemabild GetStatusFromDeclareBalanceSheetResponse .....	34
7.8. Schemabild Addressee .....	35
7.9. Schemabild SynchronizeDeclareBalanceSheet .....	36
7.10. Schemabild CaseContext .....	38
7.11. Schemabild SynchronizeDeclareBalanceSheetResponse .....	39
D.1. DialogMessage XML-Schema Struktur .....	71
D.2. Vorgehen bei der Darstellung der DialogMessage (Aktivitäts Diagramm) .....	74
D.3. Paragraph/Answer/DateTime-Element, andere Answer/<Type>-Elemente entsprechend .....	75
D.4. Struktur des DialogMessage/Previous-Elementes .....	75
E.1. URL-Encoding in der Completion .....	78
E.2. Aufbau URL gemäss RFC3986 .....	79
F.1. AddresseeType .....	82

## Tabellenverzeichnis

1. Verbindlichkeit von Anforderungen .....	iv
1.1. Beteiligte am Swissdec-Übermittlungsprozess .....	1
2.1. Kommunikationsmatrix über alle Geschäftsprozesse und Systeme .....	4
3.1. Use Cases und Operationen .....	9
4.1. Feldbeschreibungen Ping .....	10
4.2. Feldbeschreibungen UserAgent .....	11
4.3. Feldbeschreibungen PingResponse .....	11
4.4. Feldbeschreibungen UserAgent .....	12
5.1. Feldbeschreibungen CheckInteroperability .....	13
5.2. Feldbeschreibungen UserAgent .....	14
5.3. Feldbeschreibungen CheckInteroperabilityResponse .....	15
5.4. Feldbeschreibungen UserAgent .....	16
6.1. Feldbeschreibungen DeclareRawBalanceSheet .....	18
6.2. Feldbeschreibungen RequestContext .....	19
6.3. Feldbeschreibungen Job .....	19
6.4. Feldbeschreibungen Addressee .....	20
6.5. Feldbeschreibungen DeclareRawBalanceSheetResponse .....	20
6.6. Feldbeschreibungen ResponseContext .....	20
6.7. Feldbeschreibungen SynchronizeDeclareRawBalanceSheet .....	21
6.8. Feldbeschreibungen RequestContext .....	21
6.9. Feldbeschreibungen Sender .....	22
6.10. Feldbeschreibungen Addressee .....	22
6.11. Feldbeschreibungen Case .....	22
6.12. Feldbeschreibungen CaseContext .....	23
6.13. Feldbeschreibungen SynchronizeDeclareRawBalanceSheetResponse .....	24
6.14. Feldbeschreibungen ResponseContext .....	24
6.15. Feldbeschreibungen Error .....	25
6.16. Feldbeschreibungen SynchronizeDeclareRawBalanceSheetConsumer .....	25
6.17. Feldbeschreibungen AddresseeContext .....	25
6.18. Feldbeschreibungen Addressee .....	26
6.19. Feldbeschreibungen Case .....	26
7.1. Feldbeschreibungen DeclareBalanceSheet .....	30
7.2. Feldbeschreibungen RequestContext .....	30
7.3. Feldbeschreibungen Job .....	31
7.4. Feldbeschreibungen Addressee .....	31
7.5. Feldbeschreibungen DeclareBalanceSheetResponse .....	32
7.6. Feldbeschreibungen ResponseContext .....	32
7.7. Feldbeschreibungen GetStatusFromDeclareBalanceSheet .....	33
7.8. Feldbeschreibungen RequestContext .....	33
7.9. Feldbeschreibungen GetStatusFromDeclareBalanceSheetResponse .....	34
7.10. Feldbeschreibungen Addressee .....	35
7.11. Feldbeschreibungen Error .....	35
7.12. Feldbeschreibungen Success .....	36
7.13. Feldbeschreibungen SynchronizeDeclareBalanceSheet .....	36
7.14. Feldbeschreibungen RequestContext .....	37
7.15. Feldbeschreibungen Sender .....	37
7.16. Feldbeschreibungen Addressee .....	37
7.17. Feldbeschreibungen Case .....	37
7.18. Feldbeschreibungen CaseContext .....	38
7.19. Feldbeschreibungen SynchronizeDeclareBalanceSheetResponse .....	39
7.20. Feldbeschreibungen ResponseContext .....	39
7.21. Feldbeschreibungen Error .....	40
7.22. Feldbeschreibungen SynchronizeDeclareBalanceSheetConsumer .....	40
7.23. Feldbeschreibungen AddresseeContext .....	40



7.24. Feldbeschreibungen Adressee .....	41
7.25. Feldbeschreibungen Case .....	41
8.1. UC001 Initiale Meldung senden .....	42
8.2. RequestContextType .....	43
8.3. UserAgentType .....	43
8.4. UC002 Status abholen .....	44
8.5. UC003 Testmeldung kennzeichnen .....	46
8.6. UC003 Testmeldung kennzeichnen .....	47
8.7. UC005 Synchronisieren .....	48
8.8. UC006 Prozesskontrolle durchführen .....	49
8.9. UC007 Fall schliessen .....	50
8.10. UC008 Stories melden und quittieren .....	51
8.11. UC009 Stories abholen und verarbeiten .....	52
8.12. UC011 Completion aufrufen .....	53
8.13. UC012 Dialog Message abwickeln .....	54
8.14. UC013 Anmeldungsprozess (SubscribeOrganization) .....	55
8.15. UC017 Supportinformationen anzeigen .....	57
8.16. UC018 Erreichbarkeit prüfen .....	58
8.17. UC019 Interoperabilität prüfen .....	59
8.18. Vorbedingungen (Transmitter) .....	60
8.19. Auswertung und Antwort Distributor .....	60
8.20. Auswertung Transmitter .....	61
F.1. AB-01 Konfiguration .....	80
F.2. AB-02 Erreichbarkeit .....	80
F.3. AB-03 Interoperabilität .....	80
F.4. AB-04 Anmeldung .....	81
F.5. AB-05 Explizite Adressierung (direkte Adressierung) .....	81
F.6. AB-06 Implizite Adressierung (Gruppenadressierung) .....	82
F.7. AB-07 Absichtserklärung .....	83
F.8. AB-08 Identifikation eines Geschäftsvorfalles .....	83
F.9. AB-09 Ersatzmeldung .....	84
F.10. AB-10 Request und Response IDs .....	84
F.11. AB-11 Testmeldung .....	85
F.12. AB-12 Plausibilisierung .....	85
F.13. AB-13 Filterung .....	86
F.14. AB-14 Versionsmapping .....	87
F.15. AB-15 Doublettenerkennung .....	87
F.16. AB-16 Anonymisierung .....	87
F.17. AB-18 Swissdec Unternehmensauthentifizierung (SUA) .....	88
F.18. AB-19 Archivdateien .....	88
F.19. SB-01 Transportsicherheit (HTTPS / TLS) .....	89
F.20. SB-02 Transport Authentication (TLS Mutual Authentication) .....	89
F.21. SB-03 Nutzdatenverschlüsselung (WS-Encryption) .....	90
F.22. SB-04 Nutzdatensignatur (WS-Signature) .....	90
F.23. SB-05 Unternehmens-Authentifizierung SUA Signatur (WS-Signatur) .....	90
F.24. SB-06 Nichtabstreitbarkeit .....	91
F.25. PB-01 Synchrone Verteilung .....	91
F.26. PB-02 Asynchrone Verteilung .....	91
F.27. PB-03 Synchronisierung .....	92
F.28. PB-04 Dialogisierung .....	92
F.29. PB-05 Completion .....	93
F.30. PB-06 Prozessnavigation .....	93
F.31. PB-07 Digitalisierungsbereich .....	94
F.32. PB-08 Datenanfordern .....	94

## 1 Einleitung

Dieses Dokument richtet sich an Software-Hersteller und -Entwickler, die für die technische Umsetzung eines Swissdec-Standards verantwortlich sind. Es enthält funktionale, technische und zusätzliche Anforderungen an Transmitter, die an den Swissdec-Distributor übermitteln wollen. Ein Transmitter wird dazu verwendet, Meldungen aus einem Sendersystem via Distributor an einen oder mehrere Endreceiver elektronisch zu versenden, damit es im entsprechenden Empfängersystem verarbeitet werden kann.

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

Sendersystem	Beim Sendersystem handelt es sich um ein System, welches 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, welches 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 Transmitter und Distributor, nicht aber jene zwischen Distributor und Endreceiver.**

Der Transmitter wird im Rahmen der Swissdec-Zertifizierung auf seine Funktionalität geprüft. Die Testfälle zur Zertifizierung finden sich unter [\[TFBASIS\]](#)

### 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 Use Cases 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 Geschäftsprozesse zur Einreichung der eBilanz und eCH-0276

Ein klar definierter Geschäftsprozess ist essenziell für die Entwicklung und Implementierung einer Schnittstelle, da er die Grundlage für eine nahtlose und effiziente Kommunikation zwischen Systemen bildet. Ohne eine präzise Beschreibung des Geschäftsprozesses fehlen den beteiligten Systemen die notwendigen Informationen über die Reihenfolge der Abläufe, die erwarteten Eingaben und Ausgaben sowie die Bedingungen für Übergänge zwischen verschiedenen Zuständen. Dies kann zu Missverständnissen, Fehlinterpretationen und Inkompatibilitäten führen. Ein definierter Geschäftsprozess gewährleistet, dass alle Beteiligten – von Entwicklern bis hin zu Endnutzern – ein gemeinsames Verständnis darüber haben, wie die Schnittstelle genutzt werden soll. Darüber hinaus ermöglicht er eine klare Trennung von Verantwortlichkeiten, fördert die Wiederverwendbarkeit von Komponenten und unterstützt die Automatisierung von Prozessen, was letztlich zu einer höheren Effizienz, Skalierbarkeit und Zuverlässigkeit der Schnittstelle beiträgt.

Die Swissdec-Schnittstelle "eBilanz und eCH-Standard 0276" sieht folgende Geschäftsprozesse vor:

1. Übermittlung der E-Bilanz aus dem ERP-System (resp. Buchhaltungssoftware) an eine kantonseigene Deklarationslösung via den Distributor beschrieben in [Kapitel 6, DeclareRawBalanceSheet](#) .
2. Übermittlung des eCH-0276 aus einer Deklarationslösung über den Swissdec Distributor an die Haupt- und Nebensteuerdomizile beschrieben in [Kapitel 7, DeclareBalanceSheet](#) .

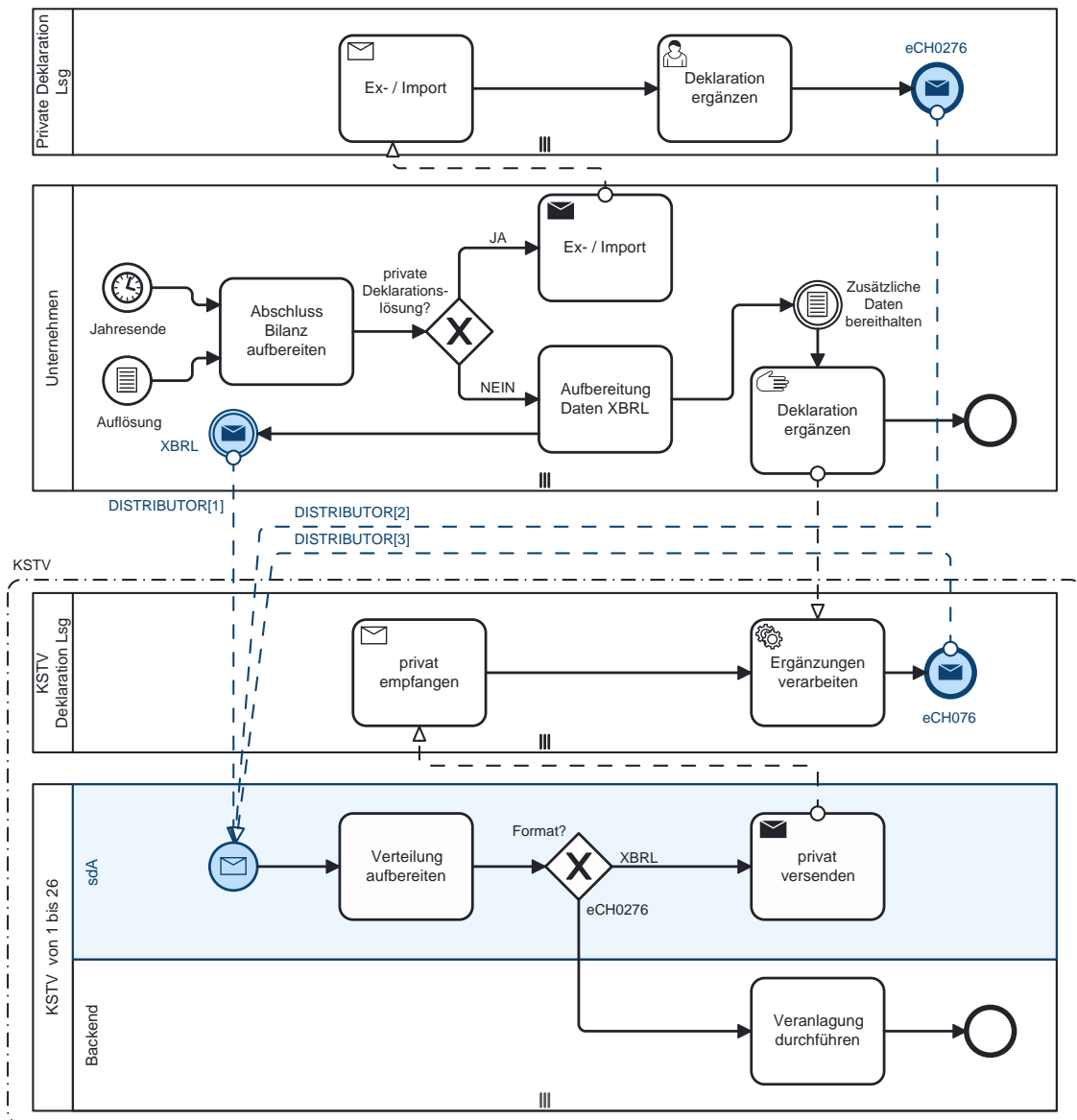


Abbildung 2.1. Prozessübersicht eBilanz

Nr.	Von	Nach	Operation	Datenformat	Bemerkung
DISTRIBU-TOR[1]	ERP	SwissdecAdapter	<a href="#">Kapitel 6, DeclareRawBalanceSheet</a>	XBRL	Einreichen der eBilanz
-	SwissdecAdapter	Kantoneigene Deklarationslösung	-	XBRL	Der SwissdecAdapter leitet die Empfangen eBilanz an die kantoneigene Deklarationslösung weiter
DISTRIBU-TOR[3]	Kantoneigene Deklarationslösung	SwissdecAdapter	<a href="#">Kapitel 7, DeclareBalanceSheet</a>	eCH-0276	Die kantoneigene Deklarationslösung sendet den eCH-0276

Nr.	Von	Nach	Operation	Datenformat	Bemerkung
					via Distribu- tor an Haupt- und Nebensteu- erdomizile
DISTRIBU- TOR[2]	Private Deklarati- onslösung	SwissdecAdapter	<a href="#">Kapitel 7,</a> <a href="#">DeclareBalan- ceSheet</a>	eCH-0276	Die Deklarations- lösung sendet den eCH-0276 via Distribu- tor an Haupt- und Nebensteu- erdomizile

Tabelle 2.1. Kommunikationsmatrix über alle Geschäftsprozesse und Systeme

## Anmerkung

Aus vertragsrechtlichen und statuarischen Gründen seitens Swissdec können aktuell nur kantonseigene Deklarationslösungen die E-Bilanz via Distributor elektronisch empfangen. Für die Anbindung von privaten Deklarationslösungen an den Distributor müssen zuerst die vertraglichen Grundlagen geschaffen werden.

## 2.1 Übermittlung der E-Bilanz (XBRL)

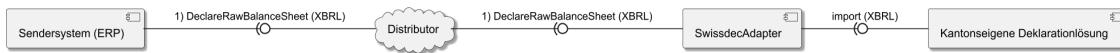


Abbildung 2.2. Prozessübersicht DeclareRawBalanceSheet

Involvierte Systeme:

- Sendersystem (ERP) als Transmitter
- SwissdecAdapter der KSTV als Receiver
- Kantonseigene Deklarationslösung als Empfänger

## 2.2 Übermittlung des eCH-0276

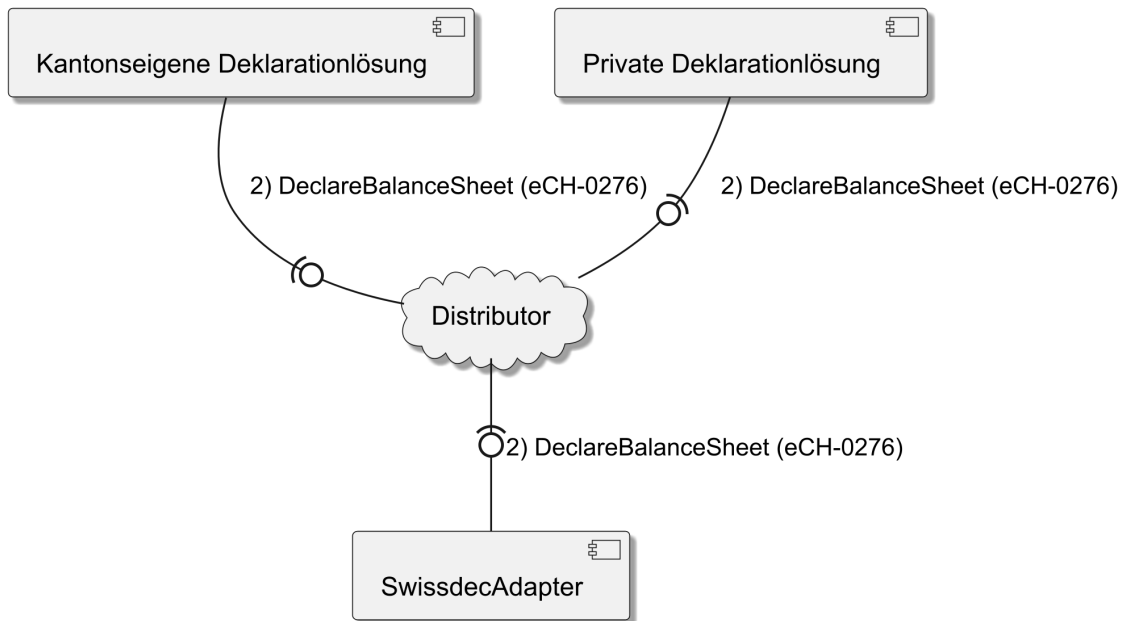


Abbildung 2.3. Prozessübersicht DeclareBalanceSheet

Involvierte Systeme:

- Kantonseigene Deklarationslösung Transmitter
- Private Deklarationslösung Transmitter
- SwissdecAdapter der KSTV als Receiver

## 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 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 Use Cases [Abschnitt 8.2, „UC002 Status abholen“](#) und [Abschnitt 8.12, „UC012 Dialog Message abwickeln“](#) werden von eBilanz nicht unterstützt.

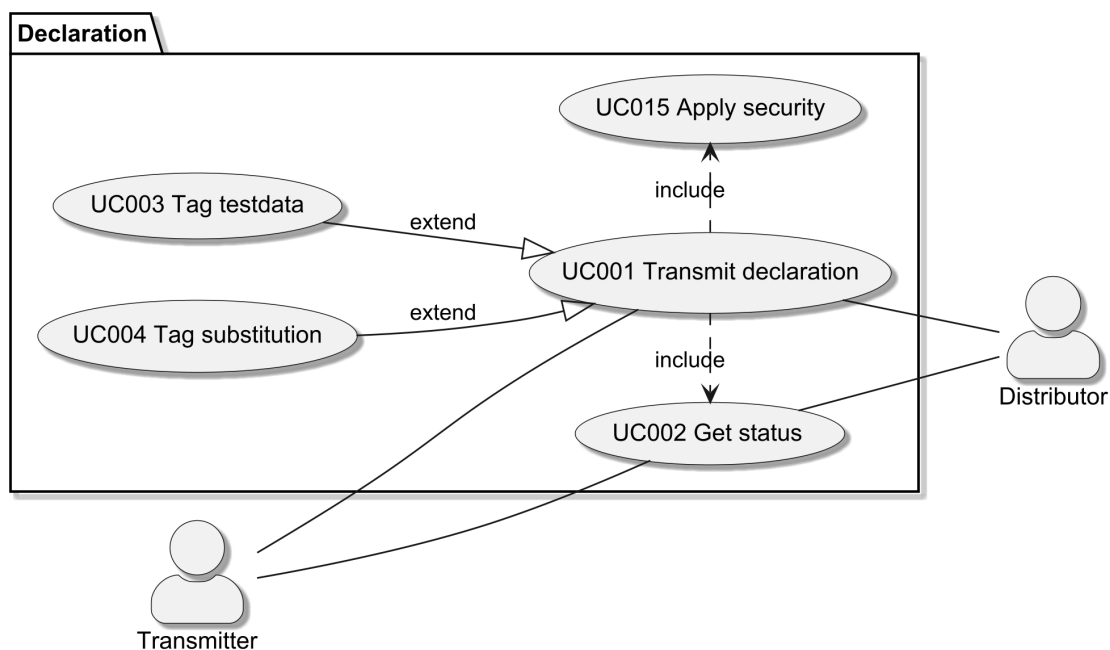


Abbildung 3.1. Use Cases Initiale Meldung



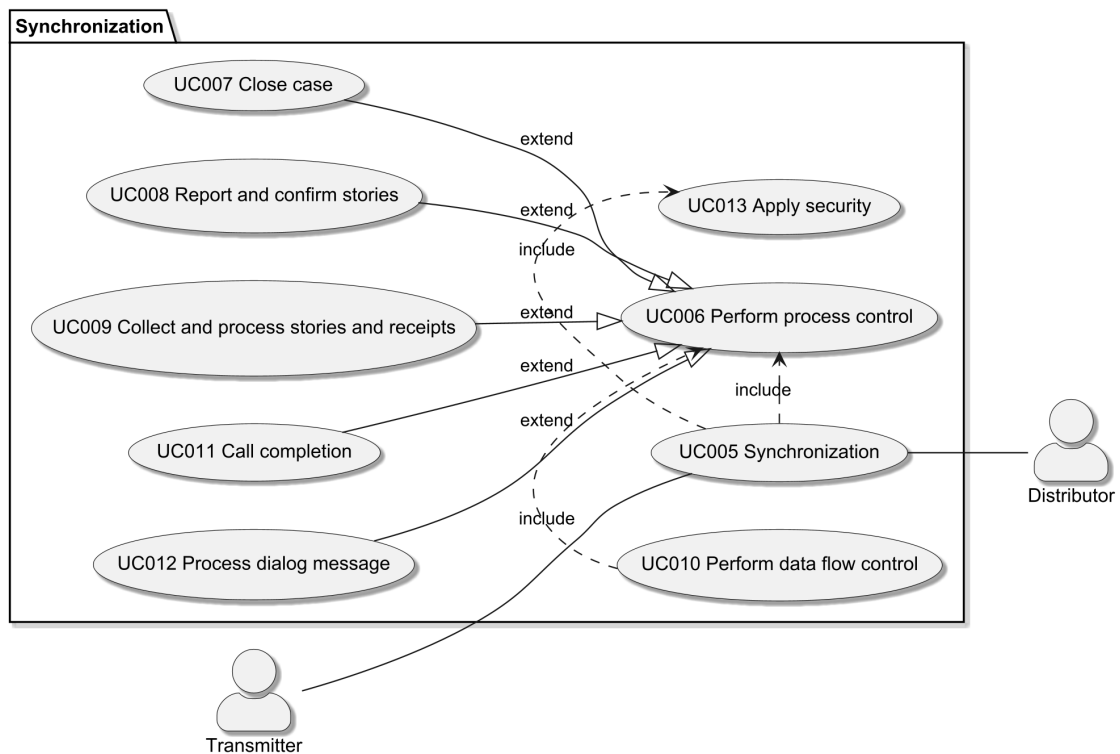


Abbildung 3.2. Use Cases Synchronisierung

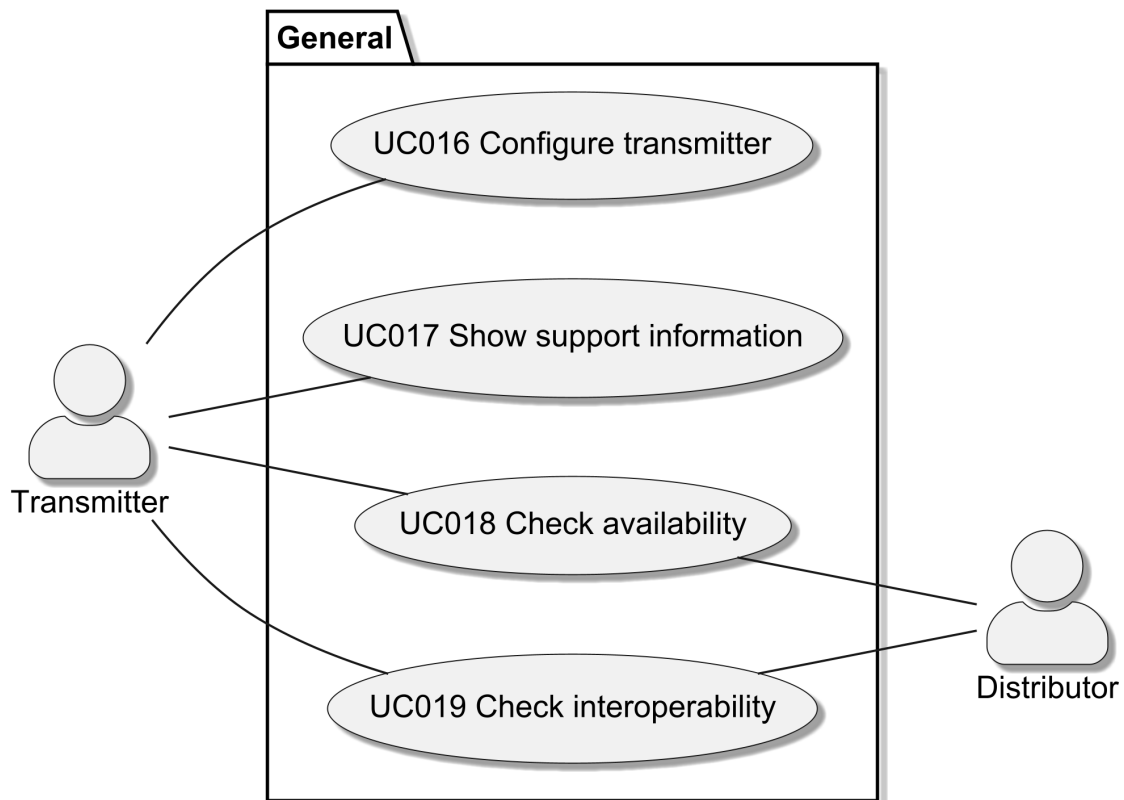


Abbildung 3.3. Andere Use Cases

### 3.3 Uses Cases und ihre Operationen

In der nachfolgenden Tabelle werden die Operationen des Finanzstandards 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>• DeclareRawBalanceSheet: Operation zur Übermittlung der E-Bilanz (XBRL)</li> <li>• DeclareRawBalanceSheetResponse: Rückmeldung zur Übermittlung der E-Bilanz (XBRL)</li> <li>• DeclareBalanceSheet: Operation zur Übermittlung des eCH-0276</li> <li>• DeclareBalanceSheetResponse: Operation zur Übermittlung des eCH-0276</li> <li>• BalanceSheetDeclarationFault: Rückmeldung im Fehlerfall</li> </ul>
UC002 Status abholen	<ul style="list-style-type: none"> <li>• GetStatusFromDeclareBalanceSheet: Statusabfrage der Übermittlung</li> <li>• GetStatusFromDeclareBalanceSheetResponse: Rückmeldung der Statusabfrage</li> <li>• BalanceSheetDeclarationFault: Rückmeldung im Fehlerfall</li> </ul>
UC005 Synchronisieren	<ul style="list-style-type: none"> <li>• SynchronizeDeclareRawSheetBalance: Aktualisierung des Ereignisses</li> <li>• SynchronizeDeclareRawSheetBalanceResponse: Rückmeldung der Aktualisierungsanfrage</li> <li>• SynchronizeDeclareBalanceSheet: Aktualisierung des Ereignisses</li> <li>• SynchronizeDeclareBalanceSheetResponse: Rückmeldung der Aktualisierungsanfrage</li> <li>• BalanceSheetDeclarationFault: Rückmeldung im Fehlerfall</li> </ul>
UC018 Erreichbarkeit prüfen	<ul style="list-style-type: none"> <li>• Ping</li> <li>• PingResponse</li> </ul>
UC019 Interoperabilität prüfen	<ul style="list-style-type: none"> <li>• CheckInteroperability</li> <li>• CheckInteroperabilityResponse</li> </ul>

Tabelle 3.1. Use Cases und Operationen

## 4 Ping

Die Erreichbarkeit des Distributors **muss** geprüft werden. Dazu wird eine einfache Anfrage an den Distributor geschickt. Die Antwort des Distributors bestätigt die Erreichbarkeit.

Mit dem Ping-Aufruf wird vom Transmitter wie auch vom Distributor die Systemzeit übermittelt, sodass es möglich ist, die Zeiten der an der Übermittlung beteiligten Systeme zu vergleichen. Da der Ping weder signiert noch verschlüsselt ist, ist es mit diesem Zeitvergleich möglich, Timestamp-Probleme aufzudecken.

Dieser Use Case dient der Qualitätssicherung bei der Installation und im Supportfall. Er **muss** stets manuell ausgeführt werden und **darf nicht** automatisiert und nicht in regelmässigen Abständen aufgerufen werden. Der Ping *kann* vom Endbenutzer ausgelöst werden. Er **muss** im Supportfall von der supportverantwortlichen Person aufgerufen werden können. Dies *kann* auch über eine, für den Endbenutzer nicht sichtbare, Administrationsoberfläche geschehen.

Die Anforderungen sind im [Abschnitt 8.18, „UC018 Erreichbarkeit prüfen“](#) beschrieben.

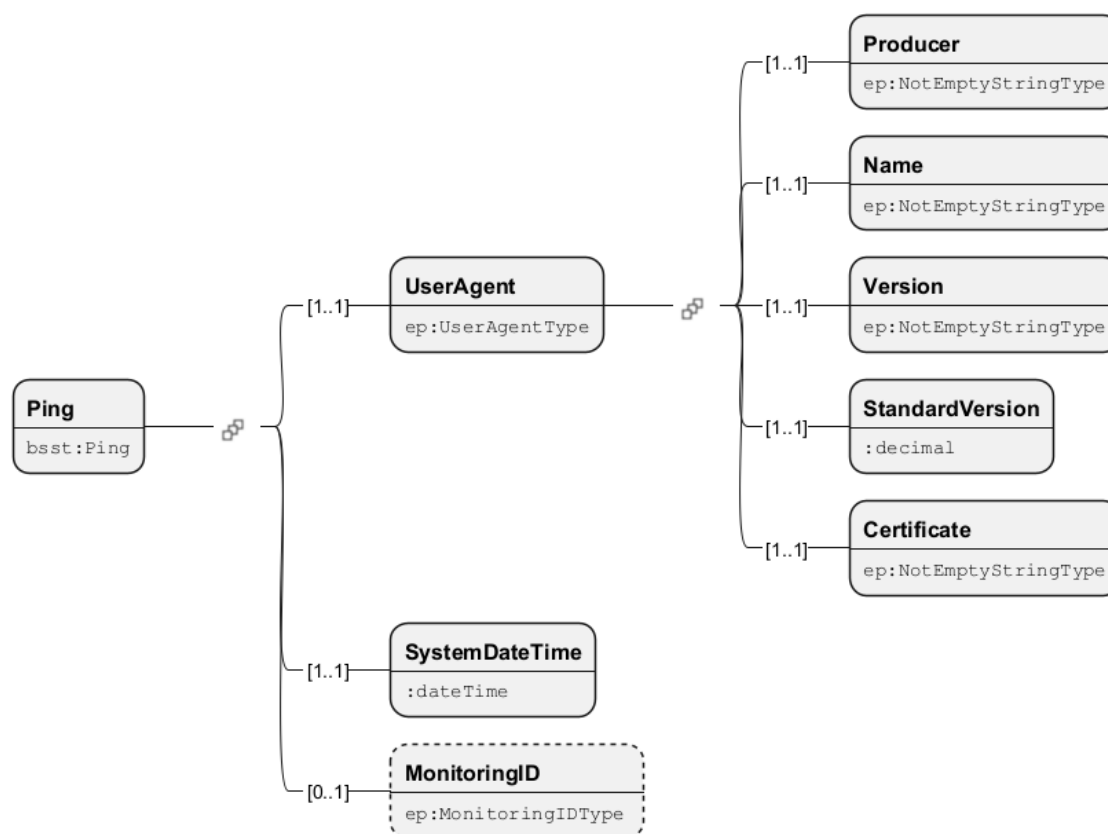


Abbildung 4.1. Schemabild Ping

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätssicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<a href="#">ep UserAgentType</a>
SystemDateTime	Aktuelle Systemzeit	xs dateTime
MonitoringID	Die KontrollID wird vor allem in der Referenzapplikation zur Aufteilung der Daten verwendet	<a href="#">ep MonitoringIDType</a>

Feldname	Beschreibung	Typ

Tabelle 4.1. Feldbeschreibungen Ping

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. Feldbeschreibungen UserAgent

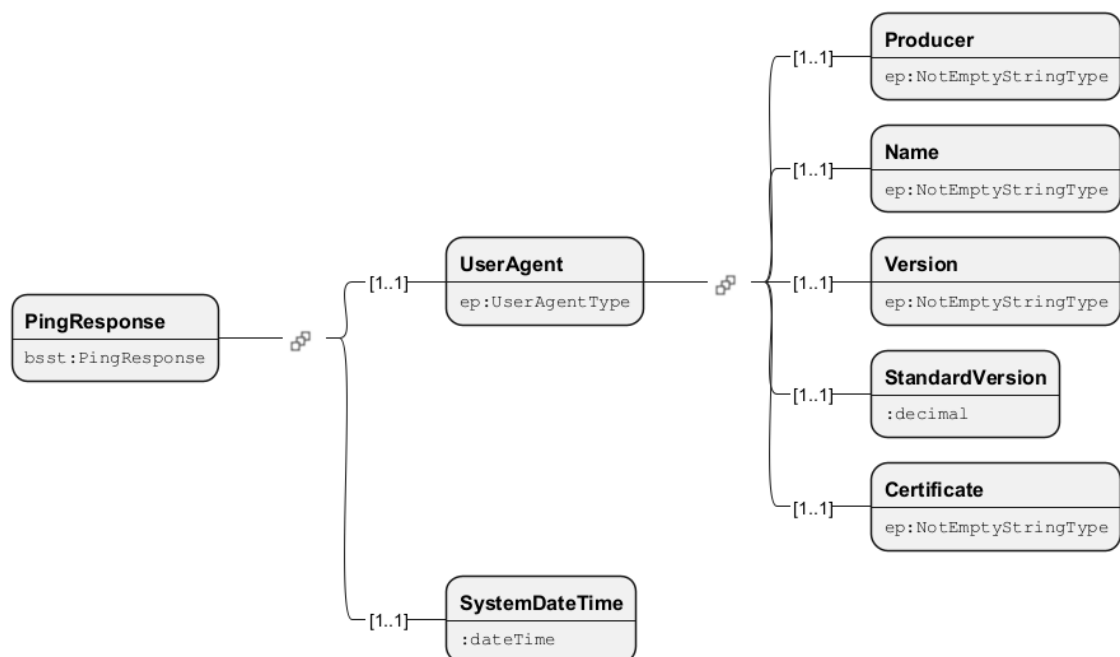


Abbildung 4.2. Schemabild PingResponse

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<a href="#">ep UserAgentType</a>

Feldname	Beschreibung	Typ
SystemDateTime	Aktuelle Systemzeit	xs dateTime

Tabelle 4.3. Feldbeschreibungen PingResponse

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 4.4. Feldbeschreibungen UserAgent

## 5 CheckInteroperability

Eine spezielle Meldung wird an den Distributor gesendet, um die Interoperabilität (z.B. Encoding, Marshalling, Zeitangaben etc.) zwischen Transmitter und Distributor zu prüfen.

Damit die Interoperabilität zwischen einem Transmitter und dem Distributor überprüft werden kann, **muss** der Transmitter einen „CheckInteroperabilityRequest“ absetzen können. Dieser **muss** manuell ausgeführt werden.

Der Interoperabilitätstest wird zu Entwicklungszwecken, bei der Installation und im Supportfall verwendet, um die Interoperabilität zwischen einem Transmitter und dem Distributor zu gewährleisten. Die grössten zu erwartenden Schwierigkeiten liegen dabei in den Bereichen der Codierung von Zeichenketten (Encoding) und der Interpretation von Fließkommazahlen. Beide Systeme (Transmitter und Distributor) **müssen** dabei bestimmte Auswertungen vornehmen, um bei einem eventuellen Fehler auf den Verursacher schliessen zu können.

Der Interoperabilitätstest **darf nicht** im Rahmen eines automatisierten Prozesses routinemässig aufgerufen werden.

Die Anforderungen sind im [Abschnitt 8.19, „UC019 Interoperabilität prüfen“](#) beschrieben.

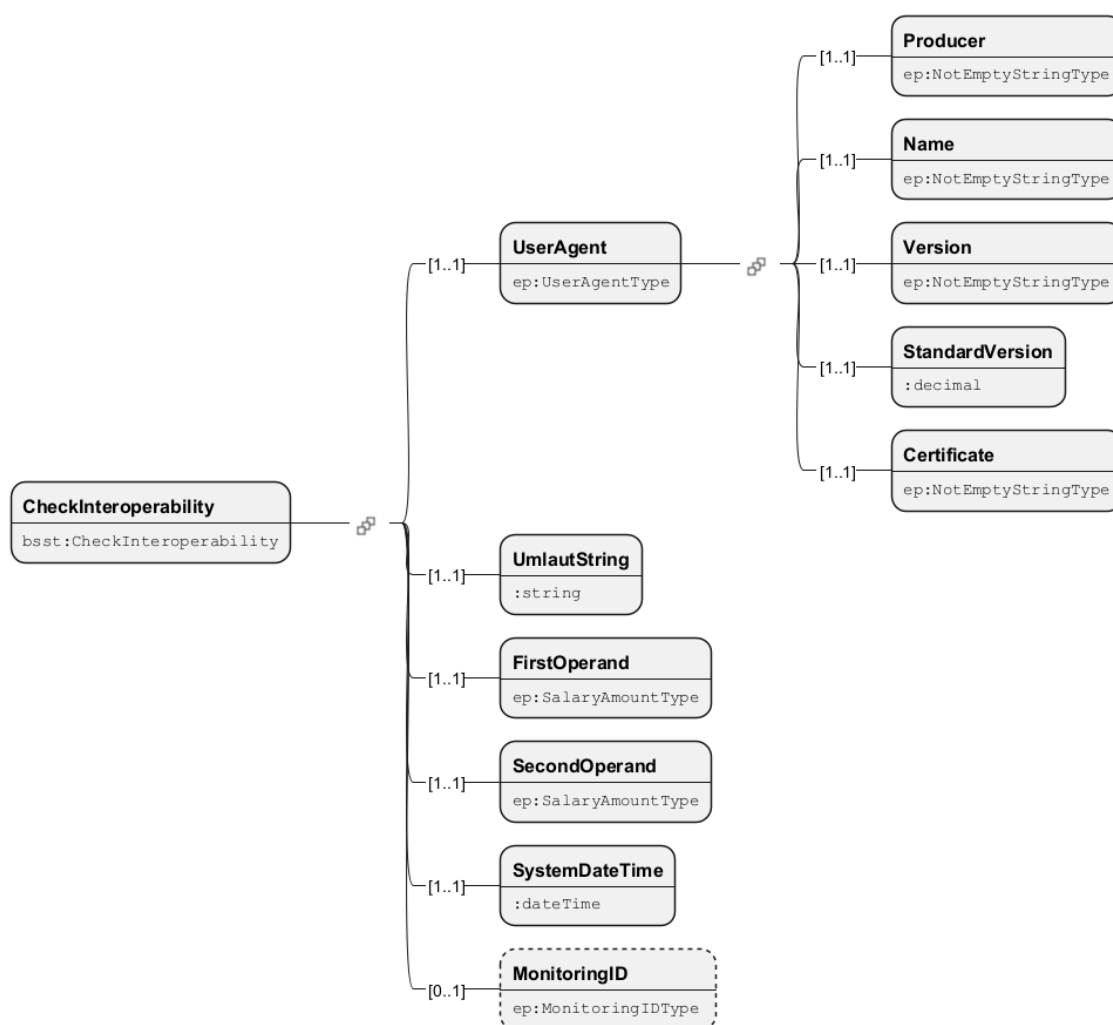


Abbildung 5.1. Schemabild CheckInteroperability

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<a href="#">ep UserAgentType</a>

Feldname	Beschreibung	Typ
UmlautString	Anordnung von Umlauten zum Testen des Encodings bei Sonderzeichen.	xs string
FirstOperand	1. Operand wird mit 2. Operand zusammengerechnet zum Testen, dass keine Datentyp-Fehler passieren.	<a href="#">ep_SalaryAmountType</a>
SecondOperand	2. Operand wird mit 1. Operand zusammengerechnet zum Testen, dass keine Datentyp-Fehler passieren.	<a href="#">ep_SalaryAmountType</a>
SystemDateTime	Aktuelle Systemzeit	xs dateTime
MonitoringID	Die KontrollID wird vor allem in der Referenzapplikation zur Aufteilung der Daten verwendet	<a href="#">ep_MonitoringIDType</a>

Tabelle 5.1. Feldbeschreibungen CheckInteroperability

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 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_NotEmptyStringType</a>

Tabelle 5.2. Feldbeschreibungen UserAgent

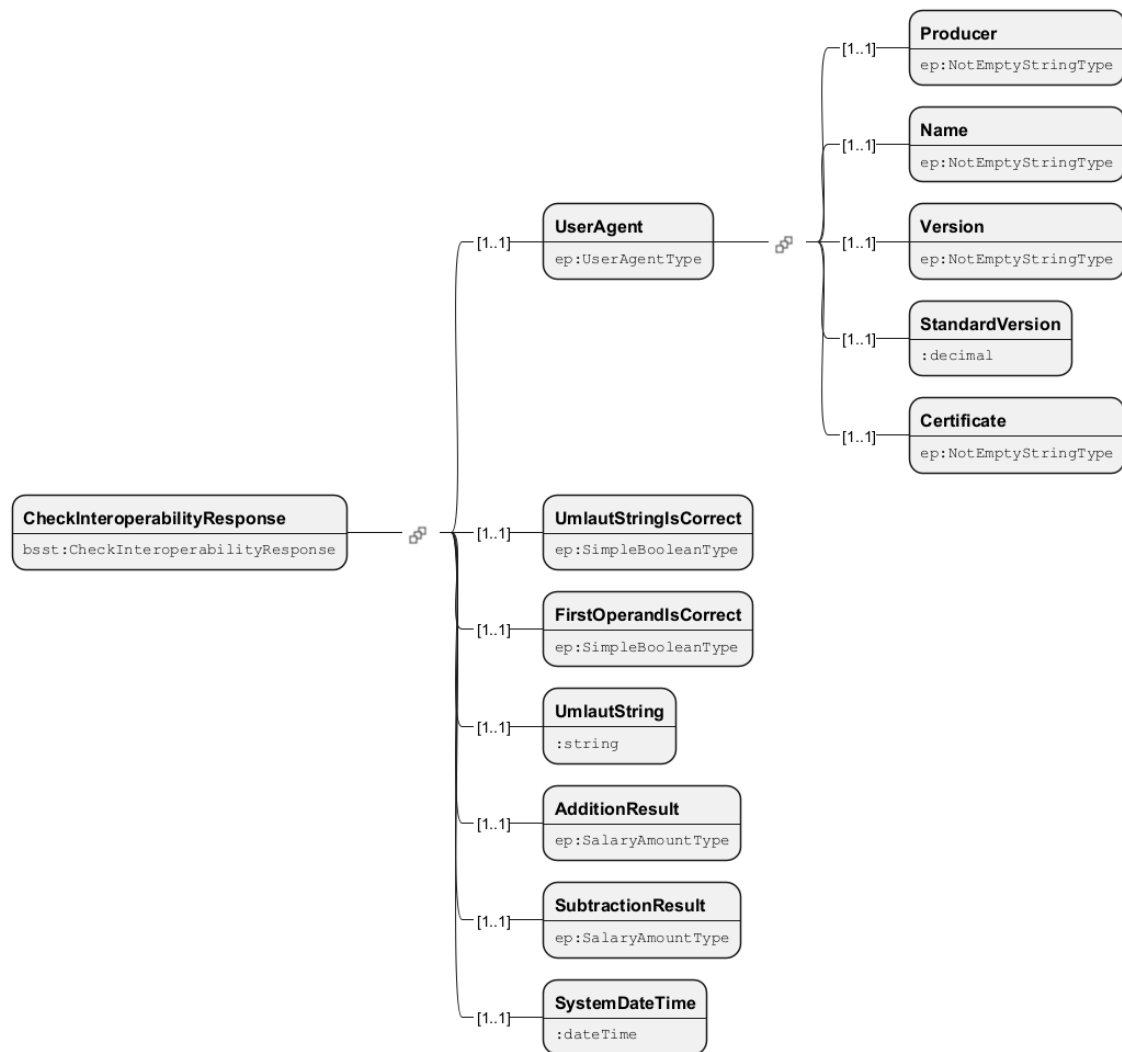


Abbildung 5.2. Schemabild CheckInteroperabilityResponse

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<a href="#">ep UserAgentType</a>
UmlautStringsCorrect	Ergebnis der Prüfung des UmlautStrings	<a href="#">ep SimpleBooleanType</a>
FirstOperandIsCorrect	Ergebnis des Vergleichs der Operanden	<a href="#">ep SimpleBooleanType</a>
UmlautString	Anordnung von Umlauten zum Testen des Encodings bei Sonderzeichen.	xs string
AdditionResult	Summe der beiden Operanden des Interoperabilitätstests	<a href="#">ep SalaryAmountType</a>
SubtractionResult	Ergebnis des Subtraktionstests in der Interoperabilitätsprüfung	<a href="#">ep SalaryAmountType</a>
SystemDateTime	Aktuelle Systemzeit	xs dateTime



Feldname	Beschreibung	Typ

Tabelle 5.3. Feldbeschreibungen CheckInteroperabilityResponse

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 5.4. Feldbeschreibungen UserAgent

## 6 DeclareRawBalanceSheet

### 6.1 Übermittlung der E-Bilanz (XBRL)

Mit der vorliegende Operation kann eine eBilanz mit dem XBRL-Format aus dem ERP-System (resp. Buchhaltungssoftware) an eine kantonseigene Deklarationslösung übermittelt werden. Der Prozess wird in [Abschnitt 2.1, „Übermittlung der E-Bilanz \(XBRL\)“](#) beschrieben.

### 6.2 Initiale synchrone Meldung

Die Aktivitätsdiagramme sind ohne den Endreceiver dargestellt. Die wesentliche Transmitterfunktionalität *sol* dargestellt werden. Die Anzeigen erfolgen wahlweise in Transmitter oder im ERP.

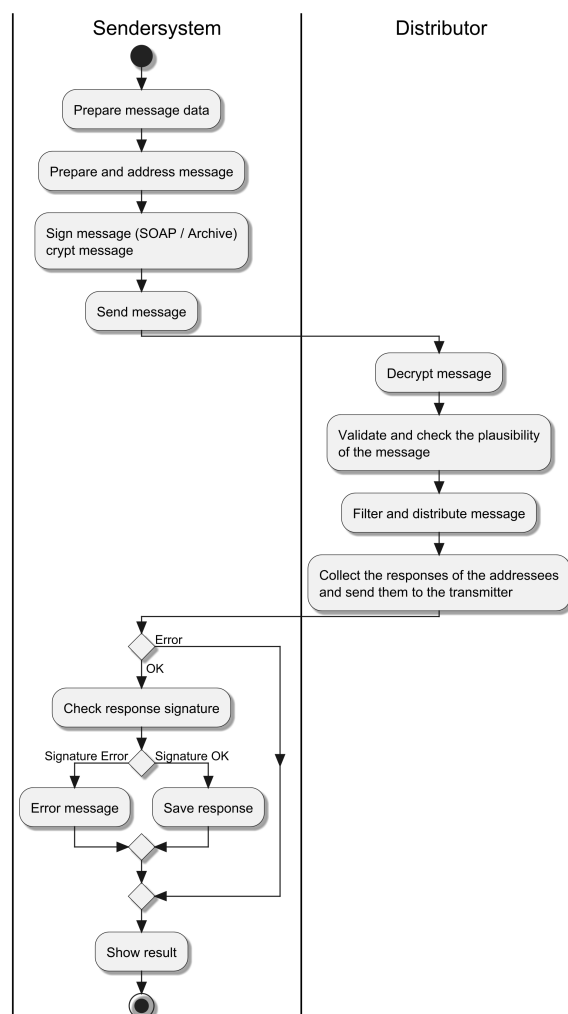


Abbildung 6.1. Synchroner Meldungsprozess

### 6.3 Ablauf / Protokoll

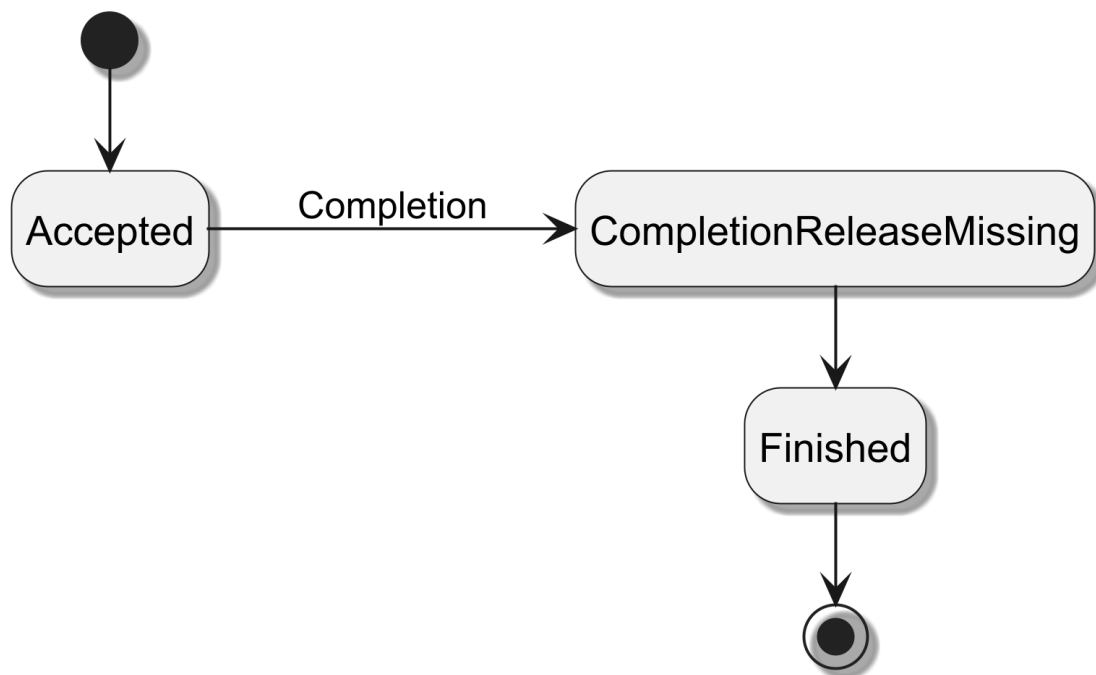


Abbildung 6.2. DeclareRawBalanceSheet Protokollstatus

### 6.4 Datenstruktur der initialen Meldung

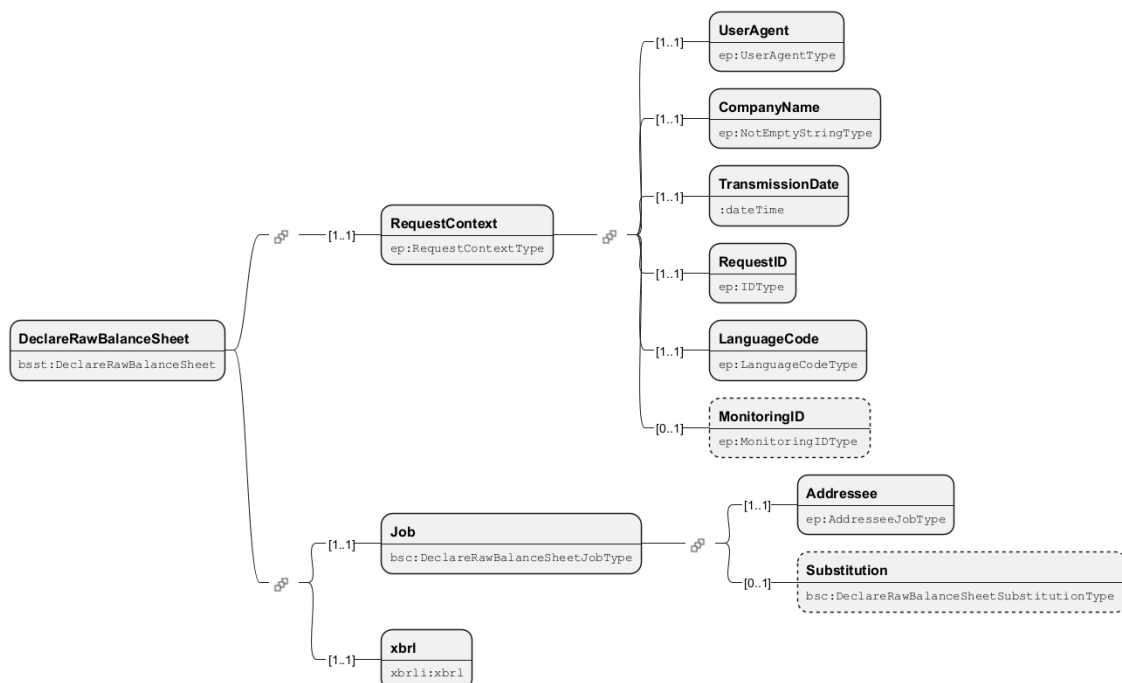


Abbildung 6.3. Schemabild DeclareRawBalanceSheet

Feldname	Beschreibung	Typ
RequestContext	Der RequestContext enthält Informationen zum Zeitpunkt der Übermittlung, dem Absender und der Sprache.	<a href="#">ep_RequestContext-Type</a>

Feldname	Beschreibung	Typ
Job	Steuerungselement für die Verarbeitung der Meldung.	<a href="#">bsc_DeclareRawBalanceSheetJobType</a>
xbml		xbml xbrl

Tabelle 6.1. Felddescriptions DeclareRawBalanceSheet

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätssicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<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>
MonitoringID	Monitoring Identifikation	<a href="#">ep_MonitoringIDType</a>

Tabelle 6.2. Felddescriptions RequestContext

Feldname	Beschreibung	Typ
Addressee	Auflistung der Endempfänger, die die übermittelten Daten erhalten sollen.	<a href="#">ep_AddresseeJobType</a>
Substitution	Entschädigung Job	<a href="#">bsc_DeclareRawBalanceSheetSubstitutionType</a>

Tabelle 6.3. Felddescriptions Job

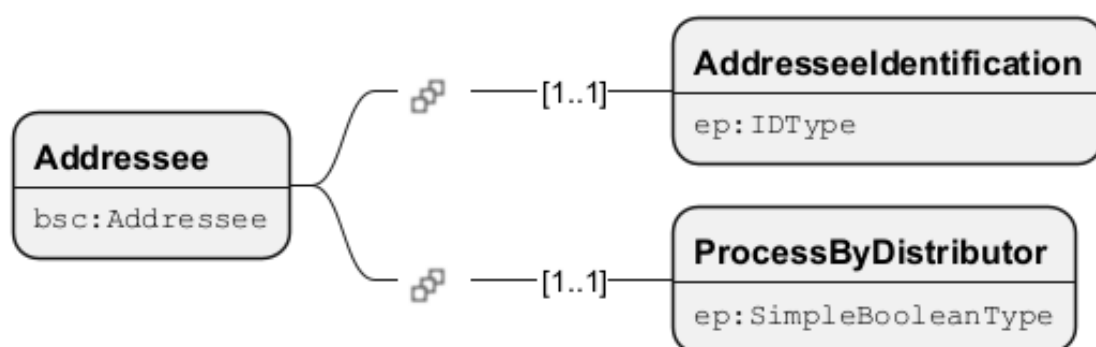


Abbildung 6.4. Schemabild Addressee

Feldname	Beschreibung	Typ
AddresseeIdentification	Identifikation des Adressaten	<a href="#">ep IDType</a>
ProcessByDistributor	Dieses Flag entscheidet, ob an den Distributor übermittelte Daten von ihm verarbeitet oder ignoriert werden sollen. Steht 'false', wird der Distributor keine Verteilung vornehmen und die gesendeten Inhalte für diese Institution ignorieren.	<a href="#">ep SimpleBooleanType</a>
@addresseeID	Referenz auf Adressat	<a href="#">ep InstanceRefIDType</a>

Tabelle 6.4. Feldbeschreibungen Addressee

## 6.5 Datenstruktur der initialen Antwort

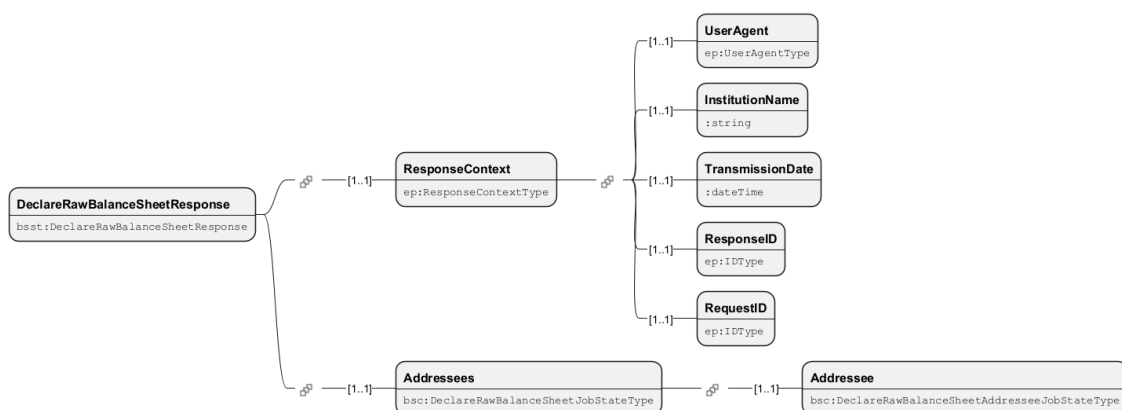


Abbildung 6.5. Schemabild DeclareRawBalanceSheetResponse

Feldname	Beschreibung	Typ
ResponseContext	Der ResponseContext enthält Informationen zum Zeitpunkt der Übermittlung, dem Absender und der Sprache.	<a href="#">ep ResponseContextType</a>
Addressees	Initialer Status des Falls.	<a href="#">bsc DeclareRawBalanceSheetJobStateType</a>

Tabelle 6.5. Feldbeschreibungen DeclareRawBalanceSheetResponse

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<a href="#">ep UserAgentType</a>
InstitutionName	Institutionsname	xs string
TransmissionDate	Übertragungszeitpunkt	xs dateTime
ResponseID	AntwortID	<a href="#">ep IDType</a>

Feldname	Beschreibung	Typ
RequestID	AuftragsID	<a href="#">ep IDType</a>

Tabelle 6.6. Feldbeschreibungen ResponseContext

## 6.6 Datenstruktur der Synchronize-Meldung

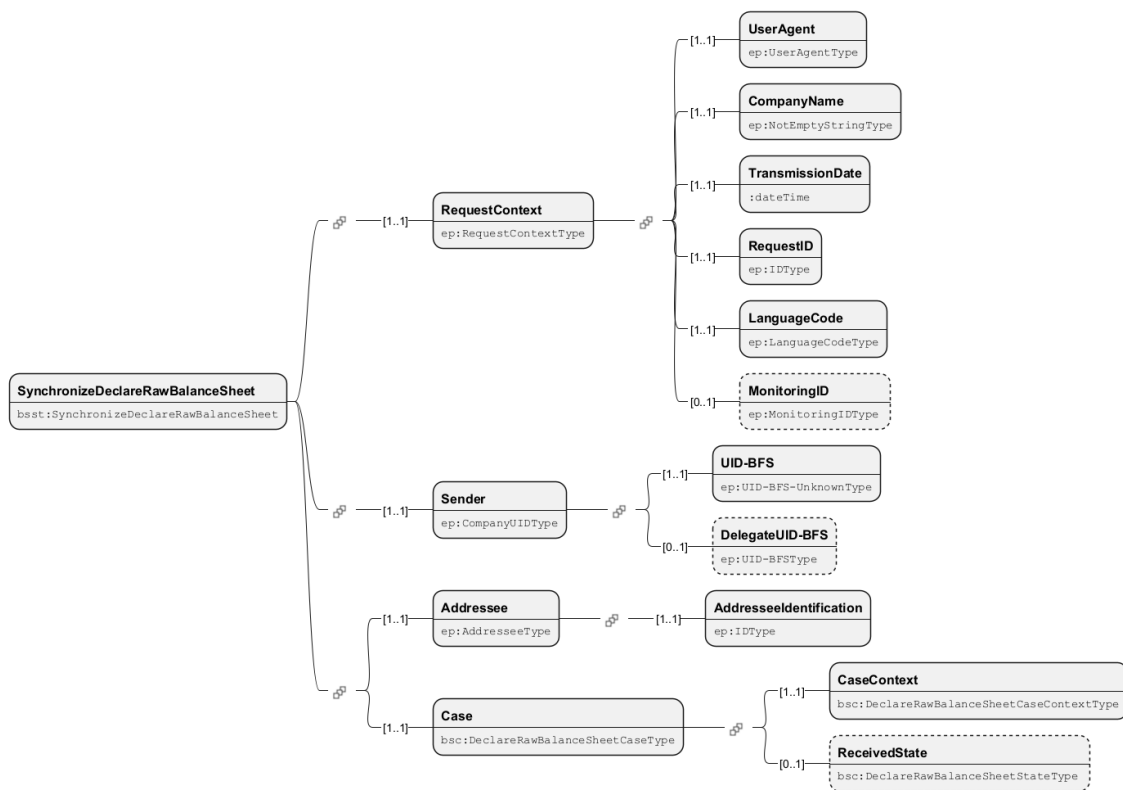


Abbildung 6.6. Schemabild SynchronizeDeclareRawBalanceSheet

Feldname	Beschreibung	Typ
RequestContext	Der RequestContext enthält Informationen zum Zeitpunkt der Übermittlung, dem Absender und der Sprache.	<a href="#">ep RequestContextType</a>
Sender	Absender der übermittelten Informationen	<a href="#">ep CompanyUIDType</a>
Addressee	Informationen zum Adressaten	<a href="#">ep AddresseeType</a>
Case	Informationen zum Fall	<a href="#">bsc DeclareRawBalanceSheetCaseType</a>

Tabelle 6.7. Feldbeschreibungen SynchronizeDeclareRawBalanceSheet

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<a href="#">ep UserAgentType</a>

Feldname	Beschreibung	Typ
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>
MonitoringID	Monitoring Identifikation	<a href="#">ep_MonitoringIDType</a>

Tabelle 6.8. Felddescriptions RequestContext

Feldname	Beschreibung	Typ
UID-BFS	Unternehmens-Identifikationsnummer / UID-BFS	<a href="#">ep_UID-BFS-UnknownType</a>
DelegateUID-BFS	Unternehmens UID-BFS Typ	<a href="#">ep_UID-BFSType</a>

Tabelle 6.9. Felddescriptions Sender

Feldname	Beschreibung	Typ
AddresseeIdentification	Identifikation des Adressaten	<a href="#">ep_IDType</a>

Tabelle 6.10. Felddescriptions Addressee

Feldname	Beschreibung	Typ
CaseContext	Dieses Element enthält Informationen und Identifikatoren zum Fall.	<a href="#">bsc_DeclareRawBalanceSheetCase-ContextType</a>
ReceivedState		<a href="#">bsc_DeclareRawBalanceSheetStateType</a>

Tabelle 6.11. Felddescriptions Case

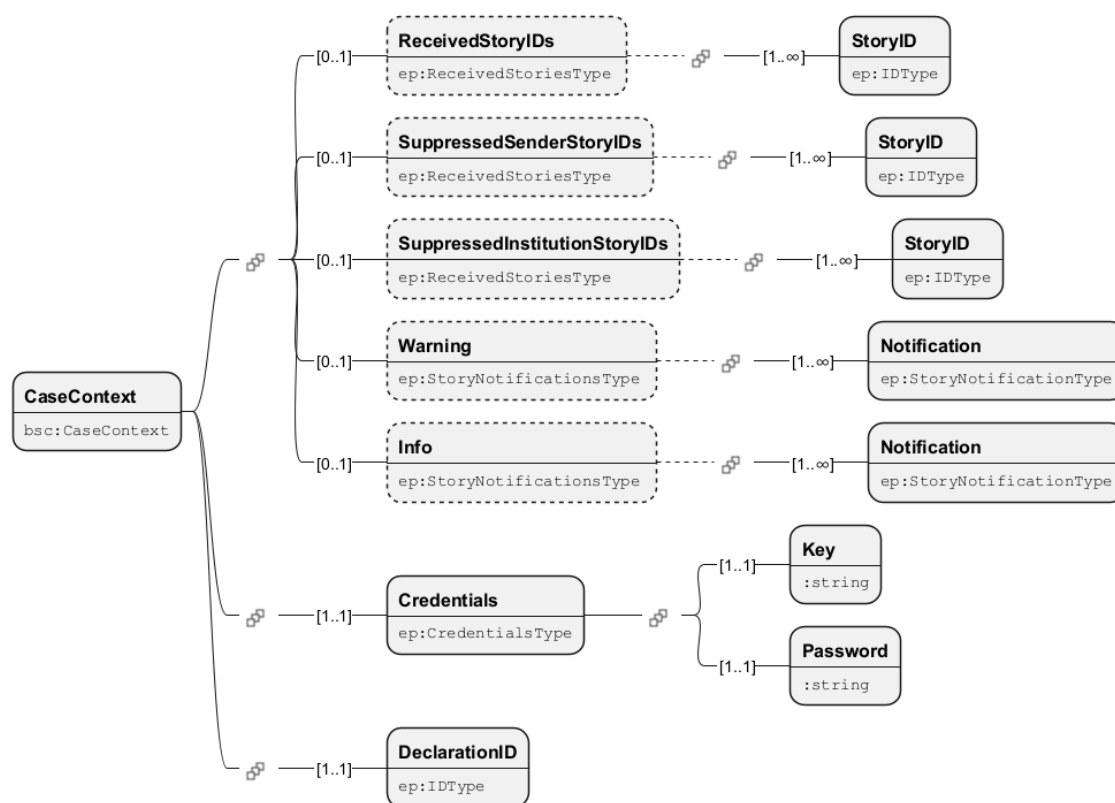


Abbildung 6.7. Schemabild CaseContext

Feldname	Beschreibung	Typ
ReceivedStoryIDs	Die Empfangenen StoryIDs als Quittungsliste.	<a href="#">ep ReceivedStories-Type</a>
SuppressedSenderStoryIDs	Die StoryIDs des Senders, die vom Distributor unterdrückt wurden. Das kann bei Mapping-Problemen geschehen.	<a href="#">ep ReceivedStories-Type</a>
SuppressedInstitutionStoryIDs	Die StoryIDs des Empfängers, die vom Distributor unterdrückt wurden. Das kann bei Mapping-Problemen geschehen.	<a href="#">ep ReceivedStories-Type</a>
Warning	Warnungen, die an den Producer / Transmitter zurückgeschickt werden.	<a href="#">ep StoryNotificationsType</a>
Info	Informationen, die an den Producer / Transmitter zurückgeschickt werden.	<a href="#">ep StoryNotificationsType</a>
Credentials	Berechtigungsnachweis	<a href="#">ep CredentialsType</a>
DeclarationID	Fachverfahrensspezifische Fallnummer des Distributors.	<a href="#">ep IDType</a>

Tabelle 6.12. Feldbeschreibungen CaseContext



## 6.7 Datenstruktur der Synchronize-Antwort

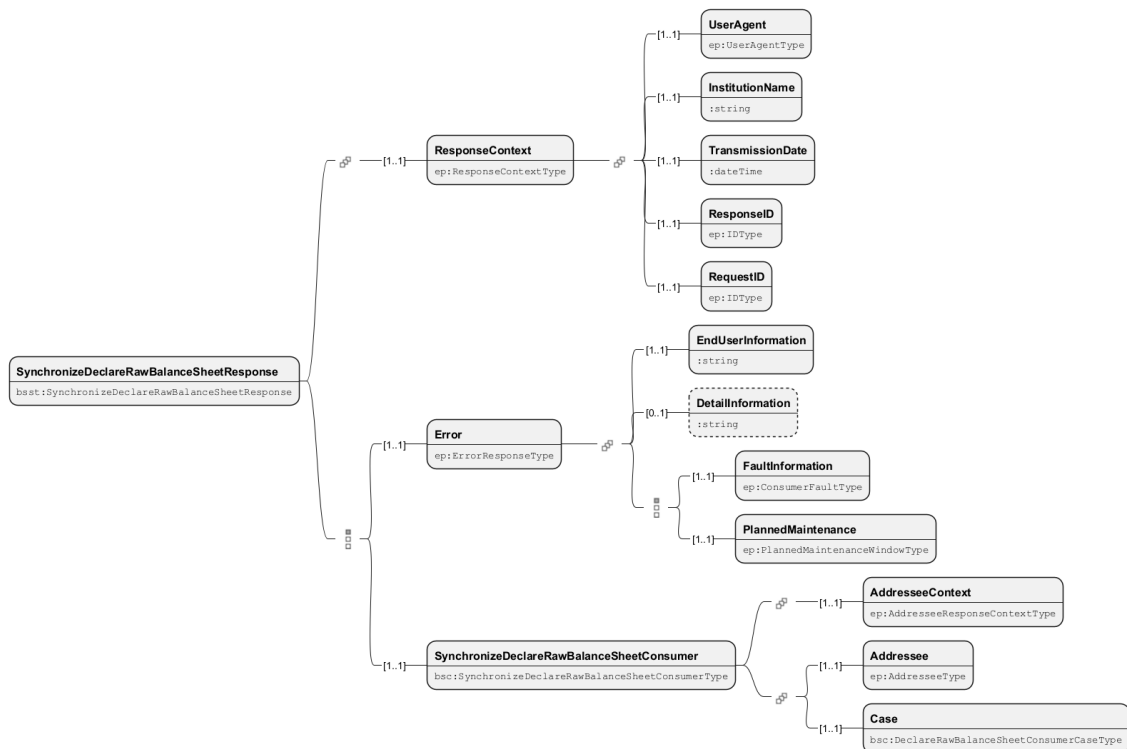


Abbildung 6.8. Schemabild SynchronizeDeclareRawBalanceSheetResponse

Feldname	Beschreibung	Typ
ResponseContext	Der ResponseContext enthält Informationen zum Zeitpunkt der Übermittlung, dem Absender und der Sprache.	<a href="#">ep_ResponseContextType</a>
Error	Die Verarbeitung war nicht erfolgreich und es sind Fehler aufgetreten.	<a href="#">ep_ErrorResponseType</a>
SynchronizeDeclareRawBalanceSheetConsumer		<a href="#">bsc_SynchronizeDeclareRawBalanceSheetConsumerType</a>

Tabelle 6.13. Felddescriptions SynchronizeDeclareRawBalanceSheetResponse

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<a href="#">ep_UserAgentType</a>
InstitutionName	Institutionsname	xs string
TransmissionDate	Übertragungszeitpunkt	xs dateTime
ResponseID	AntwortID	<a href="#">ep_IDType</a>

Feldname	Beschreibung	Typ
RequestID	AuftragsID	<a href="#">ep IDType</a>

Tabelle 6.14. Felddescriptions ResponseContext

Feldname	Beschreibung	Typ
EndUserInformation	Informationen zum Endnutzer	xs string
DetailInformation	Informationen im Detail	xs string
FaultInformation	Informationen zum Fehler	<a href="#">ep_ConsumerFault-Type</a>
PlannedMaintenance	Geplante Wartung	<a href="#">ep_PlannedMaintenanceWindowType</a>

Tabelle 6.15. Felddescriptions Error

Feldname	Beschreibung	Typ
AddresseeContext	Kontextangaben zum Adressaten	<a href="#">ep_AddresseeResponseContextType</a>
Addressee	Informationen zum Adressaten	<a href="#">ep AddresseeType</a>
Case	Informationen zum Fall	<a href="#">bsc_DeclareRawBalanceSheetConsumerCaseType</a>

Tabelle 6.16. Felddescriptions SynchronizeDeclareRawBalanceSheetConsumer

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<a href="#">ep UserAgentType</a>
InstitutionName	Institutionsname	xs string
TransmissionDate	Übertragungszeitpunkt	xs dateTime
ResponseID	AntwortID	<a href="#">ep IDType</a>
RequestID	AuftragsID	<a href="#">ep IDType</a>
ProducerResponse-Notifications	Antwort vom Producer zurückgegeben	<a href="#">ep_FeedbackNotificationsType</a>
Warning	Warnung wurde zurückgegeben	<a href="#">ep NotificationsType</a>
Info	Informationen wurden zurückgegeben	<a href="#">ep NotificationsType</a>

Tabelle 6.17. Felddescriptions AddresseeContext

Feldname	Beschreibung	Typ
AddresseeIdentification	Identifikation des Adressaten	<a href="#">ep IDType</a>

Tabelle 6.18. Felddesreibungen Addressee

Feldname	Beschreibung	Typ
CaseContext	Dieses Element enthält Informationen und Identifikatoren zum Fall.	<a href="#">bsc DeclareRawBalanceSheetCaseContextType</a>
State		<a href="#">bsc DeclareRawBalanceSheetStateType</a>
Completion	Informationen zur Freigabe der Daten.	<a href="#">ep CompletionType</a>

Tabelle 6.19. Felddesreibungen Case

## 7 DeclareBalanceSheet

### 7.1 Übermittlung des eCH-0276

Mit der vorliegende Operation kann ein eCH-0276 aus einer Deklarationslösung an die Haupt- und Nebensteuerdomizile übermittelt werden. Der Prozess wird in [Abschnitt 2.2, „Übermittlung des eCH-0276“](#) beschrieben.

#### 7.1.1 Dateien übermitteln (Attachments)

Allfällige Belege, Quittungen und weitere Dokumente können zusammen mit dem eCH-0276 übermittelt werden. Die zu übermittelnden Dateien sollen in einem ZIP-Archiv zusammengefasst und mit einem Passwort geschützt werden. Als Passwort kann die RequestId des SOAP Requests verwendet werden, da diese eindeutig ist und verschlüsselt in der SOAP Message übertragen wird.

Das ZIP wird dem HTTP POST des SOAP Requests als *multipart/form-data* angehängt. Dabei ist sicherzustellen, dass das ZIP mit einem eindeutigen Boundary-Parameter vom eigentlichen SOAP Request getrennt wird.

### 7.2 Initiale asynchrone Meldung

Die Aktivitätsdiagramme sind ohne den Endreceiver dargestellt. Die wesentliche Transmitterfunktionalität *soll* dargestellt werden. Die Anzeigen erfolgen wahlweise in Transmitter oder im ERP.

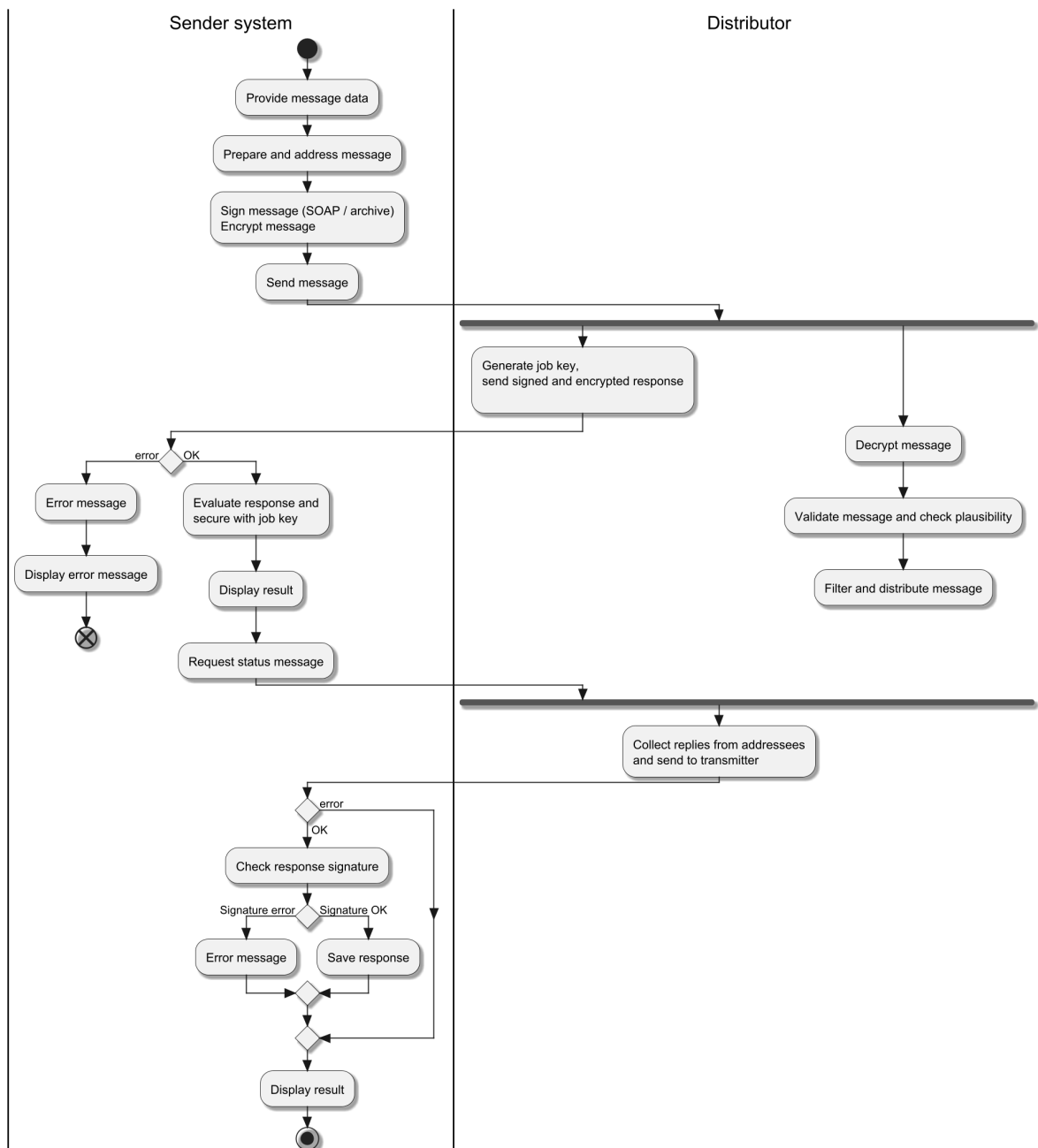


Abbildung 7.1. Asynchroner Meldungsprozess

### 7.3 Ablauf / Protokoll

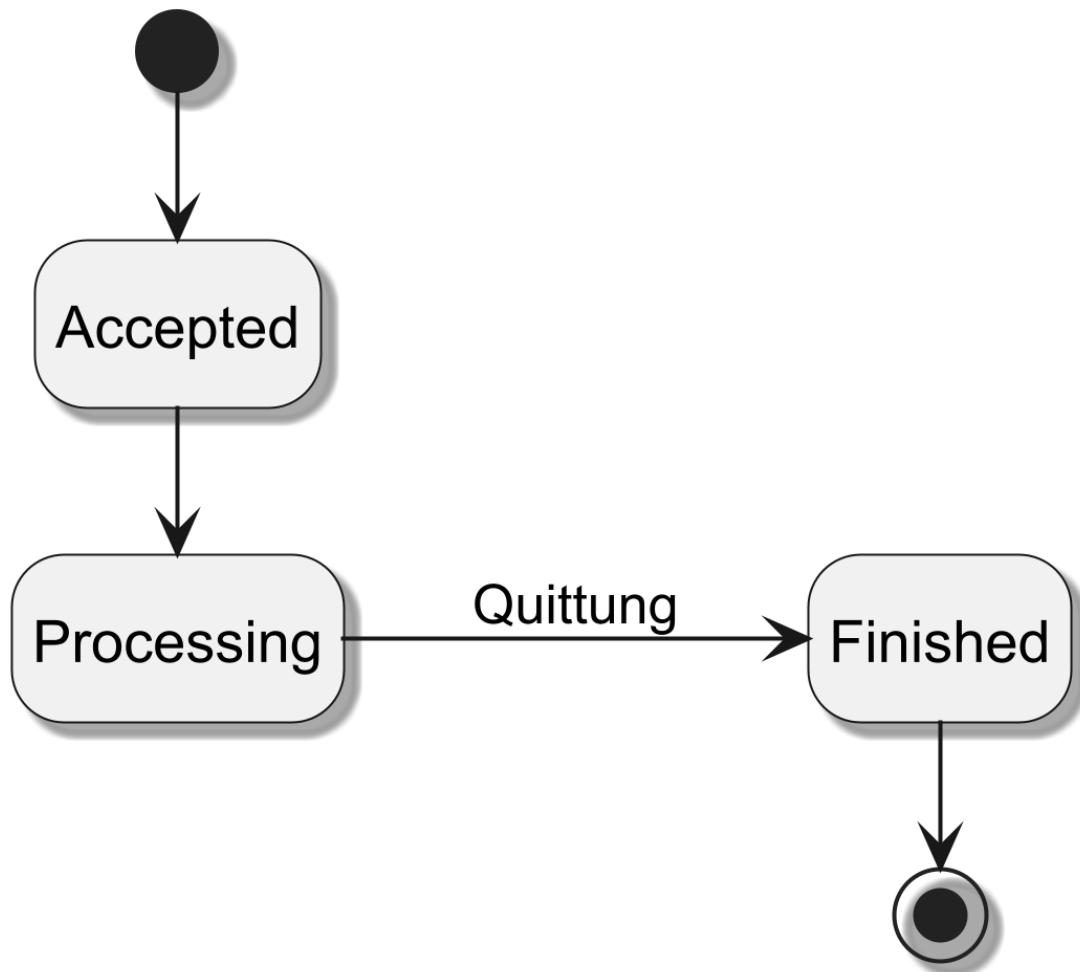


Abbildung 7.2. DeclareBalanceSheet Protokollstatus

## 7.4 Datenstruktur der initialen Meldung

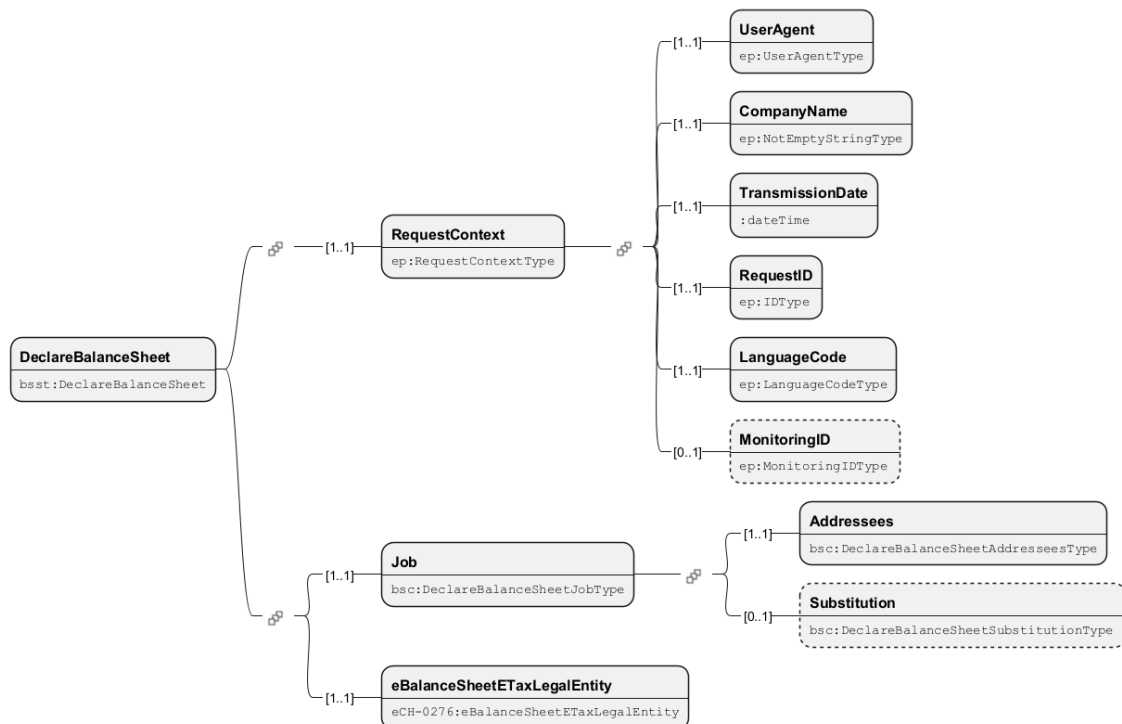


Abbildung 7.3. Schemabild DeclareBalanceSheet

Feldname	Beschreibung	Typ
RequestContext	Der RequestContext enthält Informationen zum Zeitpunkt der Übermittlung, dem Absender und der Sprache.	<a href="#">ep RequestContext-Type</a>
Job	Steuerungselement für die Verarbeitung der Meldung.	<a href="#">bsc DeclareBalanceSheetJobType</a>
eBalanceSheetETaxLegalEntity		eCH-0276 eBalanceSheetETaxLegalEntity

Tabelle 7.1. Felddescriptions DeclareBalanceSheet

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<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	Monitoring Identifikation	<a href="#">ep_MonitoringIDType</a>

Tabelle 7.2. Feldbeschreibungen RequestContext

Feldname	Beschreibung	Typ
Addressees	Angaben zum Adressaten	<a href="#">bsc_DeclareBalanceSheetAddresseeType</a>
Substitution	Entschädigung Job	<a href="#">bsc_DeclareBalanceSheetSubstitutionType</a>

Tabelle 7.3. Feldbeschreibungen Job

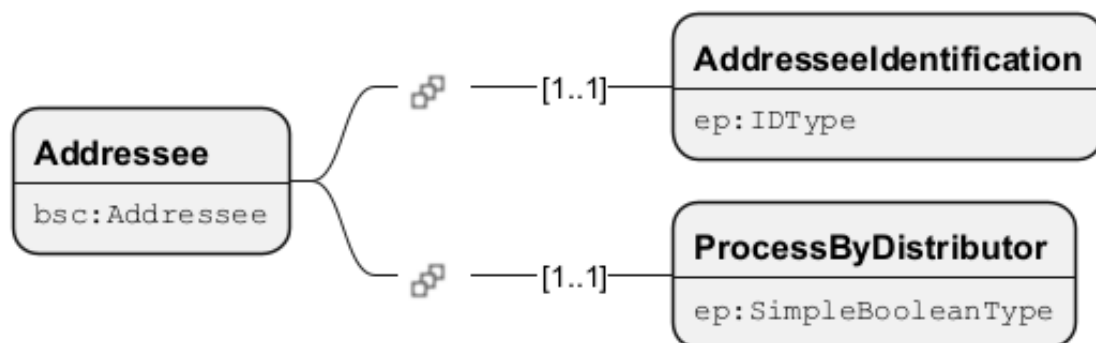


Abbildung 7.4. Schemabild Addressee

Feldname	Beschreibung	Typ
Addresseelidentification	Identifikation des Adressaten	<a href="#">ep_IDType</a>
ProcessByDistributor	Dieses Flag entscheidet, ob an den Distributor übermittelte Daten von ihm verarbeitet oder ignoriert werden sollen. Steht 'false', wird der Distributor keine Verteilung vornehmen und die gesendeten Inhalte für diese Institution ignorieren.	<a href="#">ep_SimpleBooleanType</a>
@addresseelID	Referenz auf Adressat	<a href="#">ep_InstanceRefIDType</a>

Tabelle 7.4. Feldbeschreibungen Addressee



## 7.5 Datenstruktur der initialen Antwort

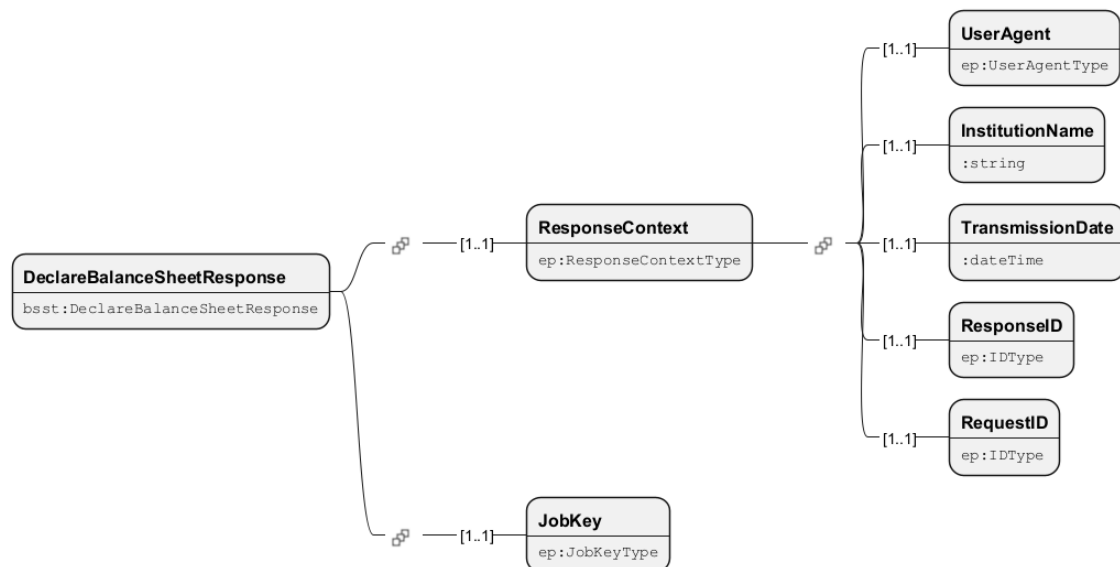


Abbildung 7.5. Schemabild DeclareBalanceSheetResponse

Feldname	Beschreibung	Typ
ResponseContext	Der ResponseContext enthält Informationen zum Zeitpunkt der Übermittlung, dem Absender und der Sprache.	<a href="#">ep ResponseContextType</a>
JobKey	Eindeutiger Bezeichner für den Job	<a href="#">ep JobKeyType</a>

Tabelle 7.5. Feldbeschreibungen DeclareBalanceSheetResponse

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<a href="#">ep UserAgentType</a>
InstitutionName	Institutionsname	xs string
TransmissionDate	Übertragungszeitpunkt	xs dateTime
ResponseID	AntwortID	<a href="#">ep IDType</a>
RequestID	AuftragsID	<a href="#">ep IDType</a>

Tabelle 7.6. Feldbeschreibungen ResponseContext

## 7.6 Datenstruktur der GetStatus-Meldung

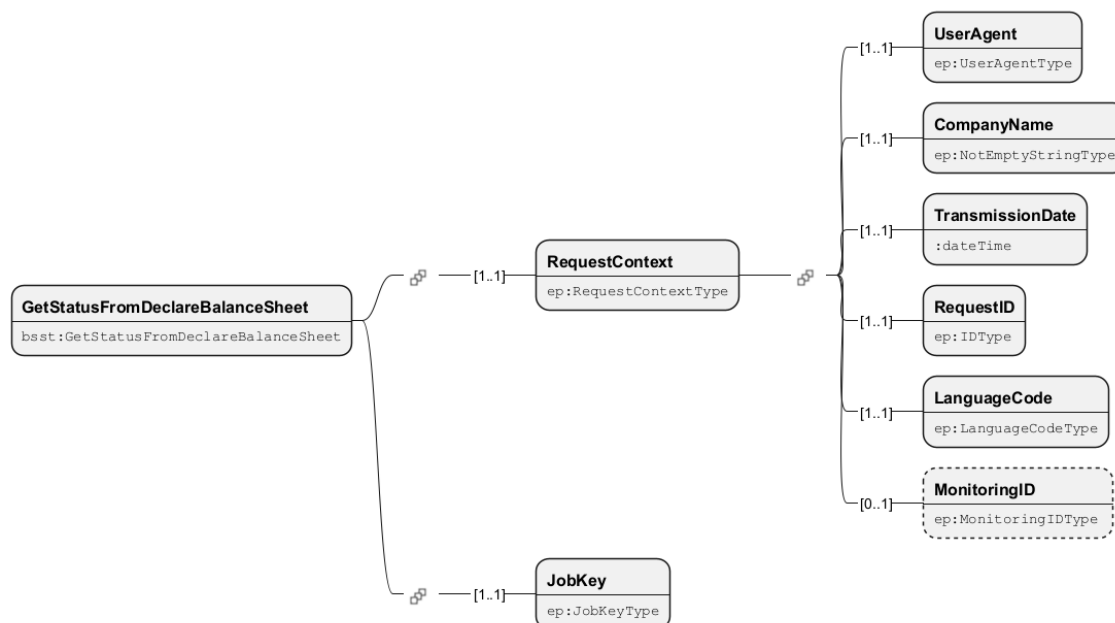


Abbildung 7.6. Schemabild GetStatusFromDeclareBalanceSheet

Feldname	Beschreibung	Typ
RequestContext	Der RequestContext enthält Informationen zum Zeitpunkt der Übermittlung, dem Absender und der Sprache.	<a href="#">ep RequestContextType</a>
JobKey	Eindeutiger Bezeichner für den Job	<a href="#">ep JobKeyType</a>

Tabelle 7.7. Felddescriptions GetStatusFromDeclareBalanceSheet

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<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	Monitoring Identifikation	<a href="#">ep MonitoringIDType</a>

Tabelle 7.8. Felddescriptions RequestContext

## 7.7 Datenstruktur der GetStatus-Antwort

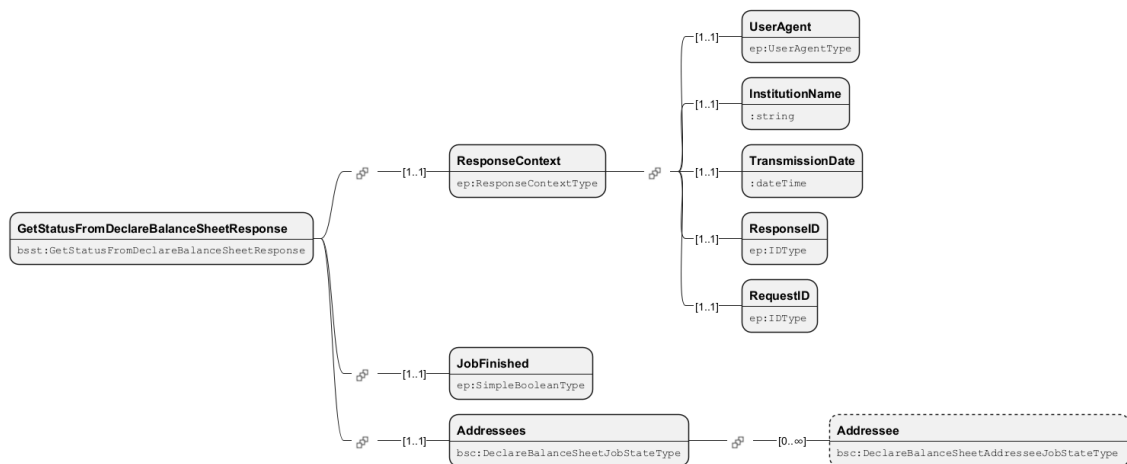


Abbildung 7.7. Schemabild GetStatusFromDeclareBalanceSheetResponse

Feldname	Beschreibung	Typ
ResponseContext	Der ResponseContext enthält Informationen zum Zeitpunkt der Übermittlung, dem Absender und der Sprache.	<a href="#">ep_ResponseContextType</a>
JobFinished	Boolean	<a href="#">ep_SimpleBooleanType</a>
Addressees	Initialer Status des Falls.	<a href="#">bsc_DeclareBalanceSheetJobStateType</a>

Tabelle 7.9. Feldbeschreibungen GetStatusFromDeclareBalanceSheetResponse

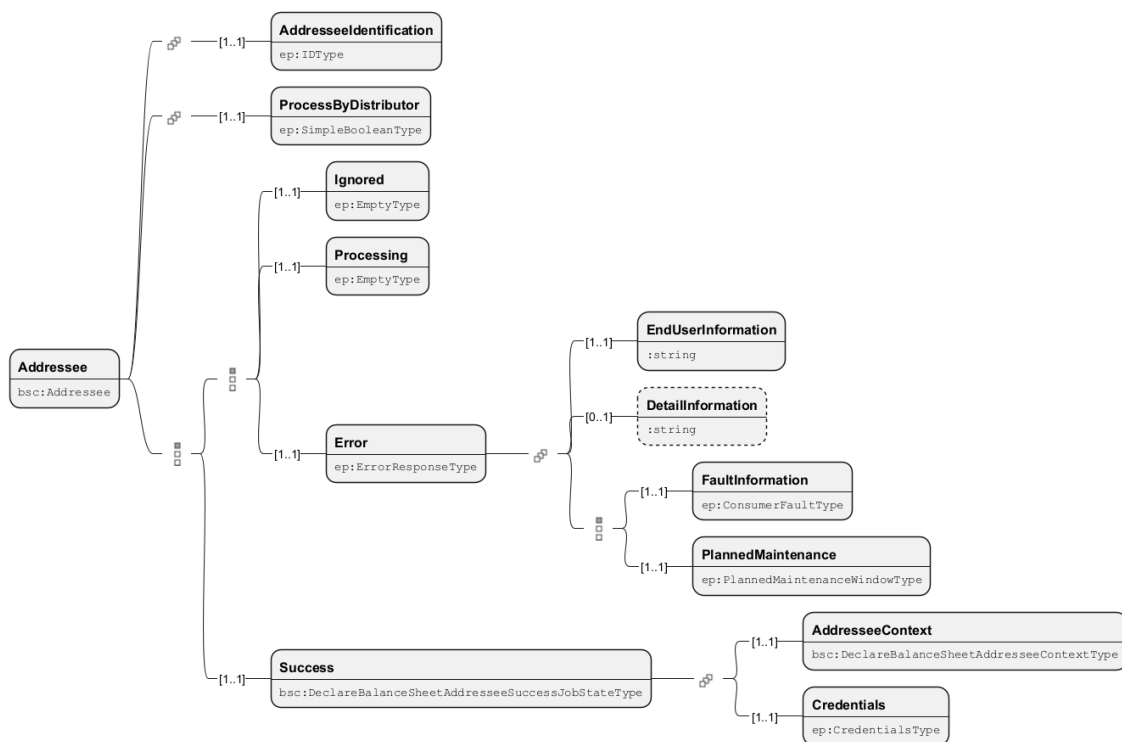


Abbildung 7.8. Schemabild Adressees

Feldname	Beschreibung	Typ
AdresseesIdentification	Identifikation des Adressaten	<a href="#">ep IDType</a>
ProcessByDistributor	Dieses Flag entscheidet, ob an den Distributor übermittelte Daten von ihm verarbeitet oder ignoriert werden sollen. Steht 'false', wird der Distributor keine Verteilung vornehmen und die gesendeten Inhalte für diese Institution ignorieren.	<a href="#">ep SimpleBooleanType</a>
@adresseesID	Referenz auf Adressat	<a href="#">ep InstanceRefIDType</a>
Ignored	Adresse ignoriert	<a href="#">ep EmptyType</a>
Processing	System ist noch am Verarbeiten	<a href="#">ep EmptyType</a>
Error	Verarbeitung war nicht erfolgreich und es gab Fehler.	<a href="#">ep ErrorResponse- Type</a>
Success	Initialer Status des Falls.	<a href="#">bsc DeclareBal- anceSheetAdres- seeSuccessJobSta- teType</a>

Tabelle 7.10. Felddescriptions Adressees

Feldname	Beschreibung	Typ
EndUserInformation	Informationen zum Endnutzer	xs string

Feldname	Beschreibung	Typ
DetailInformation	Informationen im Detail	xs string
FaultInformation	Informationen zum Fehler	<a href="#">ep_ConsumerFault-Type</a>
PlannedMaintenance	Geplante Wartung	<a href="#">ep_PlannedMaintenanceWindowType</a>

Tabelle 7.11. Feldbeschreibungen Error

Feldname	Beschreibung	Typ
AddresseeContext	Kontextangaben zum Adressaten	<a href="#">bsc_DeclareBalanceSheetAddresseeContextType</a>
Credentials	Berechtigungsnachweis	<a href="#">ep_CredentialsType</a>

Tabelle 7.12. Feldbeschreibungen Success

## 7.8 Datenstruktur der Synchronize-Meldung

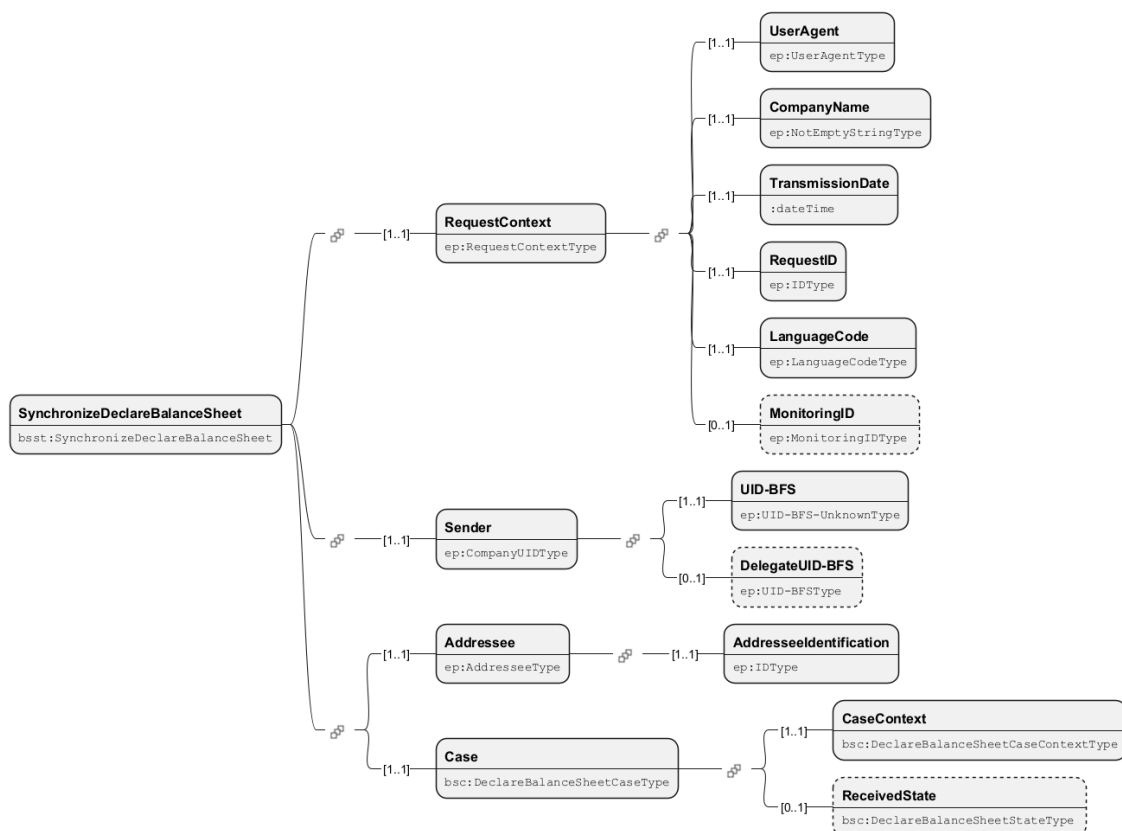


Abbildung 7.9. Schemabild SynchronizeDeclareBalanceSheet

Feldname	Beschreibung	Typ
RequestContext	Der RequestContext enthält Informationen zum Zeitpunkt der Übermittlung, dem Absender und der Sprache.	<a href="#">ep_RequestContext-Type</a>

Feldname	Beschreibung	Typ
Sender	Absender der übermittelten Informationen	<a href="#">ep_CompanyUIDType</a>
Addressee	Informationen zum Adressaten	<a href="#">ep_AddresseeType</a>
Case	Informationen zum Fall	<a href="#">bsc_DeclareBalanceSheetCaseType</a>

Tabelle 7.13. Felddescriptions SynchronizeDeclareBalanceSheet

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<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	Monitoring Identifikation	<a href="#">ep_MonitoringIDType</a>

Tabelle 7.14. Felddescriptions RequestContext

Feldname	Beschreibung	Typ
UID-BFS	Unternehmens-Identifikationsnummer / UID-BFS	<a href="#">ep_UID-BFS-UnknownType</a>
DelegateUID-BFS	Unternehmens UID-BFS Typ	<a href="#">ep_UID-BFSType</a>

Tabelle 7.15. Felddescriptions Sender

Feldname	Beschreibung	Typ
AddresseeIdentification	Identifikation des Adressaten	<a href="#">ep_IDType</a>

Tabelle 7.16. Felddescriptions Addressee

Feldname	Beschreibung	Typ
CaseContext	Dieses Element enthält Informationen und Identifikatoren zum Fall.	<a href="#">bsc_DeclareBalanceSheetCaseContextType</a>
ReceivedState		<a href="#">bsc_DeclareBalanceSheetStateType</a>

Tabelle 7.17. Felddescriptions Case

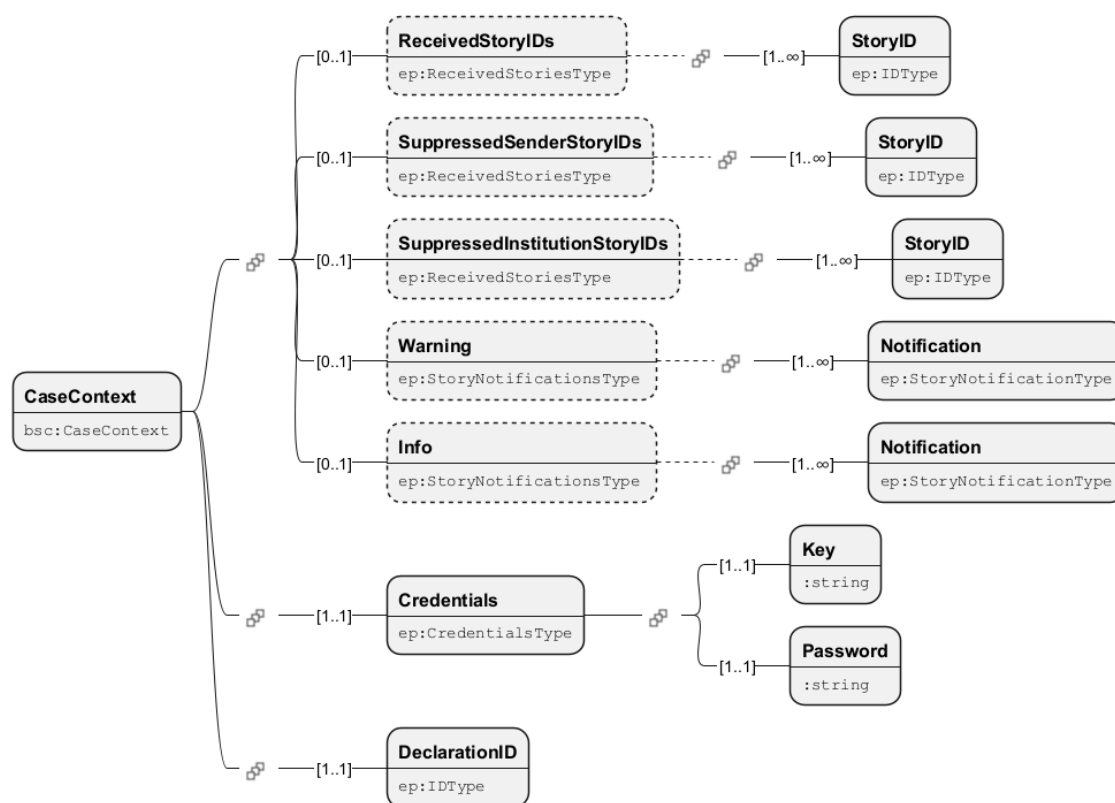


Abbildung 7.10. Schemabild CaseContext

Feldname	Beschreibung	Typ
ReceivedStoryIDs	Die Empfangenen StoryIDs als Quittungsliste.	<a href="#">ep ReceivedStoriesType</a>
SuppressedSenderStoryIDs	Die StoryIDs des Senders, die vom Distributor unterdrückt wurden. Das kann bei Mapping-Problemen geschehen.	<a href="#">ep ReceivedStoriesType</a>
SuppressedInstitutionStoryIDs	Die StoryIDs des Empfängers, die vom Distributor unterdrückt wurden. Das kann bei Mapping-Problemen geschehen.	<a href="#">ep ReceivedStoriesType</a>
Warning	Warnungen, die an den Producer / Transmitter zurückgeschickt werden.	<a href="#">ep StoryNotificationsType</a>
Info	Informationen, die an den Producer / Transmitter zurückgeschickt werden.	<a href="#">ep StoryNotificationsType</a>
Credentials	Berechtigungsnachweis	<a href="#">ep CredentialsType</a>
DeclarationID	Fachverfahrensspezifische Fallnummer des Distributors.	<a href="#">ep IDType</a>

Tabelle 7.18. Feldbeschreibungen CaseContext

## 7.9 Datenstruktur der Synchronize-Antwort

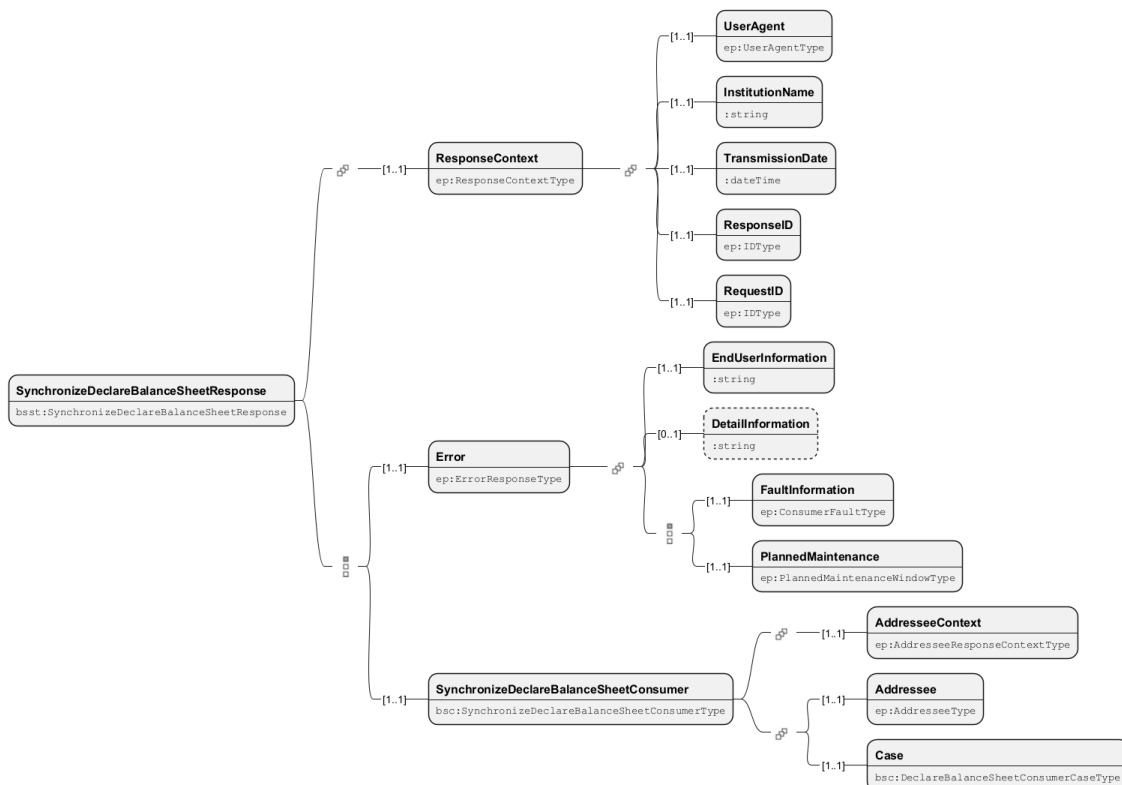


Abbildung 7.11. Schemabild SynchronizeDeclareBalanceSheetResponse

Feldname	Beschreibung	Typ
ResponseContext	Der ResponseContext enthält Informationen zum Zeitpunkt der Übermittlung, dem Absender und der Sprache.	<a href="#">ep ResponseContextType</a>
Error	Die Verarbeitung war nicht erfolgreich und es sind Fehler aufgetreten.	<a href="#">ep ErrorResponse-Type</a>
SynchronizeDeclareBalanceSheet-Consumer		<a href="#">bsc SynchronizeDeclareBalanceSheet-ConsumerType</a>

Tabelle 7.19. Felddescriptions SynchronizeDeclareBalanceSheetResponse

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<a href="#">ep UserAgentType</a>
InstitutionName	Institutionsname	xs string
TransmissionDate	Übertragungszeitpunkt	xs dateTime
ResponseID	AntwortID	<a href="#">ep IDType</a>



Feldname	Beschreibung	Typ
RequestID	AuftragsID	<a href="#">ep IDType</a>

Tabelle 7.20. Feldbeschreibungen ResponseContext

Feldname	Beschreibung	Typ
EndUserInformation	Informationen zum Endnutzer	xs string
DetailInformation	Informationen im Detail	xs string
FaultInformation	Informationen zum Fehler	<a href="#">ep_ConsumerFault-Type</a>
PlannedMaintenance	Geplante Wartung	<a href="#">ep_PlannedMaintenanceWindowType</a>

Tabelle 7.21. Feldbeschreibungen Error

Feldname	Beschreibung	Typ
AddresseeContext	Kontextangaben zum Adressaten	<a href="#">ep_AddresseeResponseContextType</a>
Addressee	Informationen zum Adressaten	<a href="#">ep AddresseeType</a>
Case	Informationen zum Fall	<a href="#">bsc_DeclareBalanceSheetConsumer-CaseType</a>

Tabelle 7.22. Feldbeschreibungen SynchronizeDeclareBalanceSheetConsumer

Feldname	Beschreibung	Typ
UserAgent	Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert	<a href="#">ep UserAgentType</a>
InstitutionName	Institutionsname	xs string
TransmissionDate	Übertragungszeitpunkt	xs dateTime
ResponseID	AntwortID	<a href="#">ep IDType</a>
RequestID	AuftragsID	<a href="#">ep IDType</a>
ProducerResponse-Notifications	Antwort vom Producer zurückgegeben	<a href="#">ep_FeedbackNotificationsType</a>
Warning	Warnung wurde zurückgegeben	<a href="#">ep NotificationsType</a>
Info	Informationen wurden zurückgegeben	<a href="#">ep NotificationsType</a>

Tabelle 7.23. Feldbeschreibungen AddresseeContext

Feldname	Beschreibung	Typ
AddresseIdentifica- tion	Identifikation des Adressaten	<a href="#">ep IDType</a>

Tabelle 7.24. Felddesreibungen Addressee

Feldname	Beschreibung	Typ
CaseContext	Dieses Element enthält Informationen und Identifikatoren zum Fall.	<a href="#">bsc DeclareBalan- ceSheetCaseCon- textType</a>
State		<a href="#">bsc DeclareBalan- ceSheetStateType</a>

Tabelle 7.25. Felddesreibungen Case

## 8 Use Cases

### 8.1 UC001 Initiale Meldung senden

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

Kurzbeschreibung	Eine elektronische Meldung <b>muss</b> an einen oder mehrere Endreceiver versendet werden. Die Rückantwort der Endreceiver wird ausgewertet und abgelegt. Ein Archiv-File der gesendeten Meldung wird ebenfalls gesichert.
Akteure	Sendersystem, Distributor, Endreceiver
Auslöser	Das Sendersystem detektiert eine Situation, die per Swissdec gemeldet werden muss/kann.
Vorbedingungen	Das Sendersystem ist in der Lage, elektronische Swissdec-Meldungen zu versenden und die Antwort zu verarbeiten.
Nachbedingungen	Die Meldung wurde vom Endreceiver empfangen und durch eine Rückantwort quittiert.  Bei einem Fehlschlag:  • Fehlermeldung
Included Use Cases	<a href="#">Abschnitt 8.2, „UC002 Status abholen“</a> , <a href="#">Abschnitt 8.15, „UC015 Security anwenden“</a>
Standardablauf	<ol style="list-style-type: none"> <li>1. Das Sendersystem übergibt dem Transmitter die Daten mit den Empfängeradressen.</li> <li>2. Der Transmitter bereitet die Meldung als SOAP-Request mit zugehöriger Adressierung (Job) auf.</li> <li>3. Die Meldung wird nach Spezifikation signiert und verschlüsselt.</li> <li>4. Der Transmitter sendet die aufbereitete und signierte Meldung über HTTPS an den Distributor. Dieser Schritt hat 2 mögliche Ausprägungen: <ol style="list-style-type: none"> <li>a. Asynchroner Aufruf, bei dem der Distributor mit einem JobKey antwortet und der Transmitter mittels &lt;GetStatus&gt; Polling die Antworten der Endreceiver erhält.</li> <li>b. Synchroner Aufruf, ein blockierender Aufruf, auf dem die Antworten der Endreceiver direkt als Antwort auf die Meldung zurückgegeben werden. Beim synchronen Aufruf sind die Informationen, welche beim asynchronen Aufruf mittels GetStatus abgeholt werden bereits in der Response vorhanden.</li> </ol> <p>Welche Methode bei welcher Operation benötigt wird, ist in den operationsspezifischen Kapiteln definiert.</p> </li> <li>5. Der Distributor prüft die Meldung auf Validität und Plausibilität.</li> <li>6. Der Distributor bereitet eine oder mehrere Meldungen für die gewählten Endreceiver auf und sendet diese an den/die Endreceiver.</li> <li>7. Der Endreceiver prüft die Meldung und startet die Verarbeitung des Jobs.</li> <li>8. Der Endreceiver meldet seine Antwort dem Distributor, der sie dem Sendersystem zukommen lässt (synchron oder asynchron).</li> <li>9. Der Transmitter wartet die Rückantwort des Distributors ab.</li> <li>10. Das Ergebnis der Übermittlung wird auf Transmitterseite aufbereitet und angezeigt.</li> </ol>
Alternative Abläufe	<a href="#">Abschnitt 8.3, „UC003 Testmeldung kennzeichnen“</a>  {nach Schritt 1}

	<p>1. b) Die Meldung wird als Testmeldung gekennzeichnet. (Ein Element Test-Case wird im Job Element eingefügt).</p> <p>{weiter mit Schritt 2}</p> <p>ACHTUNG: Wird ein Fall als Testfall gemeldet, <b>müssen</b> auch alle weiteren Synchronisierungen (im CaseContext) als Testfall markiert sein ( <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a> ).</p> <p>{nach Schritt 1}</p> <p>2. c) Die Meldung wird als Ersatzmeldung gekennzeichnet. (Ein Element Substitution mit dem Identifikator der vorangegangenen Meldung wird in die neue Meldung eingefügt <a href="#">[[CONTAINERXSD]]</a> ).</p> <p>{weiter mit Schritt 2}</p>
Fehlerliste	<p>Fachliche Fehler:</p> <ul style="list-style-type: none"> <li>• Die Meldung verstösst gegen die Plausibilisierungsregeln</li> </ul> <p>Technische Fehler:</p> <ul style="list-style-type: none"> <li>• Fehler beim Signieren oder Verschlüsseln. In diesem Fall <i>sollte</i> die Übermittlung zu einem späteren Zeitpunkt wiederholt werden, oder, falls das Problem langfristig besteht, Rückfrage beim Hersteller genommen werden.</li> <li>• Der Endreceiver befindet sich in einem geplanten Wartungsfenster (nur bei synchroner Übermittlung möglich). Der Benutzer <i>sollte</i> darauf hingewiesen werden, dass nach Ende des Wartungsfensters erneut übermittelt kann.</li> <li>• Der Endreceiver ist nicht erreichbar (nur bei synchroner Übermittlung möglich). Die Übermittlung kann zu einem späteren Zeitpunkt erneut versucht werden.</li> <li>• Der Endreceiver ist nicht adressierbar (nur bei synchroner Übermittlung möglich). Überprüfung der Adressierung seitens Sendersystem/Transmitter.</li> <li>• Die vom Sendersystem aufbereitete Meldung entspricht nicht dem Schema (Validität nicht gegeben)</li> <li>• JobKey ungültig. Dies kann sowohl ein fehlerhafter JobKey sein, als auch ein Ablauf der Gültigkeit.</li> </ul>

Tabelle 8.1. UC001 Initiale Meldung senden

## 8.1.1 Spezielle Anforderungen

### 8.1.1.1 RequestContext

Jede Swissdec Operation enthält das Element RequestContext, welches korrekt abgefüllt werden muss. Der RequestContext dient zur Identifikation und Nachvollziehbarkeit von Übermittlungen.

UserAgent	<p><b>Wichtig!</b> Der userAgent muss im Anschluss an die abgeschlossene Swissdec-Zertifizierung aktualisiert werden und muss sämtliche Informationen zum ERP-System (und nicht zum Endbenutzer) enthalten.</p>	
	Producer	Name des Softwareherstellers
	Name	Name des Produktes
	Version	Version der Transmittersoftware

	StandardVersion	Version des Swissdec Standards (inkl. Minor-Version), z.b. 6.0
	Certificate	Zertifikat ID aus der Swissdec Zertifizierung.
Tabelle 8.3. UserAgentType		
CompanyName	Name der übermittelten Firma	
TransmissionDate	Zeitstempel der Übermittlung	
RequestID	Eindeutige RequestID der Übermittlung (technisch).	
LanguageCode	Sprache der übermittelnden Firma oder verantwortlichen Person (wird für Fehlerausgaben verwendet)	
MonitoringID	Diese ID wird verwendet, um auf den Swissdec Testsystemen den Softwarehersteller zu identifizieren. Die MonitoringID ist für die Verwendung der Testsysteme zwingend, wird aber im produktiven Umfeld nicht gebraucht. <i>(Sollte auf produktiven Systemen leer sein)</i>	

Tabelle 8.2. RequestContextType

### 8.1.1.2 Auswahl der Empfänger und Filterung

Im Element <Job> werden die Endempfänger adressiert, die bei dieser Übermittlung Daten erhalten sollen (Addressees). Für jeden Addressee gibt es das Element <ProcessByDistributor> welches als Boolean auf true oder false gesetzt werden kann. Ist der Wert 'false', wird diese Institution unabhängig von den übermittelten Daten vom Distributor ignoriert und nicht verarbeitet.

Das Element <ProcessByDistributor> *kann* somit genutzt werden, um die Verarbeitung einer Institution momentan auszusetzen. Dies kann zum Beispiel beim Erstellen einer Ersatzmeldung von Nutzen sein. Es wird empfohlen, dass bei der Übermittlung grösserer Datenmengen (>2000 Personen) die Daten bereits vorher gefiltert werden und die Filterung nicht dem Distributor überlassen wird.

### 8.1.1.3 Archiv-Files erstellen

Mit dieser Anforderung *wird* sichergestellt, dass eine Kopie jeder gesendeten und empfangenen Meldung gesichert wird. Die Daten **müssen** zu einem SOAP-Request aufbereitet und als XML-Instanzdokument abgelegt werden. Archivdateien **müssen** signiert sein, dürfen aber nicht verschlüsselt sein. Welche Dateien langfristig archiviert werden und wie lange sie zur Verfügung stehen müssen, ist abhängig von der Domäne/Institution und wird von der Fachseite vorgegeben.

## 8.2 UC002 Status abholen

Kurzbeschreibung	<p>Siehe <a href="#">Abschnitt 8.1, „UC001 Initiale Meldung senden“</a> . Für jede asynchrone Operation existiert eine dazugehörige &lt;GetStatusFrom*&gt; Operation, um eine Statusmeldung für die übermittelten Daten abzuholen. Diese zeigt an, ob die adressierten Endreceiver die übermittelten Daten aus technischer Sicht akzeptiert haben.</p> <p>Der JobKey in der Antwort der <a href="#">Abschnitt 8.1, „UC001 Initiale Meldung senden“</a> wird dazu verwendet, die zur Meldung gehörige Statusmeldung abzuholen.</p>
Akteure	Sendersystem, Distributor
Auslöser	Es wurde eine Lohnmeldung übermittelt und vom Distributor eine Bestätigung mit JobKey empfangen.

Vorbedingungen	Das Sendersystem verfügt über einen JobKey, der es ermöglicht, den Status einer laufenden Übermittlung abzufragen.
Nachbedingungen	Der aktuelle Status der Übermittlung wurde beim Distributor abgeholt.  Bei einem Fehlschlag: <ul style="list-style-type: none"><li>• Fehlermeldung</li></ul>
Included Use Cases	<a href="#">Abschnitt 8.15, „UC015 Security anwenden“</a>
Standardablauf	<ol style="list-style-type: none"> <li>1. Der Transmitter bereitet eine Statusabfrage, die den JobKey des aktuellen Falls enthält, auf.</li> <li>2. Die Meldung wird nach Spezifikation signiert und verschlüsselt.</li> <li>3. Der Transmitter sendet die aufbereitete und signierte Meldung über HTTPS an den Distributor.</li> <li>4. Der Distributor prüft die Meldung auf Validität und Plausibilität.</li> <li>5. Der Distributor überprüft, ob der JobKey bekannt ist und antwortet dem Transmitter mit der dem JobKey entsprechenden Statusmeldung.</li> <li>6. Der Transmitter wartet die Rückantwort des Distributors ab.</li> <li>7. Das Ergebnis der Job-Verarbeitung wird auf Transmitterseite aufbereitet und angezeigt.</li> </ol>
Alternative Abläufe	keine
Fehlerliste	<p>Technische Fehler:</p> <ul style="list-style-type: none"> <li>• Fehler beim Signieren oder Verschlüsseln. In diesem Fall <i>sollte</i> die Übermittlung zu einem späteren Zeitpunkt wiederholt werden, oder, falls das Problem langfristig besteht, Rückfrage beim Hersteller genommen werden.</li> <li>• JobKey ungültig. Dies kann sowohl ein fehlerhafter JobKey sein, als auch ein Ablauf der Gültigkeit.</li> <li>• Der Endrezeiver befindet sich in einem geplanten Wartungsfenster. Der Benutzer <i>sollte</i> darauf hingewiesen werden, dass nach Ende des Wartungsfensters erneut übermittelt kann.</li> <li>• Der Endrezeiver ist nicht erreichbar. Die Übermittlung kann zu einem späteren Zeitpunkt erneut versucht werden.</li> <li>• Der Endrezeiver ist nicht adressierbar. Überprüfung der Adressierung seitens Sendersystem/Transmitter.</li> </ul>

Tabelle 8.4. UC002 Status abholen

## 8.2.1 Spezielle Anforderungen

### 8.2.1.1 Polling

Die Statusabfrage **muss** mehrmals durchgeführt werden können. Sobald das Element <JobFinished> auf ‚true‘ steht, ist die Übermittlung abgeschlossen. Es darf keine weitere Statusabfrage durchgeführt werden können. Die Abfrage des Status kann automatisiert werden. Der Abstand zwischen den einzelnen automatisierten Abfragen **muss** jedoch mindestens zehn Sekunden betragen. Eine erste Statusabfrage *kann* direkt im Anschluss an die Initialmeldung durchgeführt werden. Weitere Statusabfragen **müssen** in Abständen erfolgen, die abhängig von der Grösse der gesendeten Initialmeldung sind. Nach 2-3 Pollingversuchen *kann* der automatische Prozess gestoppt werden. Er **muss** in diesem Fall aber manuell wieder angestossen werden können.

### 8.2.1.2 Identifikation des Prozess (nachfolgend DomainID)

Jeder Swissdec Prozess enthält eine eindeutige Identifikation. Diese ID wird in der Response der GetStatus Operation zurückgegeben.

Diese DomainID dient während der ganzen Phase des Prozesses als Identifizierung und muss deutlich erkennbar angezeigt werden.

### 8.2.1.3 Credentials

In der Antwort sind für Prozesse ohne SUA Signatur die Credentials für das Weiterverarbeiten des Prozessablaufs zu finden.

Ist für eine Operation eine SUA Signatur zwingend, kann auf die Credentials verzichtet werden.

### 8.2.1.4 ProducerResponseNotifications (Plausibilisierung)

Der Distributor prüft den fachlichen Inhalt auf Plausibilität. Falls das Regelwerk Informationen oder Warnungen findet, werden diese im Element ProducerResponseNotifications zurückgegeben. Diese Notifications müssen für den Endkunden klar ersichtlich sein, so das allfällige fachliche Unklarheiten bereinigt werden können.

### 8.2.1.5 Plausibilisierungsfehler

Wenn die Daten fehlerhafte Daten enthalten (fachlich) detektiert werden, so werden diese im Error Element zurückgegeben.

### 8.2.1.6 Wartungsfenster

Befindet sich der Empfänger in einem Wartungsfenster, so wird das Zeitfenster und der Grund der Wartungsarbeiten zurückgegeben. Erst wenn die Zeit des Wartungsfenster vorbei ist, kann eine erneute Übermittlung versucht werden. Möglicherweise gleichzeitig adressierte Endempfänger, die kein Wartungsfenster melden, können normal weiterverarbeitet werden.

## 8.3 UC003 Testmeldung kennzeichnen

Kurzbeschreibung	<p>Es <b>muss</b> möglich sein, die Meldung als Testcase zu kennzeichnen, um Tests bei der Installation oder bei Problemen in der Produktion zu ermöglichen. Der Ablauf einer Testmeldung, sowie die Beschreibung, wie Testdaten gekennzeichnet werden <b>müssen</b>, sind im <a href="#">Abschnitt 8.1, „UC001 Initiale Meldung senden“</a> unter alternative Abläufe zu finden.</p> <p>Das TestCase Flag ist jeweils auf dem Job bei der initialen Meldung oder im CaseContext bei der Synchronisation gesetzt.</p> <p>Die technischen Operationen, wie GetStatus oder ein leeres Synchronize können kein TestCase Flag enthalten.</p>
Akteure	Sendersystem, Distributor, Endreceiver
Auslöser	Das Sendersystem will eine Testmeldung übermitteln (Supportfall, Test, etc.).
Vorbedingungen	Das Sendersystem ist in der Lage, elektronische Swissdec-Meldungen zu versenden und die Antwort zu verarbeiten.
Nachbedingungen	<p>Die Meldung wurde vom Endreceiver empfangen. Sie <b>darf nicht</b> produktiv weiterverarbeitet werden.</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 übergibt dem Transmitter Daten, welche als Testfall markiert sind.</li> <li>2. Der Transmitter bereitet die Meldung als SOAP-Request mit zugehöriger Adressierung (Job) auf.</li> <li>3. Die Meldung wird nach Spezifikation signiert und verschlüsselt.</li> <li>4. Der Transmitter sendet die aufbereitete und signierte Meldung über HTTPS an den Distributor.</li> </ol>

	<p>5. Der Distributor prüft die Meldung auf Validität und Plausibilität.</p> <p>6. Der Distributor bereitet eine oder mehrere Meldungen für die gewählten Endreceiver auf und sendet diese an den/die Endreceiver.</p> <p>7. Der Endreceiver prüft die Meldung, erkennt sie als Testfall und stoppt die Verarbeitung.</p>
Alternative Abläufe	keine
Fehlerliste	Analog zu <a href="#">Abschnitt 8.1, „UC001 Initiale Meldung senden“</a> , <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a>

Tabelle 8.5. UC003 Testmeldung kennzeichnen

## 8.4 UC004 Ersatzmeldung kennzeichnen

Kurzbeschreibung	<p>Falls die Meldung unvollständig oder (fachlich) fehlerhaft übermittelt und bereits freigegeben wurde, <i>muss</i> es möglich sein, eine neue Übermittlung zu machen und diese als Ersatzmeldung zu kennzeichnen. Der Ablauf wie die Meldung als Ersatzmeldung gekennzeichnet werden muss, ist im <a href="#">Abschnitt 8.1, „UC001 Initiale Meldung senden“</a> unter alternative Abläufe zu finden.</p> <p>Die Details zur Ersatzmeldung ist in den operationsspezifischen Kapiteln definiert.</p>
Akteure	Sendersystem, Distributor, Endreceiver
Auslöser	Das Sendersystem will eine Ersatzmeldung übermitteln.
Vorbedingungen	Das Sendersystem ist in der Lage, elektronische Swissdec-Meldungen zu versenden und die Antwort zu verarbeiten.
Nachbedingungen	<p>Die Meldung wurde vom Endreceiver empfangen und wird vom Empfänger-system verarbeitet.</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 übergibt dem Transmitter Daten, welche als Ersatzmeldung markiert sind. Die zu ersetzende Meldung wird anhand ihrer DeclarationID ausgewählt.</li> <li>2. Der Transmitter bereitet die Meldung als SOAP-Request mit zugehöriger Adressierung (Job) auf.</li> <li>3. Die Meldung wird nach Spezifikation signiert und verschlüsselt.</li> <li>4. Der Transmitter sendet die aufbereitete und signierte Meldung über HTTPS an den Distributor.</li> <li>5. Der Distributor prüft die Meldung auf Validität und Plausibilität.</li> <li>6. Der Distributor bereitet eine oder mehrere Meldungen für die gewählten Endreceiver auf und sendet diese an den/die Endreceiver. Sollte ein Endempfänger diese Meldung als Duplikat erhalten, wird dies vom Distributor entsprechend gekennzeichnet.</li> <li>7. Der Endreceiver prüft die Meldung, erkennt sie als Ersatzmeldung und handelt entsprechend.</li> </ol>
Alternative Abläufe	keine
Fehlerliste	Analog zu <a href="#">Abschnitt 8.1, „UC001 Initiale Meldung senden“</a> , <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a>

Tabelle 8.6. UC003 Testmeldung kennzeichnen



## 8.5 UC005 Synchronisieren

Nach jeder Meldung ist mit dem Aufruf <Synchronize> der Fall zu synchronisieren. 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 melden und quittieren“](#)
- [Abschnitt 8.9, „UC009 Stories abholen und verarbeiten“](#)

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 senden“</a> und eventuell <a href="#">Abschnitt 8.2, „UC002 Status abholen“</a> wurde erfolgreich durchgeführt</li> <li>• Alle Identifikatoren für den Fall sind bekannt</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> <a href="#">Abschnitt 8.8, „UC008 Stories melden und quittieren“</a> <a href="#">Abschnitt 8.9, „UC009 Stories abholen und verarbeiten“</a> <a href="#">Abschnitt 8.15, „UC015 Security anwenden“</a>
Standardablauf	<ol style="list-style-type: none"> <li>1. Der Akteur wählt die Stories aus, die er übermitteln will und sendet diese dem Endreceiver. Zusätzlich quittiert er den aktuellen State (siehe <a href="#">Abschnitt 8.5.1.2, „Quittierung“</a>)</li> <li>2. Dieser wertet die Stories aus und meldet zurück, was bei ihm noch an offenen Stories vorhanden ist und/oder was der Akteur von ihm verlangt hat.</li> <li>3. Der Akteur analysiert die erhaltene Antwort, überprüft die empfangenen Stories und pflegt alle relevanten Daten ins Sendersystem ein.</li> </ol>
Alternative Abläufe	<ol style="list-style-type: none"> <li>1. Der Akteur wählt keinen Fall und keine Stories aus, die er übermitteln will und sendet ein leeres Synchronize.</li> <li>2. Der Endreceiver analysiert alle offenen Fälle dieses Vertrages und sammelt die DomainID's im Available Element.</li> <li>3. Der Akteur analysiert die erhaltene Antwort, und synchronisiert die Fälle aus dem Available mittels Standardablauf.</li> </ol>
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> <li>• Eine Story wurde nicht quittiert. Die Story muss erneut gesendet werden.</li> </ul> <p>Technische Fehler:</p> <ul style="list-style-type: none"> <li>• Fehler beim Signieren oder Verschlüsseln</li> </ul>

	<ul style="list-style-type: none"> <li>• Der Endreceiver befindet sich in einem geplanten Wartungsfenster</li> <li>• Der Endreceiver ist nicht erreichbar</li> <li>• Der Endreceiver ist nicht adressierbar</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 8.7. UC005 Synchronisieren

## 8.5.1 Spezielle Anforderungen

### 8.5.1.1 Unique StoryID

Die StoryID's müssen Fallübergreifend eindeutig sein.

### 8.5.1.2 Quittierung

Bei jeder Synchronisierung **muss** der bei der letzten Synchronisierung erhaltene State quittiert werden. Dies geschieht dadurch, dass der in der vorhergehenden Response enthaltene State im Feld <ReceivedState> zurückgegeben wird. Ausnahme: Beim Abschluss des Prozesses (State=Finished) *kann* diese Quittierung des States weggelassen werden.

### 8.5.1.3 Available (Neue Informationen verfügbar)

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

Sind Available Elemente vorhanden muss der Sender diese synchronisieren und seine offenen Prozesse aktualisieren.

### 8.5.1.4 OpenCase

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

Anschliessend wird der effektive Case mittels dieser Identifikation synchronisiert.

### 8.5.1.5 Leere Synchronisierung

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

## 8.6 UC006 Prozesskontrolle durchführen

Kurzbeschreibung	<p>Bei jeder Synchronisierung <b>muss</b> das Sendersystem 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 ( AwaitStory )</li> <li>• Hat der Endreceiver noch Stories aus anderen Prozessen aus dem selben Vertrag, die er dem Sendersystem liefern muss, müssen diese mittels Available zurückgegeben werden.</li> <li>• Hat der Endreceiver noch Prozesse, bei welchem der aktuelle State vom Sendersystem nicht bestätigt wurde müssen diese mittels Available zurückgegeben werden.</li> </ul>
------------------	---

	<ul style="list-style-type: none"> <li>• Sind weitere Änderungen notwendig?</li> </ul> <p>Das Sendersystem muss dem Benutzer aufzeigen, welche Stories er dem Endreceiver liefern muss und bis wann, falls unter der entsprechenden Story eine Frist gesetzt wurde.</p> <p>Ebenfalls <b>muss</b> dem Benutzer klar aufgezeigt werden, für welchen Fall noch Stories abgeholt werden müssen.</p>
Akteure	Sendersystem, Distributor, Endreceiver
Auslöser	Das Sendersystem führt eine Synchronisierung durch.
Vorbedingungen	Das Sendersystem hat mindestens eine Meldung gemacht, die nun synchronisiert werden kann.
Nachbedingungen	<p>Das Sendersystem informiert den User klar und verständlich darüber, welche Schritte als Nächstes auszuführen sind.</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 Transmitter überprüft, ob in der Response AwaitStory-Elemente enthalten sind. <b>Achtung!</b> Diese Funktion existiert nicht in allen Swissdec-Standards.</li> <li>3. Der Transmitter überprüft, ob in der Response Available-Elemente enthalten sind. <b>Achtung!</b> Diese Funktion existiert nicht in allen Swissdec-Standards.</li> <li>4. Dem Benutzer wird dargestellt, ob bestimmte Stories abgeholt oder geliefert werden müssen.</li> </ol>
Alternative Abläufe	keine
Fehlerliste	Analog zu <a href="#">Abschnitt 8.1, „UC001 Initiale Meldung senden“</a> , <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a>

Tabelle 8.8. UC006 Prozesskontrolle durchführen

## 8.7 UC007 Fall schliessen

Kurzbeschreibung	Ein Fall wird vom Endreceiver 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, Endreceiver
Auslöser	Seitens Sendersystem ist ein Fall abgeschlossen.
Vorbedingungen	<ul style="list-style-type: none"> <li>• Es besteht ein Fall, der Sendersystem und Endreceiver betrifft.</li> <li>• Der Endreceiver hat keine Stories mehr zu liefern ( &lt;Available&gt; )</li> <li>• Der Endreceiver erwartet keine Stories mehr ( &lt;AwaitStory&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 Prozesskontrolle durchführen“</a>

Standardablauf	<ol style="list-style-type: none"> <li>1. Das Sendersystem synchronisiert den zu schliessenden Fall.</li> <li>2. Das Sendersystem bearbeitet die Antwort vom Endreceiver gemäss dem normalen Ablauf ( <a href="#">Abschnitt 8.9, „UC009 Stories abholen und verarbeiten“</a> ).</li> <li>3. Das Sendersystem entscheidet anhand standardspezifischer Informationen, ob der Fall geschlossen werden kann. Dies wird in den Kapiteln zu den standardspezifischen Operationen genauer beschrieben.</li> <li>4. Das Sendersystem archiviert den Fall.</li> </ol>
Alternative Abläufe	<p><b>Der Endreceiver 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 8.9. UC007 Fall schliessen

## 8.8 UC008 Stories melden und quittieren

Kurzbeschreibung	Das Sendersystem meldet eine oder mehrere Stories an den Endreceiver
Akteure	Sendersystem, Distributor, Endreceiver
Auslöser	Das Sendersystem hat Informationen, die an den Endempfänger gelangen sollten.
Vorbedingungen	<ul style="list-style-type: none"> <li>• Der initiale 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 hat die Stories empfangen und quittiert.</li> </ul>
Included UCs	<a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a> , <a href="#">Abschnitt 8.6, „UC006 Prozesskontrolle durchführen“</a>
Standardablauf	<ol style="list-style-type: none"> <li>1. Im Sendersystem wird ausgewählt, welche Stories gemeldet werden sollen.</li> <li>2. Im Element &lt;ReceivedStoryIDs&gt; im &lt;CaseContext&gt; der vorangehenden Synchronize Operation quittiert der Endreceiver aller erhaltenen Stories.  Wurden vom Transmitter Stories versandt und diese werden nicht quittiert, <b>müssen</b> diese erneut gesendet werden.</li> <li>3. Die StoryID's aller vom Endreceiver empfangenen Stories (vom aktuellen Fall), werden im Element &lt;ReceivedStoryIDs&gt; im &lt;CaseContext&gt; eingetragen.</li> <li>4. Es wird ein SynchronizeRequest aufbereitet, der die ausgewählten Informationen beinhaltet.</li> <li>5. Der Fall wird synchronisiert ( <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a> )</li> </ol>
Alternative Abläufe	<ol style="list-style-type: none"> <li>1. <b>Keine Stories zu quittieren</b>  Es werden nur neue Stories gemeldet, da keine erhaltenen Stories zu quittieren sind.</li> <li>2. <b>Keine Stories zu melden</b></li> </ol>

	Das Sendersystem quittiert nur erhaltene Stories, hat dabei aber keine neuen Stories zu melden. Der Request enthält nur einen aktualisierten CaseContext aber keine weiteren Stories.
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:  Die Story mit diesen IDs können vom Endreceiver nicht empfangen werden. (z.B: Mapping)  Die ID's müssen für jeden folgenden Synchronize des aktuellen Prozesses versandt werden.</li> <li>3. Im Element &lt;CaseContext&gt;/&lt;SuppressedInstitutionStoryIDs&gt; sind StoryID's vorhanden:  Die Story mit diesen IDs können vom Transmitter nicht empfangen werden. (z.B: Mapping)  Die ID's müssen für jeden folgenden Synchronize 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.  Im CaseContext der Response können Info und Warning Notification vorhanden sein. Diese können mit der StoryID versehen werden, so dass der Akteur informiert werden kann, welche Story Probleme bereitet.</li> </ol>

Tabelle 8.10. UC008 Stories melden und quittieren

## 8.9 UC009 Stories abholen und verarbeiten

Kurzbeschreibung	Das Sendersystem holt Stories ab, die beim Endreceiver bereitliegen und verarbeitet sie im System.
Akteure	Sendersystem, Distributor, Endreceiver
Auslöser	Dem Sendersystem stehen Informationen zur Verfügung, die beim Endempfänger abgeholt werden <b>müssen</b> .
Vorbedingungen	<ul style="list-style-type: none"> <li>• Der Fall wurde dem Endreceiver erfolgreich gemeldet.</li> <li>• Identifikatoren sind bekannt und korrekt gesetzt.</li> <li>• Der Endreceiver hat dem Akteur mitgeteilt, dass Stories zur Abholung bereitliegen ( Available ).</li> </ul>
Nachbedingungen	<ul style="list-style-type: none"> <li>• Die vom Endreceiver bereitgestellten Stories wurden abgeholt</li> <li>• Die erhaltenen Informationen wurden ins System eingepflegt</li> </ul>
Included UCs	<a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a> , <a href="#">Abschnitt 8.6, „UC006 Prozesskontrolle durchführen“</a>
Standardablauf	<ol style="list-style-type: none"> <li>1. Der Akteur synchronisiert den Fall, von dem der Endreceiver mitgeteilt hat, dass noch Stories abzuholen sind. ( <a href="#">Abschnitt 8.5, „UC005 Synchronisieren“</a> )</li> <li>2. Der Akteur liest aus der SynchronizeResponse die erhaltenen Stories aus und pflegt sie ins Sendersystem ein.</li> <li>3. Im Element &lt;CaseContext&gt;/&lt;SuppressedSenderStoryIDs&gt; sind StoryIDs vorhanden:</li> </ol>

	<p>Die Storys mit diesen IDs können vom Endreceiver nicht empfangen werden. (z.B: Mapping)</p> <p>Die ID's <b>müssen</b> für jeden folgenden Synchronize des aktuellen Prozesses versandt werden.</p> <p>4. Im Element <code>&lt;CaseContext&gt;/&lt;SuppressedInstitutionStoryIDs&gt;</code> sind StoryID's vorhanden:</p> <p>Die Storys mit diesen IDs können vom Transmitter nicht empfangen werden (z.B: Mapping).</p> <p>Die ID's <b>müssen</b> für jeden folgenden Synchronize des aktuellen Prozesses versandt werden, so wird dem Endreceiver signalisiert, dass der Transmitter diese Story nicht unterstützt.</p>
Fehlerliste	<p>Fachliche Fehler:</p> <ul style="list-style-type: none"> <li>• Es werden keine Stories zurückgegeben.</li> <li>• Die Stories können nicht verarbeitet werden.</li> <li>• Die vom Sendersystem angeforderte Story-ID ist ungültig</li> </ul> <p>Technische Fehler:</p> <ul style="list-style-type: none"> <li>• Es gibt Probleme bei der Übermittlung. Die Übermittlung <i>kann</i> wiederholt werden.</li> </ul>

Tabelle 8.11. UC009 Stories abholen und verarbeiten

## 8.10 UC010 Datenflusskontrolle durchführen

Die Datenflusskontrolle soll verhindern, dass es zu einer Überlastung der betroffenen Systeme der Übermittlung kommt.

Der Endreceiver kann von sich aus beeinflussen, wie viele Informationen er innerhalb einer Response senden will, um die Auslastung seines Systems zu optimieren. Um die Infrastruktur eines schwächeren Sendersystems allerdings nicht zu überlasten, sind diesem die Möglichkeiten geboten, den Umfang einer Response einzuschränken.

Der Endreceiver teilt dem Sendersystem jeweils mit, zu welchen Fällen er noch Informationen zu liefern hat ( `<Available>` ). Das Sendersystem ist jedoch nicht verpflichtet, diese alle innerhalb des nächsten Requests abzurufen. Vielmehr besteht die Möglichkeit, den Umfang des Datenflusses zu kontrollieren, indem nur ein oder wenige Fälle zur Synchronisation ausgewählt werden. Der Endreceiver reagiert darauf, indem er die Antworten zu diesen Fällen liefert und für jene, die nicht abgefragt wurden, weiterhin mit `<Available>` reagiert.

Der Transmitter muss dem Benutzer die Möglichkeit geben, aus der Liste der verfügbaren Fälle jene auszuwählen, die bei der nächsten Synchronisation abgefragt werden sollen.

## 8.11 UC011 Completion aufrufen

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> </ul>

Nachbedingungen	<ul style="list-style-type: none"> <li>Ein Browserfenster mit eingetragenen Feldern Key und Passwort wurde geöffnet.</li> <li>Bei einem Fehlschlag: Webseite nicht erreichbar: Fehlermeldung</li> </ul>
Included UCs	-
Standardablauf	<ol style="list-style-type: none"> <li>Der Akteur wählt den Endreceiver aus, bei welchem eine Completion durchgeführt werden soll.</li> <li>Das Sendersystem stellt eine URL mit den Zusatzparametern Key und Passwort zusammen.</li> <li>Das Sendersystem öffnet ein Browserfenster mit der zusammengestellten URL.</li> </ol>
Alternative Abläufe	<p>Daten aus Story nicht vorhanden</p> <p>{nach Schritt 2} 3. Fehlermeldung {Ende}</p> <ol style="list-style-type: none"> <li>Der Akteur wählt den Endreceiver aus, bei welchem eine Completion durchgeführt werden soll.</li> <li>Das Sendersystem stellt eine URL mit den Zusatzparametern Key und Passwort sowie weiteren Parametern zusammen.</li> <li>Das Sendersystem öffnet ein Browserfenster mit der zusammengestellten URL.</li> </ol> <p>Alternativer Ablauf: DialogMessage</p> <p>Anstelle einer Completion-URL erhält das Sendersystem die Information, dass die Completion per DialogMessage durchgeführt werden soll (siehe <a href="#">Abschnitt 8.12, „UC012 Dialog Message abwickeln“</a>).</p>
Fehlerliste	<p>Fehler:</p> <ul style="list-style-type: none"> <li>Daten aus Story nicht vorhanden</li> </ul>

Tabelle 8.12. UC011 Completion aufrufen

## 8.11.1 Spezielle Anforderungen

### 8.11.1.1 Aufbau und Verarbeitung der Completion URL

[Anhang E, Detailspezifikation Completion URL](#)

## 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>Der Akteur wählt den Endreceiver aus, bei welchem eine DialogMessage abgewickelt werden soll.</li> <li>Der Endreceiver sendet mit der &lt;DialogMessage&gt; Story die zu ergänzenden Daten.</li> <li>Der Sachbearbeiter ergänzt die geforderten Angaben und sendet diese mit der &lt;DialogMessage&gt; Story mittels &lt;Synchronize&gt; an den Endreceiver.</li> </ol>



	4. Der Endreceiver bestätigt mit <ReceivedStoryIDs> den Erhalt der DialogMessage.
Alternative Abläufe	<p><b>Daten aus Statusmeldung nicht vorhanden</b></p> <p>{nach Schritt 1} 2. Fehlermeldung {Ende}</p> <p><b>DialogMessage nicht abgeschlossen</b></p> <p>{nach Schritt 4}</p> <p>1. Der Endreceiver bestätigt mit &lt;ReceivedStoryIDs&gt; den Erhalt der DialogMessage und verlangt weitere Ergänzungsangaben. {Weiter mit Schritt 4}</p> <p><b>DialogMessage nicht quittiert</b></p> <p>{nach Schritt 3}</p> <p>1. Der Endreceiver bestätigt den Erhalt der DialogMessage nicht. Die Story wird noch einmal vom Transmitter an den Empfänger geschickt. {Weiter mit Schritt 3}</p>
Fehlerliste	<p>Fehler:</p> <ul style="list-style-type: none"> <li>• Daten aus Statusmeldung nicht vorhanden</li> </ul>

Tabelle 8.13. UC012 Dialog Message abwickeln

### 8.13 UC013 Anmeldeprozess (SubscribeOrganization)

Kurzbeschreibung	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.
Akteure	Sachbearbeiter, ERP-System
Auslöser	Der Akteur möchte die Kommunikation via Swissdec bei einem Empfänger anmelden.
Vorbedingungen	<ul style="list-style-type: none"> <li>• Das Sendersystem ist in der Lage, elektronische Swissdec-Meldungen zu versenden und die Antwort zu verarbeiten.</li> <li>• Der Empfänger ist korrekt konfiguriert.</li> </ul>
Nachbedingungen	Die Anmeldung wurde vom Endreceiver empfangen und durch eine Rückantwort quittiert.
Included UCs	<a href="#">Abschnitt 8.1, „UC001 Initiale Meldung senden“</a> , <a href="#">Abschnitt 8.2, „UC002 Status abholen“</a> , <a href="#">Abschnitt 8.15, „UC015 Security anwenden“</a>
Standardablauf	<ol style="list-style-type: none"> <li>1. Das Sendersystem übergibt dem Transmitter die Daten mit den Empfängeradressen.</li> <li>2. Der Transmitter bereitet die Meldung als SOAP-Request mit zugehöriger Adressierung (Job) auf.</li> <li>3. Die Meldung wird nach Spezifikation signiert und verschlüsselt.</li> <li>4. Der Transmitter sendet die aufbereitete und signierte Meldung als synchronen Aufruf über HTTPS an den Distributor (siehe <a href="#">Abschnitt 8.1, „UC001 Initiale Meldung senden“</a>).</li> <li>5. Der Distributor prüft die Meldung auf Validität und Plausibilität.</li> <li>6. Der Distributor bereitet eine oder mehrere Meldungen für die gewählten Endreceiver auf und sendet diese an den/die Endreceiver.</li> </ol>



	<p>7. Der Endreceiver prüft die Meldung und startet die Verarbeitung des Jobs.</p> <p>8. Der Endreceiver meldet seine Antwort dem Distributor, der sie dem Sendersystem zukommen lässt.</p> <p>9. Der Transmitter wartet die Rückantwort des Distributors ab.</p> <p>10. Das Ergebnis der Übermittlung wird auf Transmitterseite aufbereitet und angezeigt.</p>
Alternative Abläufe	-
Fehlerliste	<p>Fachliche Fehler:</p> <ul style="list-style-type: none"> <li>• Die Meldung verstösst gegen die Plausibilisierungsregeln</li> <li>• Vertrag unbekannt.</li> </ul> <p>Technische Fehler:</p> <ul style="list-style-type: none"> <li>• Fehler beim Signieren oder Verschlüsseln. In diesem Fall <i>sollte</i> die Übermittlung zu einem späteren Zeitpunkt wiederholt werden, oder, falls das Problem langfristig besteht, Rückfrage beim Hersteller genommen werden.</li> <li>• Der Endreceiver befindet sich in einem geplanten Wartungsfenster (nur bei synchroner Übermittlung möglich). Der Benutzer <i>sollte</i> darauf hingewiesen werden, dass nach Ende des Wartungsfensters erneut übermittelt kann.</li> <li>• Der Endreceiver ist nicht erreichbar (nur bei synchroner Übermittlung möglich). Die Übermittlung kann zu einem späteren Zeitpunkt erneut versucht werden.</li> <li>• Der Endreceiver ist nicht adressierbar (nur bei synchroner Übermittlung möglich). Überprüfung der Adressierung seitens Sendersystem/Transmitter.</li> <li>• Die vom Sendersystem aufbereitete Meldung entspricht nicht dem Schema (Validität nicht gegeben)</li> </ul>

Tabelle 8.14. UC013 Anmeldungsprozess (SubscribeOrganization)

## 8.14 UC014 Unternehmensauthentifizierung (RegisterOrganization)

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. Weitere Informationen zu SUA finden sich unter [Anhang C, Detailspezifikation Swissdec Unternehmens-Authentifizierung SUA](#).

## 8.15 UC015 Security anwenden

Ausser dem Erreichbarkeitstest *muss* jede Übermittlung signiert und verschlüsselt werden. Einzelheiten dazu finden sich in den Dokumenten zur Sicherheit auf Transmitterseite [\[SECPDF\]](#), sowie in den zusätzlichen Informationen zur doppelten Signierung (SUA) [Anhang C, Detailspezifikation Swissdec Unternehmens-Authentifizierung SUA](#).

## 8.16 UC016 Transmitter konfigurieren

Das Sendersystem ist für die korrekte Adressierung der Endreceiver im Transmitter verantwortlich. Hierbei handelt es sich um die Pflege und regelmässige Aktualisierung der Profile mit gültiger Identifikation der einzelnen Institutionen.

Die jeweils gültigen Profile können direkt beim Endempfänger bezogen werden. Jedoch sind die an den Distributor gekoppelten Institutionen ebenfalls auf der Webseite der Swissdec (<https://www.swissdec.ch>) zum Download bereit.

Auf welche Weise und wie regelmässig die Profile aktualisiert werden, liegt in der Verantwortung des Sendersystem-Herstellers oder des Endbenutzers.

## 8.17 UC017 Supportinformationen anzeigen

Kurzbeschreibung	Fehler, Warnungen und Informationen gemäss <a href="#">[ACKNOTIF]</a> <b>müssen</b> ausgewertet und dem Benutzer angezeigt werden. IDs müssen verwendet werden.
Akteure	ERP, Transmitter, Distributor
Auslöser	Eine Meldung oder eine Anfrage wurde via Distributor an einen Endreceiver gesendet. Die Antwort wird via Distributor empfangen.
Vorbedingungen	Distributor sendet eine Antwort
Nachbedingungen	<ul style="list-style-type: none"> <li>Fehler, Warnungen und Informationen aus der Antwort werden aufbereitet und dem Benutzer angezeigt.</li> <li>Bei einem Fehlschlag: Distributor nicht erreichbar: Fehlermeldung</li> </ul>
Included UCs	<a href="#">Abschnitt 8.13, „UC013 Anmeldeprozess (SubscribeOrganization)“</a>
Standardablauf	<ol style="list-style-type: none"> <li>Das ERP übergibt dem Transmitter die Daten für eine Meldung.</li> <li>Der Transmitter bereitet die Meldung als SOAP-Request mit zugehöriger Adressierung (Job) und RequestID auf.</li> <li>Die Meldung wird mit dem privaten Schlüssel/Zertifikat des Herstellers nach Spezifikation <a href="#">[SECPDF]</a> signiert.</li> <li>Der Transmitter sendet die aufbereitete und signierte Meldung über HTTPS an den Distributor.</li> <li>Der Distributor prüft die Meldung auf Validität und Plausibilität.</li> <li>Der Distributor bereitet die Meldung für die gewählten Endempfänger inklusive einer Identifikation des Geschäftsvorfalles auf und sendet sie an den Endreceiver.</li> <li>Der Endreceiver prüft die Meldung. Eine Antwort wird generiert und an den Distributor gesendet.</li> <li>Der Distributor sendet eine Antwort inklusive einer Identifikation des Geschäftsvorfalles an den Transmitter.</li> <li>Der Transmitter wertet die Antwort aus und stellt sie für das ERP bereit.</li> <li>Fehler, Warnungen und Informationen werden aufbereitet und angezeigt. Die IDs werden gesichert.</li> </ol>
Alternative Abläufe	-
Fehlerliste	<p>Technische Fehler:</p> <ul style="list-style-type: none"> <li>Fehler beim Signieren</li> <li>Fehler beim ver-/ entschlüsseln</li> <li>der Distributor ist nicht erreichbar</li> <li>die vom ERP aufbereitete Meldung entspricht nicht dem Schema (Validität nicht gegeben)</li> </ul>

Tabelle 8.15. UC017 Supportinformationen anzeigen

### 8.17.1 Spezielle Anforderungen

#### 8.17.1.1 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 Transmitter 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.

### 8.17.1.2 Gleiche Fehlermeldungen und Informationen von verschiedenen Endempfängern darstellen

Bei gleichen Fehlermeldungen und Warnungen (Notifications) von verschiedenen Endempfängern muss für den Benutzer nachvollziehbar dargestellt sein, von welchen Empfängern welche Meldungen stammen. Ziel dieser Sicht ist es, den Support durch die Endempfänger zu erleichtern. Die Quittung eines Endempfängers bietet dem Endanwender und dem Support hierbei die gleichen Informationen.

Die elektronische Übermittlung ist ein dynamisches System, bei welchem die Antworten automatisch generiert werden, vergleiche [ACKNOTIF]. Einige Fehlermeldungen werden dabei von unterschiedlichen Endempfängern gleich lauten. Ist beispielsweise die Plausibilisierungsregel „Alter muss kleiner als 100 Jahre sein“ bei einer Person verletzt, so wird diese Fehlermeldung von allen Endempfängern kommen, bei welchen die Daten dieser Person verarbeitet wurden. Daher können neben der Forderung in [Abschnitt 8.17.1.2, „Gleiche Fehlermeldungen und Informationen von verschiedenen Endempfängern darstellen“](#) [58] zusätzliche Views, welche das Handling für den Endbenutzer erleichtern (Sortierungen, Filterung der redundanten Meldungen), erstellt werden.

### 8.18 UC018 Erreichbarkeit prüfen

Kurzbeschreibung	Die Erreichbarkeit des Distributors <b>muss</b> geprüft werden. Dazu wird eine einfache Anfrage an den Distributor geschickt. Die Rückantwort des Distributors bestätigt die Erreichbarkeit.
Akteure	Sachbearbeiter, Distributor
Auslöser	Die Erreichbarkeit des Distributors soll geprüft werden.
Vorbedingungen	-
Nachbedingungen	<ul style="list-style-type: none"><li>Die Rückantwort des Distributors enthält einen Timestamp mit der Systemzeit des Distributors.</li></ul> <p>Bei einem Fehlschlag:</p> <ul style="list-style-type: none"><li>Distributor nicht erreichbar: Fehlermeldung</li><li>Inhalt verschieden <a href="#">[ACKNOTIF]</a> : Fehlermeldung</li></ul>
Included UCs	-
Standardablauf	<ol style="list-style-type: none"><li>Der Akteur löst die Überprüfung aus.</li><li>Der Transmitter sendet eine einfache Serveranfrage (Ping) an die Zieladresse des Distributors.</li><li>Der Transmitter wertet die Rückantwort des Distributors aus <a href="#">[ACKNOTIF]</a> .</li></ol>
Alternative Abläufe	<b>Distributor nicht erreichbar</b>

	{nach Schritt 1} 2. Fehlermeldung {Ende}
Fehlerliste	Fehler: <ul style="list-style-type: none"> <li>• der Distributor ist nicht erreichbar</li> <li>• der Distributor sendet eine falsche Antwort</li> </ul>

Tabelle 8.16. UC018 Erreichbarkeit prüfen

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

## 8.19 UC019 Interoperabilität prüfen

Kurzbeschreibung	Damit die Interoperabilität zwischen einem Transmitter und dem Distributor überprüft werden kann, <b>muss</b> der Transmitter einen „CheckInteroperability-Request“ absetzen können.
Akteure	System-Tester, Distributor
Auslöser	Installation <i>soll</i> getestet werden
Vorbedingungen	<a href="#">Abschnitt 8.19.1.1, „Vorbedingungen“</a>
Nachbedingungen	<a href="#">Abschnitt 8.19.1.1, „Vorbedingungen“</a>  Bei einem Fehlschlag: <ul style="list-style-type: none"> <li>• Distributor nicht erreichbar: Fehlermeldung</li> <li>• Inhalt verschieden <a href="#">[[ACKNOTIF]]</a> : Fehlermeldung</li> </ul>
Included UCs	<a href="#">Abschnitt 8.13, „UC013 Anmeldeprozess (SubscribeOrganization)“</a>
Standardablauf	<ol style="list-style-type: none"> <li>1. Der Akteur startet die Interoperabilitätsprüfung und gibt Werte für Operand 2 ein.</li> <li>2. Der Akteur löst das Senden der Daten aus.</li> <li>3. Der Transmitter bereitet die Serveranfrage auf.</li> <li>4. Die Meldung wird mit dem privaten Schlüssel/Zertifikat des Herstellers nach Spezifikation <a href="#">[[SECPDF]]</a> signiert.</li> <li>5. Die Meldung wird mit einem elektronischen Verschlüsselungsverfahren wie in <a href="#">[[SECPDF]]</a> beschrieben verschlüsselt, die Antwort entschlüsselt.</li> <li>6. Der Transmitter sendet die Serveranfrage über HTTPS an den Distributor.</li> <li>7. Der Distributor bearbeitet die gesendeten Daten (Transformation Umlautstring, Berechnung „FirstOperand +- SecondOperand“) und schickt die Antwort an den Transmitter.</li> <li>8. Der Transmitter wertet die Antwort des Distributors aus <a href="#">[[ACKNOTIF]]</a> .</li> <li>9. Der Transmitter zeigt die Antwort des Distributors an.</li> </ol>
Alternative Abläufe	<b>Meldung signieren, Sender</b>  {nach Schritt 4}  b) Die Meldung wird mit dem privaten Schlüssel/Zertifikat des Sender nach Spezifikation <a href="#">[[SECPDF]]</a> ein zweites Mal signiert.  {weiter mit Schritt 5}  <b>Distributor nicht erreichbar</b>  {nach Schritt 5}  6. Fehlermeldung

	{Ende}
Fehlerliste	<p>Fachliche Fehler:</p> <ul style="list-style-type: none"> <li>• Interoperabilität ist nicht gegeben</li> </ul> <p>Technische Fehler:</p> <ul style="list-style-type: none"> <li>• Fehler beim Signieren</li> <li>• Fehler beim ver-/ entschlüsseln</li> <li>• der Distributor ist nicht erreichbar</li> </ul> <p>siehe <a href="#">[ACKNOTIF]</a></p>

Tabelle 8.17. UC019 Interoperabilität prüfen

### 8.19.1 Spezielle Anforderungen

Der Interoperabilitätstest wird zu Entwicklungszwecken und bei der Installation verwendet, um die Interoperabilität zwischen einem Transmitter und dem Distributor zu gewährleisten. Die grössten zu erwartenden Schwierigkeiten liegen dabei in den Bereichen Codierung von Zeichenketten (Encoding) und Interpretation von Fließkommazahlen. Beide Systeme (Transmitter und Distributor) müssen dabei bestimmte Auswertungen vornehmen, um bei einem eventuellen Fehler auf den Verursacher schliessen zu können.

#### 8.19.1.1 Vorbedingungen

Der Transmitter sendet folgende Daten:

Parametername	Wert	Bemerkung
UmlautString	ÄËÖÜÁÉÓÚÀÈÒÙÂÊÔÛ	fester Wert
FirstOperand	999000000000.00	fester Wert, 999 Milliarden
SecondOperand	keine Vorgabe	beliebige Fließkommazahl
SystemDateTime	Datum und Zeit des Transmitters	Systemdatum und –zeit

Tabelle 8.18. Vorbedingungen (Transmitter)

#### 8.19.1.2 Nachbedingungen

Auswertung und Antwort des Distributors:

Parametername	Auswertung / Berechnung	Bemerkung
UmlautStringIsCorrect	UmlautStringTRANS ÄËÖÜÁÉÓÚÀÈÒÙÂÊÔÛ	= Rückgabe: true / false
FirstOperandIsCorrect	999000000000.00	Rückgabe: true / false
UmlautString	äëöüáéóúàèòùâêôû	Rückgabe: UmlautStringDISTRIGross-zu Kleinbuchstaben.
AdditionResult	AdditionResultDISTRIGross-zu Kleinbuchstaben = FirstOperandTRANS + SecondOperandTRANS	Rückgabe: berechneter Wert AdditionResultDISTRIGross-zu Kleinbuchstaben.
SubtractionResult	AddSubtractionResultDISTRIGross-zu Kleinbuchstaben = FirstOperandTRANS - SecondOperandTRANS	Rückgabe: berechneter Wert SubtractionResultDISTRIGross-zu Kleinbuchstaben.
SystemDateTime	Datum und Zeit des Distributors	Rückgabe: Systemdatum und –zeit

Tabelle 8.19. Auswertung und Antwort Distributor

Auswertung des Transmitters:

Parametername	Auswertung / Berechnung	Bemerkung
UmlautStringIsCorrect	UmlautStringIsCorrect = true	<i>muss</i> true sein
FirstOperandIsCorrect	FirstOperandIsCorrect = true	<i>muss</i> true sein
UmlautString	äëöüáéóúàèòùâêôû	<i>muss</i> äëöüáéóúàèòùâêôû sein
AdditionResult	FirstOperandTRANS + SecondOperandTRANS = AdditionResultDISTR	Berechnung und Vergleich, Genauigkeitsgrad 2 Nachkommastellen
SubstractionResult	FirstOperandTRANS – SecondOperandTRANS = AdditionResultDISTR	Berechnung und Vergleich, Genauigkeitsgrad 2 Nachkommastellen
SystemDateTime	SystemDateTimeDISTR – SystemDateTimeTRANS  < 1 min	Betrag Zeitdifferenz sollte < 1 Minute sein

Tabelle 8.20. Auswertung Transmitter

## 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 Identisch mit Angaben beim Versicherer
Bestehende Vertragsverbindung (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.

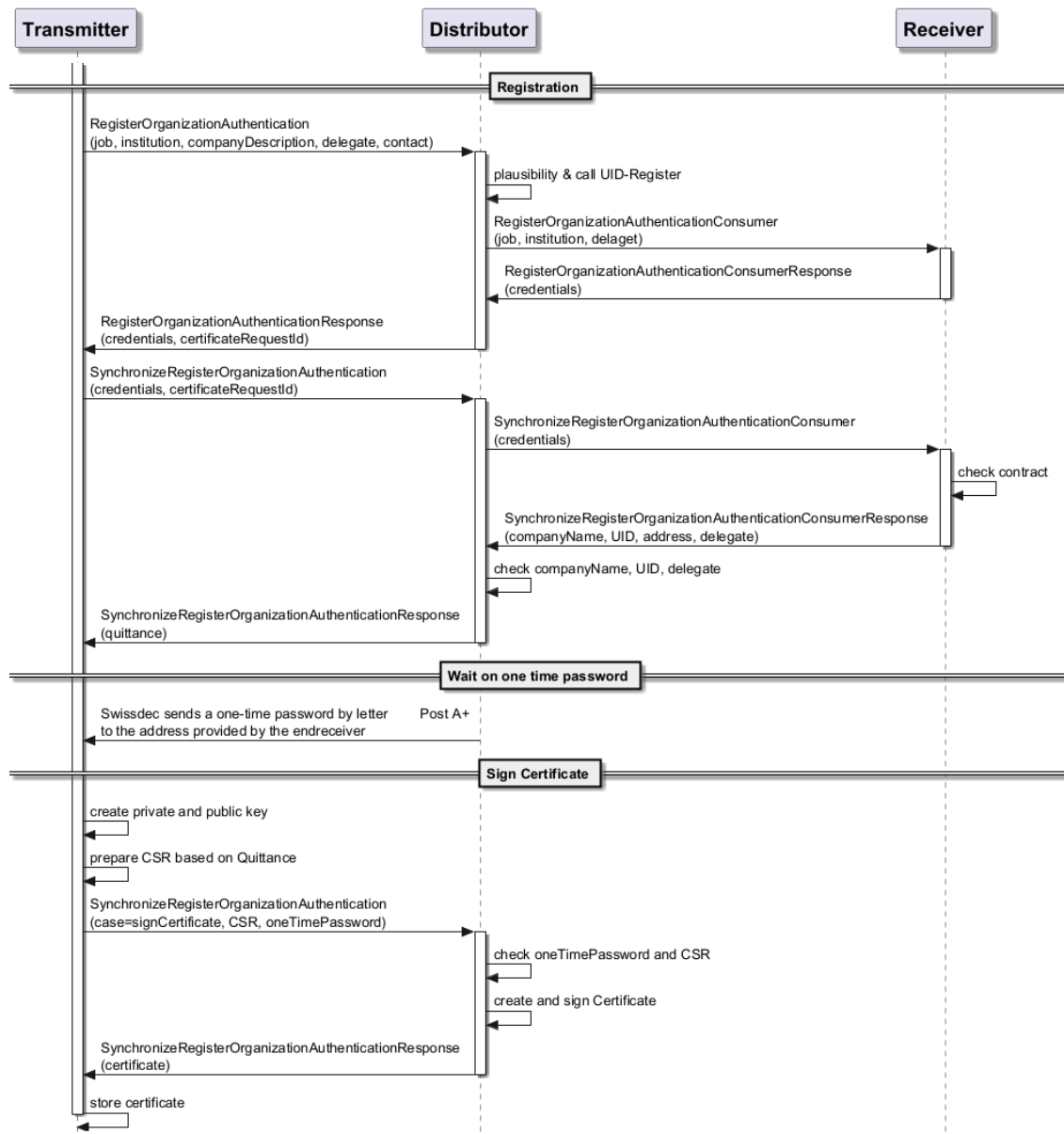
Ein Mitarbeiter des Unternehmens meldet die benötigten Stammdaten des Unternehmens über den Transmitter beim Distributor an. Dafür werden die UID-Nummer, der Firmenname sowie die bestehende Vertragsbeziehung mit dem adressierten Versicherer an den Distributor gemeldet.

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-

heissmassnahme 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. 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 RSA 3072 RSA 4096	Akzeptierte Schlüssellänge
Signature Algorithm	Sha256WithRSA	Verwendeter Signature Algorithmus
Signature Value	{Wert}	Signatur des Zertifikats
<b>Issuer</b>		
CN	Verein Swissdec 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 <i>Optional</i>	{Kanton des Unternehmens}	StateOrProvinceName (OID 2.5.4.8)
L <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 <Antwort>	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.

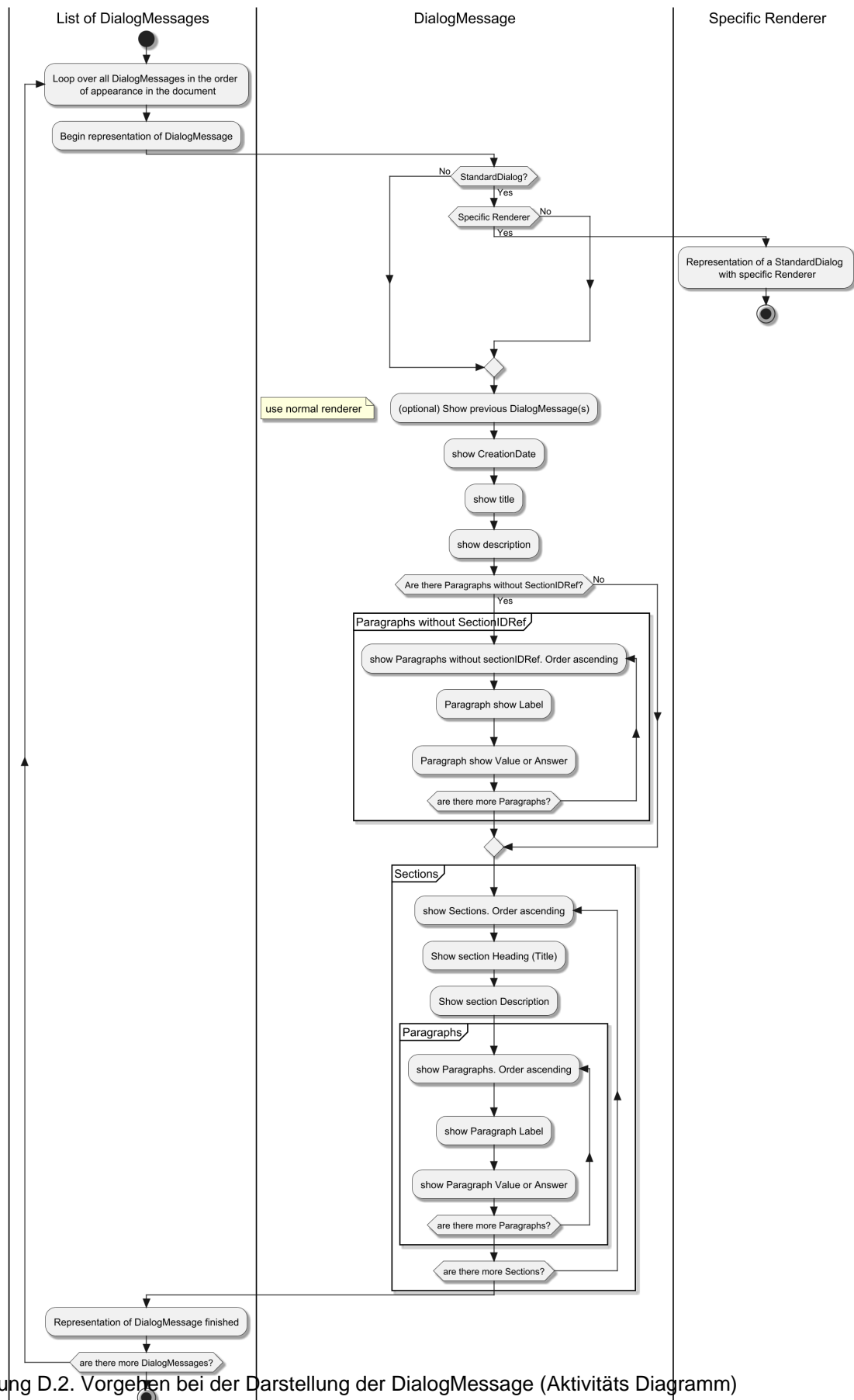


Abbildung D.2. Vorgehen bei der Darstellung der DialogMessage (Aktivitäts Diagramm)

## 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`

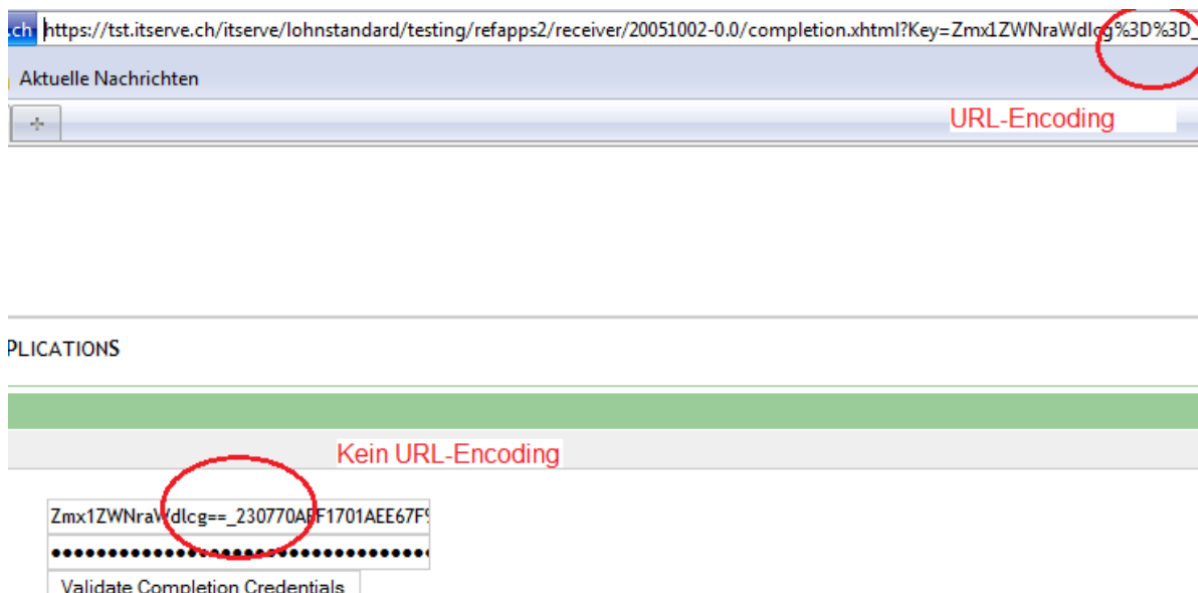


Abbildung E.1. URL-Encoding in der Completion

### 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%23dl%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[\[RFC3986\]](#):

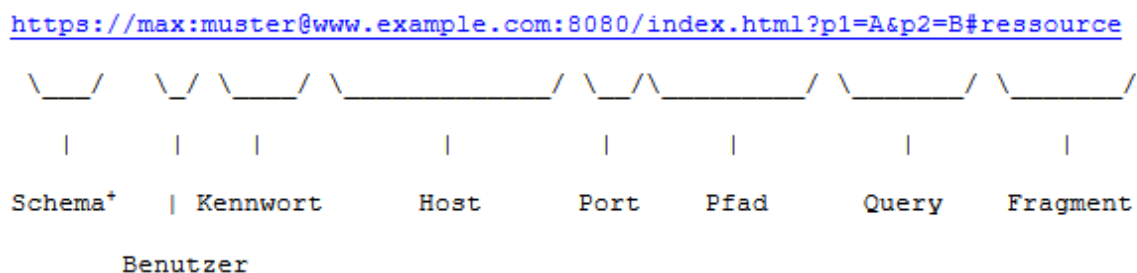


Abbildung E.2. Aufbau URL gemäss RFC3986



## 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 Distributors muss geprüft werden. Dazu wird eine einfache Anfrage an den Distributor geschickt. Die Rückantwort des Distributors bestätigt die Erreichbarkeit
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 einfache Serveranfrage (Ping) an die Zieladresse des Distributors.</li> <li>3. Das Sendersystem wertet die Rückantwort des Distributors aus.</li> </ol>
Ergebnis	Die Rückantwort des Distributors enthält einen Timestamp mit der Systemzeit des Distributors oder eine Fehlermeldung
Fehlerbehandlung	1. Distributor 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	<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	<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. Das Sendersystem muss dem Benutzer die benötigten IDs anzeigen. Dies darf abhängig vom Benutzerstatus sein (Endbenutzer, tech. Support, etc.)</li> <li>3. 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>

<b>AB-08 Identifikation eines Geschäftsvorfalles</b>	
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 Sendersystem sendet die Ersatzmeldung an den Distributor.</li> <li>6. Das Sendersystem wertet die Rückantwort des Distributors aus.</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	keine

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. Eine 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 Laufes 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> <p>Damit soll die Qualität des gesamten Geschäftsprozesses erhöht werden, d.h.</p>

AB-12 Plausibilisierung	
	<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 Die eingehenden und ausgehenden Meldungen werden gegen das XML-Schema validiert. Beim Distributor lassen sich neue Checks für den Geschäftsprozess „einfach“ einführen. Die <b>Plausibilisierungsregeln</b> müssen daher nur hier umgesetzt und geprüft werden.</li> <li>• System Transmitter (Client, Requester) Die eingehenden und ausgehenden Meldungen werden gegen das XML-Schema validiert. Die ausgehenden Meldungen <i>können</i> ebenfalls gemäss den definierten Plausibilisierungsregeln geprüft werden.</li> <li>• Endrezeiver (Responder, Server) Die eingehenden und ausgehenden Meldungen werden gegen das XML-Schema validiert. Nur der Endrezeiver 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. Das Sendersystem kann die Meldung vor dem Versenden auf ihre Plausibilität prüfen. Dies verhindert etwaige Fehlermeldungen der Plausibilisierungsstufe nach dem Versand der Daten.</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

AB-13 Filterung	
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 Endrezeivern). Eine gleichzeitige Verteilung an mehrere Endrezeiver 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	Etwaige 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 Sendersystem
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. 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 geändert werden kann/darf. Aus diesen Gründen müssen bestimmte Teile der Struktur anonymisiert werden.



<b>AB-16 Anonymisierung</b>	
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. 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	1. Das Sendersystem muss ein SUA-Zertifikat anfordern und verwalten können. 2. Das Sendersystem muss die zu übermittelnde Meldung mit dem SUA-Zertifikat signieren (doppelte Signatur).
Ergebnis	SUA wird vom Sendersystem gemäss <a href="#">Anhang C, Detailspezifikation Swissdec Unternehmens-Authentifizierung SUA</a> umgesetzt.
Fehlerbehandlung	Fehlermeldungen im Rahmen des SUA-Prozesses müssen dem Teilnehmer dargestellt 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 AB-19 Archivdateien

<b>AB-19 Archivdateien</b>	
Fachliche Anforderung	Übermittelte Requests und empfangene Responses <b>müssen</b> signiert aber unverschlüsselt archiviert werden. Die Dauer der Archivierung hängt von fachlichen Anforderungen ab. Nicht relevante Daten müssen nicht archiviert werden (Polling, etc. Siehe Use Cases)
Abhängige Bausteine	keine
Ausschliessende Bausteine	keine

<b>AB-19 Archivdateien</b>	
Anforderungen an Akteure	<ol style="list-style-type: none"> <li>1. Der Akteur sichert fachlich relevante Requests nach dem Signieren aber vor der Verschlüsselung.</li> <li>2. Der Akteur sichert fachlich relevante Responses nach dem Entschlüsseln, belässt aber die Signatur auf der Response-Datei.</li> <li>3. Archivdateien müssen gemäss gesetzlichen Anforderungen aufbewahrt werden.</li> </ol>
Ergebnis	Sämtliche inhaltlich relevanten Requests und Responses werden mit gültiger Signatur abgelegt. Der Benutzer kann während eines fachlich vorgegebenen Zeitraums darauf zugreifen.
Fehlerbehandlung	Fehler bei der Archivierung von Daten müssen dem Benutzer kommuniziert werden.
Technische Implikationen	keine

Tabelle F.18. AB-19 Archivdateien

## F.20 SB-01 Transportsicherheit (HTTPS / TLS)

<b>SB-01 Transportsicherheit (HTTPS / TLS)</b>	
Fachliche Anforderung	keine
Abhängige Bausteine	keine
Ausschliessende Bausteine	keine
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 Sendersystem und Distributor ist verschlüsselt.
Fehlerbehandlung	Fehler in der Transportsicherheit: Abbruch der Übermittlung
Technische Implikationen	Siehe <a href="#">[SECPDF]</a>

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

## F.21 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.20. SB-02 Transport Authentication (TLS Mutual Authentication)

## F.22 SB-03 Nutzdatenverschlüsselung (WS-Encryption)

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

## F.23 SB-04 Nutzdatensignatur (WS-Signature)

SB-04 Nutzdatensignatur (WS-Signature)	
Fachliche Anforderung	keine
Abhängige Bausteine	keine
Ausschliessende Bausteine	keine
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.22. SB-04 Nutzdatensignatur (WS-Signature)

## F.24 SB-05 Unternehmens-Authentifizierung SUA Signatur (WS-Signatur)

SB-05 Unternehmens-Authentifizierung SUA Signatur (WS-Signatur)	
Fachliche Anforderung	keine
Abhängige Bausteine	keine
Ausschliessende Bausteine	keine
Anforderungen an Akteure	SUA wird gemäss Spezifikation umgesetzt. Es kann ein SUA-Zertifikat angefordert und installiert werden, und mit diesem Zertifikat kann ein Request doppelt signiert werden.
Ergebnis	Ein gültiges SUA-Zertifikat kann angefordert und installiert werden. Mit diesem Zertifikat kann doppelt signiert werden.
Fehlerbehandlung	<ol style="list-style-type: none"> <li>1. Kein SUA-Zertifikat vorhanden: Der SUA-Prozess zum Anfordern eines Zertifikats muss durchgeführt werden.</li> <li>2. Fehler im Anforderungsprozess eines SUA-Zertifikats. Abbruch und erneute Ausführung ab Beginn.</li> </ol>

<b>SB-05 Unternehmens-Authentifizierung SUA Signatur (WS-Signatur)</b>	
	3. Fehler bei der Nachrichtensicherheitsüberprüfung. Abbruch der Übermittlung.
Technische Implikationen	Siehe <a href="#">[SECPDF]</a>

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

## F.25 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.24. SB-06 Nichtabstreitbarkeit

## F.26 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.
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.25. PB-01 Synchrone Verteilung

## F.27 PB-02 Asynchrone Verteilung

<b>PB-02: Asynchrone 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

<b>PB-02: Asynchrone Verteilung</b>	
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.26. PB-02 Asynchrone Verteilung

## F.28 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. Zusätzlich muss zwischen den technischen und fachlichen Quittungen unterschieden werden. Der Datenaustausch zwischen Unternehmen und Versicherer erfolgt in strukturierter Form während des gesamten Fallverlaufs, bei dem beide Seiten den gleichen Informationsstand zum Ereignis aufrechterhalten.

Tabelle F.27. PB-03 Synchronisierung

## F.29 PB-04 Dialogisierung

<b>PB-04: Dialogisierung</b>	
Fachliche Anforderung	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. Weitere Details finden sich bei der Swissdec DialogMessage.
Abhängige Bausteine	
Ausschliessende Bausteine	
Anforderungen an Akteure	Innerhalb der Automatisierung muss bei neuen DialogMessages auch eine manuelle Verarbeitung möglich sein.

<b>PB-04: Dialogisierung</b>	
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.28. PB-04 Dialogisierung

### F.30 PB-05 Completion

<b>PB-05: Completion</b>	
Fachliche Anforderung	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.
Abhängige Bausteine	
Ausschliessende Bausteine	
Anforderungen an Akteure	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. Weitere Details finden sich bei der Swissdec ELM Completion.
Ergebnis	
Fehlerbehandlung	
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.29. PB-05 Completion

### F.31 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. Die einfachste Form ist ein Fortschrittsbalken mit %-Angaben.
Abhängige Bausteine	Es besteht evtl. eine semantische Abhängigkeit oder Überlappung mit den möglichen <b>Protokoll-Zuständen</b> .
Ausschliessende Bausteine	
Anforderungen an Akteure	Einerseits muss diese Navigation geführt und übermittelt werden. Andererseits muss der Kontext mit dem aktuellen Ort visualisiert werden.
Ergebnis	Alle Teilnehmer können sich dadurch orientieren und gewinnen damit mehr Verständnis, Gewissheit und Vertrauen.

<b>PB-06: Prozessnavigation</b>	
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.30. PB-06 Prozessnavigation

## F.32 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.31. PB-07 Digitalisierungsbereich

## F.33 PB-08 Datenanfordern

<b>PB-08: Datenanfordern</b>	
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	

<b>PB-08: Datenanfordern</b>	
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.32. PB-08 Datenanfordern



## G Technische Dokumentationen

### G.1 Schema Dokumentation BalanceSheetDeclarationServiceTypes.xsd

Target Namespace	urn:ch:swissdec:ebilanz:v1:20260306:balancesheetdeclaration:service:types
Declared Namespaces	<ul style="list-style-type: none"> <li>eCH-0276 : http://www.ech.ch/xmlns/eCH-0276/1</li> <li>bsc : urn:ch:swissdec:ebilanz:v1:20260306:balancesheetdeclaration:container</li> <li>ep : urn:ch:swissdec:basis:v1:20260306:components</li> <li>xs : http://www.w3.org/2001/XMLSchema</li> <li>bsst : urn:ch:swissdec:ebilanz:v1:20260306:balancesheetdeclaration:service:types</li> <li>xbml : http://www.xbml.org/2003/instance</li> </ul>
Version	0.0

### G.2 Schema Dokumentation BalanceSheetDeclarationContainer.xsd

Target Namespace	urn:ch:swissdec:ebilanz:v1:20260306:balancesheetdeclaration:container
Declared Namespaces	<ul style="list-style-type: none"> <li>eCH-0276 : http://www.ech.ch/xmlns/eCH-0276/1</li> <li>bsc : urn:ch:swissdec:ebilanz:v1:20260306:balancesheetdeclaration:container</li> <li>ep : urn:ch:swissdec:basis:v1:20260306:components</li> <li>xs : http://www.w3.org/2001/XMLSchema</li> <li>xbml : http://www.xbml.org/2003/instance</li> </ul>
Version	0.0

### ComplexType: DeclareBalanceSheetAddresseeContextType

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;DeclarationID&gt;„SimpleType: IDType“ &lt;/DeclarationID&gt; [1..1] &lt;/...&gt; </pre>

### ComplexType: DeclareBalanceSheetAddresseeJobStateType

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 ] </pre>

	<pre> &lt;ProcessByDistributor&gt; <a href="#">„SimpleType: SimpleBooleanType“</a> &lt;/ProcessByDistribu\ tor&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;Success&gt; <a href="#">„ComplexType: DeclareBalanceSheetAddresseeSuccessJobStateType“</a> &lt;/Suc\ cess&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre>
--	--

### ComplexType: DeclareBalanceSheetAddresseeSuccessJobStateType

Abstract	no
Translation	OpenCase Status
Short description	Initialer Status des Falls.
XML Instance Representation	<pre> &lt;...&gt; &lt;AddresseeContext&gt; <a href="#">„ComplexType: DeclareBalanceSheetAddresseeContextType“</a> &lt;/Addres\ seeContext&gt; [1..1] &lt;Credentials&gt; <a href="#">„ComplexType: CredentialsType“</a> &lt;/Credentials&gt; [1..1] &lt;/...&gt; </pre>

### ComplexType: DeclareBalanceSheetAddresseeType

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: DeclareBalanceSheetCaseContextType

Abstract	no
Parent type	<a href="#">„ComplexType: CaseContextType“</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: CaseContextType“</a> ] [ BASE TYPE <a href="#">„ComplexType: CaseContextBaseType“</a> ] &lt;ReceivedStoryIDs&gt; <a href="#">„ComplexType: ReceivedStoriesType“</a> &lt;/ReceivedSto\ ryIDs&gt; [0..1] &lt;SuppressedSenderStoryIDs&gt; <a href="#">„ComplexType: ReceivedStoriesType“</a> &lt;/SuppressedSen\ derStoryIDs&gt; [0..1] &lt;SuppressedInstitutionStoryIDs&gt; <a href="#">„ComplexType: ReceivedStoriesType“</a> &lt;/Suppresse\ dInstitutionStoryIDs&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;Credentials&gt; <a href="#">„ComplexType: CredentialsType“</a> &lt;/Credentials&gt; [1..1] [ END BASE TYPE ] &lt;DeclarationID&gt; <a href="#">„SimpleType: IDType“</a> &lt;/DeclarationID&gt; [1..1] &lt;/...&gt; </pre>

### ComplexType: DeclareBalanceSheetCaseType

Abstract	no
Translation	Fall
Technical description	Informationen zum Fall
XML Instance Representation	<pre> &lt;...&gt; &lt;CaseContext&gt; <a href="#">„ComplexType: DeclareBalanceSheetCaseContextType“</a> &lt;/CaseCon\ text&gt; [1..1] &lt;ReceivedState&gt; <a href="#">„SimpleType: DeclareBalanceSheetStateType“</a> &lt;/ReceivedState&gt; [0..1] &lt;/...&gt; </pre>

### ComplexType: DeclareBalanceSheetConsumerCaseType

Abstract	no
----------	----

Translation	Fall
Technical description	Informationen zum Fall
XML Instance Representation	<pre>&lt;...&gt;   &lt;CaseContext&gt; „ComplexType: DeclareBalanceSheetCaseContextType“ &lt;/CaseCon\     text&gt; [1..1]   &lt;State&gt; „SimpleType: DeclareBalanceSheetStateType“ &lt;/State&gt; [1..1] &lt;/...&gt;</pre>

### ComplexType: DeclareBalanceSheetJobStateType

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

### ComplexType: DeclareBalanceSheetJobType

Abstract	no
Translation	Job Steuerung
Technical description	Steuerungselement für die Verarbeitung der Meldung.
XML Instance Representation	<pre>&lt;...&gt;   &lt;Addressees&gt; „ComplexType: DeclareBalanceSheetAddresseesType“ &lt;/Addressees&gt; [1..1]   &lt;Substitution&gt; „ComplexType: DeclareBalanceSheetSubstitutionType“ &lt;/Substituti\     on&gt; [0..1] &lt;/...&gt;</pre>

### ComplexType: DeclareBalanceSheetRequestType

Abstract	no
Parent type	„ComplexType: RequestType“
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: DeclareBalanceSheetJobType“ &lt;/Job&gt; [1..1]   &lt;null&gt;bsc:blubber&lt;/null&gt; [1..1] &lt;/...&gt;</pre>

### ComplexType: DeclareBalanceSheetSubstitutionType

Abstract	no
Technical description	Entschädigung Job
XML Instance Representation	<pre>&lt;...&gt;   &lt;PredecessorDeclarationIDWithAcceptedState&gt; „SimpleType: IDType“ &lt;/PredecessorDecla\     rationIDWithAcceptedState&gt; [1..1] &lt;/...&gt;</pre>

### ComplexType: DeclareRawBalanceSheetAddresseeContextType

Abstract	no
Parent type	„ComplexType: AddresseeResponseContextType“
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]</pre>

	<pre> &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;DeclarationID&gt; „SimpleType: IDType“ &lt;/DeclarationID&gt; [1..1] &lt;/...&gt; </pre>
--	---

### ComplexType: DeclareRawBalanceSheetAddresseeJobStateType

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: DeclareRawBalanceSheetAddresseeSuccessJobStateType“ &lt;/Success&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre>

### ComplexType: DeclareRawBalanceSheetAddresseeSuccessJobStateType

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

### ComplexType: DeclareRawBalanceSheetAddresseesType

Abstract	no
Technical description	Angaben zum Adressaten
XML Instance Representation	<pre> &lt;...&gt; &lt;/...&gt; </pre>

### ComplexType: DeclareRawBalanceSheetCaseContextType

Abstract	no
Parent type	<a href="#">„ComplexType: CaseContextType“</a>
Translation	Kontext des Falls
Technical description	Dieses Element enthält Informationen und Identifikatoren zum Fall.
XML Instance Representation	<pre> &lt;...&gt; [ BASE TYPE „ComplexType: CaseContextType“ ] [ 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] </pre>

	<pre> [ END BASE TYPE ] &lt;Credentials&gt;„ComplexType: CredentialsType“ &lt;/Credentials&gt; [1..1] [ END BASE TYPE ] &lt;DeclarationID&gt;„SimpleType: IDType“ &lt;/DeclarationID&gt; [1..1] &lt;/...&gt; </pre>
--	---

### ComplexType: DeclareRawBalanceSheetCaseType

Abstract	no
Translation	Fall
Technical description	Informationen zum Fall
XML Instance Representation	<pre> &lt;...&gt; &lt;CaseContext&gt;„ComplexType: DeclareRawBalanceSheetCaseContextType“ &lt;/CaseCon\ text&gt; [1..1] &lt;ReceivedState&gt;„SimpleType: DeclareRawBalanceSheetStateType“ &lt;/ReceivedSta\ te&gt; [0..1] &lt;/...&gt; </pre>

### ComplexType: DeclareRawBalanceSheetConsumerCaseType

Abstract	no
Translation	Fall
Technical description	Informationen zum Fall
XML Instance Representation	<pre> &lt;...&gt; &lt;CaseContext&gt;„ComplexType: DeclareRawBalanceSheetCaseContextType“ &lt;/CaseCon\ text&gt; [1..1] &lt;State&gt;„SimpleType: DeclareRawBalanceSheetStateType“ &lt;/State&gt; [1..1] &lt;Completion&gt;„ComplexType: CompletionType“ &lt;/Completion&gt; [0..1] &lt;/...&gt; </pre>

### ComplexType: DeclareRawBalanceSheetJobStateType

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

### ComplexType: DeclareRawBalanceSheetJobType

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;Substitution&gt;„ComplexType: DeclareRawBalanceSheetSubstitutionType“ &lt;/Substituti\ on&gt; [0..1] &lt;/...&gt; </pre>

### ComplexType: DeclareRawBalanceSheetRequestType

Abstract	no
Parent type	„ComplexType: RequestType“
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: DeclareRawBalanceSheetJobType“ &lt;/Job&gt; [1..1] &lt;null&gt;bsc:blubber&lt;/null&gt; [1..1] &lt;/...&gt; </pre>

## ComplexType: DeclareRawBalanceSheetResponseType

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: DeclareRawBalanceSheetJobStateType“ &lt;/Addres\ sees&gt; [1..1] &lt;/...&gt;</pre>

## ComplexType: DeclareRawBalanceSheetSubstitutionType

Abstract	no
Technical description	Entschädigung Job
XML Instance Representation	<pre>&lt;...&gt; &lt;PredecessorDeclarationIDWithAcceptedState&gt; „SimpleType: IDType“ &lt;/PredecessorDecla\ rationIDWithAcceptedState&gt; [1..1] &lt;/...&gt;</pre>

## ComplexType: GetStatusFromDeclareBalanceSheetResponseType

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

## ComplexType: SynchronizeDeclareBalanceSheetConsumerType

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: DeclareBalanceSheetConsumerCaseType“ &lt;/Case&gt; [1..1] &lt;/...&gt;</pre>

## ComplexType: SynchronizeDeclareBalanceSheetRequestType

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: AddresseeType“ &lt;/Addressee&gt; [1..1] &lt;Case&gt; „ComplexType: DeclareBalanceSheetCaseType“ &lt;/Case&gt; [1..1] &lt;/...&gt;</pre>

## ComplexType: SynchronizeDeclareBalanceSheetResponseType

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;SynchronizeDeclareBalanceSheetConsumer&gt; „ComplexType: SynchronizeDeclareBalanceSheetConsumerType“ &lt;/SynchronizeDeclareBalanceSheetConsumer&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre>

## ComplexType: SynchronizeDeclareRawBalanceSheetConsumerType

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;/AddresseeContext&gt; [1..1] [ END BASE TYPE ] &lt;Addressee&gt; „ComplexType: AddresseeType“ &lt;/Addressee&gt; [1..1] &lt;Case&gt; „ComplexType: DeclareRawBalanceSheetConsumerCaseType“ &lt;/Case&gt; [1..1] &lt;/...&gt; </pre>

## ComplexType: SynchronizeDeclareRawBalanceSheetRequestType

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: AddresseeType“ &lt;/Addressee&gt; [1..1] &lt;Case&gt; „ComplexType: DeclareRawBalanceSheetCaseType“ &lt;/Case&gt; [1..1] &lt;/...&gt; </pre>

## ComplexType: SynchronizeDeclareRawBalanceSheetResponseType

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;SynchronizeDeclareRawBalanceSheetConsumer&gt; „ComplexType: SynchronizeDeclareRawBalanceSheetConsumerType“ &lt;/SynchronizeDeclareRawBalanceSheetConsumer&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre>

## SimpleType: DeclareBalanceSheetStateType

Content type	Restriction
Base type	xs:string
Facets	<ul style="list-style-type: none"> <li>Enum: Accepted</li> <li>Enum: Processing</li> <li>Enum: Finished</li> </ul>

## SimpleType: DeclareRawBalanceSheetStateType

Content type	Restriction
--------------	-------------

Base type	xs:string
Facets	<ul style="list-style-type: none"> <li>• Enum: Accepted</li> <li>• Enum: CompletionReleaseMissing</li> <li>• Enum: Finished</li> </ul>

### G.3 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] &lt;/...&gt;</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 „ComplexType: AddresseeType“ ]   &lt;AddresseeIdentification&gt;„SimpleType: IDType“ &lt;/AddresseeIdentification&gt; [1..1]   [ END BASE TYPE ]   &lt;ProcessByDistributor&gt;„SimpleType: SimpleBooleanType“ &lt;/ProcessByDistribu\ tor&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 „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;/Producer\ ResponseNotifications&gt; [1..1]   &lt;Warning&gt;„ComplexType: NotificationsType“ &lt;/Warning&gt; [0..1]   &lt;Info&gt;„ComplexType: NotificationsType“ &lt;/Info&gt; [0..1] &lt;/...&gt;</pre>

#### ComplexType: AddresseeResponseType

Abstract	no
XML Instance Representation	<pre>&lt;...&gt;   &lt;AddresseeContext&gt;„ComplexType: AddresseeResponseContextType“ &lt;/AddresseeCon\ text&gt; [1..1] &lt;/...&gt;</pre>



### ComplexType: AddresseeType

Abstract	no
Short description	Informationen zum Adressaten
XML Instance Representation	<pre>&lt;...&gt;   &lt;AddresseeIdentification&gt;_SimpleType: IDType_ &lt;/AddresseeIdentification&gt; [1..1] &lt;/...&gt;</pre>

### ComplexType: AnswerAmountType

Abstract	no
XML Instance Representation	<pre>&lt;...&gt;   &lt;Default&gt;_SimpleType: SalaryAmountType_ &lt;/Default&gt; [0..1]   &lt;Value&gt;_SimpleType: SalaryAmountType_ &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;/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] &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;/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;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 „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;Completion&gt;„ComplexType: AccessInformationType“&lt;/Completion&gt; [1..1] &lt;Credentials&gt;„ComplexType: ResultType“&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 „ComplexType: ResponseType“ ] &lt;ResponseContext&gt;„ComplexType: ResponseContextType“&lt;/ResponseContext&gt; [1..1] [ END BASE TYPE ] &lt;FaultState&gt;„ComplexType: FaultStateType“&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] &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;/ProducerResponseNotifications&gt; [1..1] &lt;ConsumerNotifications&gt;<a href="#">ComplexType: FeedbackNotificationsType</a>&lt;/ConsumerNotifications&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;EndUserInfo&gt;xs:string&lt;/EndUserInfo&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;/PlannedMaintenance&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 „ComplexType: ResponseType“ ] &lt;ResponseContext&gt; „ComplexType: ResponseContextType“ &lt;/ResponseContext&gt; [1..1] [ END BASE TYPE ] [ START CHOICE ] &lt;Fault&gt; „ComplexType: FaultStateType“ &lt;/Fault&gt; [1..1] &lt;PlannedMaintenance&gt; „ComplexType: PlannedMaintenanceWindowType“ &lt;/PlannedMaintenance&gt; [1..1] [ END CHOICE ] &lt;/...&gt; </pre>

## ComplexType: FeedbackNotificationsType

Abstract	no
XML Instance Representation	<pre> &lt;...&gt; &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 ] &lt;Notifications&gt;„ComplexType: NotificationsType“&lt;/Notifications&gt; [1..1] &lt;/...&gt;</pre>

## ComplexType: NotificationsType

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

## ComplexType: ParagraphAnswerType

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

## ComplexType: ParagraphType

Abstract	no
XML Instance Representation	<pre>&lt;...&gt; &lt;ID&gt;xs:short&lt;/ID&gt; [1..1] &lt;Label&gt;xs:token&lt;/Label&gt; [1..1] [ START CHOICE ] &lt;Value&gt;„ComplexType: ParagraphValueType“&lt;/Value&gt; [0..1] &lt;Answer&gt;„ComplexType: ParagraphAnswerType“&lt;/Answer&gt; [0..1] [ END CHOICE ] &lt;/...&gt;</pre>

## ComplexType: ParagraphValueType

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

## ComplexType: PingConsumerRequestType

Abstract	no
XML Instance Representation	<pre>&lt;...&gt; &lt;UserAgent&gt;„ComplexType: UserAgentType“&lt;/UserAgent&gt; [1..1] &lt;Timestamp&gt;xs:dateTime&lt;/Timestamp&gt; [1..1] &lt;NextCheck&gt;xs:dateTime&lt;/NextCheck&gt; [1..1] &lt;RegisteredMaintenance&gt;„ComplexType: RegisteredMaintenanceType“&lt;/RegisteredMain\ tenance&gt; [0..1] &lt;/...&gt;</pre>

## ComplexType: PingConsumerResponseType

Abstract	no
XML Instance Representation	<pre>&lt;...&gt; &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;/PlannedMaintenan\ ce&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]   &lt;X509Certificate&gt;__ComplexType: X509CertificateType__&lt;/X509Certificate&gt;    [1..unbounded] &lt;/...&gt;</pre>

## ComplexType: ReceivedStoriesType

Abstract	no
XML Instance Representation	<pre>&lt;...&gt;   &lt;StoryID&gt;__SimpleType: IDType__&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;__ComplexType: UserAgentType__&lt;/UserAgent&gt;    [1..1]   &lt;CompanyName&gt;__SimpleType: NotEmptyStringType__&lt;/CompanyName&gt;    [1..1]   &lt;TransmissionDate&gt;xs:dateTime&lt;/TransmissionDate&gt;    [1..1]   &lt;RequestID&gt;__SimpleType: IDType__&lt;/RequestID&gt;    [1..1]   &lt;LanguageCode&gt;__SimpleType: LanguageCodeType__&lt;/LanguageCode&gt;    [1..1]   &lt;MonitoringID&gt;__SimpleType: MonitoringIDType__&lt;/MonitoringID&gt;    [0..1] &lt;/...&gt;</pre>

## ComplexType: RequestType

Abstract	no
XML Instance Representation	<pre>&lt;...&gt;   &lt;RequestContext&gt;__ComplexType: RequestContextType__&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;„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] &lt;/...&gt; </pre>
-----------------------------	---

## ComplexType: ResponseType

Abstract	no
XML Instance Representation	<pre> &lt;...&gt; &lt;ResponseContext&gt;„ComplexType: ResponseContextType“&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 „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] &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; „SimpleType: UID-BFSType“ &lt;/UID&gt; [1..1]   &lt;Unknown&gt; „ComplexType: EmptyType“ &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; „SimpleType: NotEmptyStringType“ &lt;/Producer&gt; [1..1]   &lt;Name&gt; „SimpleType: NotEmptyStringType“ &lt;/Name&gt; [1..1]   &lt;Version&gt; „SimpleType: NotEmptyStringType“ &lt;/Version&gt; [1..1]   &lt;StandardVersion&gt;xs:decimal&lt;/StandardVersion&gt; [1..1]   &lt;Certificate&gt; „SimpleType: NotEmptyStringType“ &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;_SimpleType: EmailAddressType_&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. 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 Validity: erste Stufe; Plausibility: zweite Stufe; Acceptance: dritte Stufe; 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>