



Qualitätssicherung. **Vom Landwirt bis zur Ladentheke.**

Schnittstellenspezifikation **Bündler** Landwirtschaft/Erzeugung





Dokument Versionen					
Bezeichnung	Version	Developer	Autor	Beschreibung der Version	Datum
Nachrichten zwischen Bündlerdatenbanken und zentraler QS-Kontrolldatenbank	1.4.2	SLA	Hermann Grevemeyer, Timm Melchert	Ausländische OGK-Betriebe, die nach EurepGAP zertifiziert wurden, können sich am QS-System anmelden. Um die Informationen der Prüfung an die QS-Datenbank übermitteln zu können, wurden neue Nachrichten definiert.	28.08.2006
	1.4.1	SLA	Hermann Grevemeyer, Timm Melchert	In dieser Version wird der Anredeschlüssel wieder genau definiert.	
	1.3	SLA	Hermann Grevemeyer, Timm Melchert	An einigen Stellen sind Elemente aus den Nachrichten entfernt worden, andere Nachrichten haben zusätzliche Felder erhalten. Zu dieser Beschreibung gibt es eine aktualisierte, passende WSDL-Datei.	
	1.2	SLA	Hermann Grevemeyer, Timm Melchert	Aktualisierung der Schnittstelle zur Kommunikation von Geschäftsnachrichten mit der zentralen QS-Kontrolldatenbank (Ersetzung der asynchronen durch synchrone Nachrichten zu). Änderungen in der Kodierung der Produktionsarten. Aufgrund einer Änderung der Schnittstellen zwischen der QS und Salmonellendatenbank sind die Herdenmeldungen aus diesem Dokument	



Dokument Versionen					
				herausgenommen worden.	
	1.1	SLA	Hermann Grevemeyer, Timm Melchert	Beschreibung weiterer Schnittstellen zur Kommunikation von Geschäftsnachrichten mit der QS-Kontrolldatenbank: Aktualisieren von Daten über Systempartner oder Standorte, die bereits im System vorhanden sind. Beschreibung der Schnittstellen zur Abfrage von Prüfergebnissen und nach QS-Teilnehmern	
Änderung der Bündler-Schnittstelle zum Datenaustausch mit der QS-Software-Plattform	Rev 0.2	International Software Industry Parc GmbH	Thorsten Löffeler, Reza Eslami		15.03.2007
Projekt QS-Plattform Bündlerschnittstelle White Paper	1.3	VBO-Services GmbH	Reza Eslami	Endversion	21.06.2007
	1.4	VBO-Services GmbH	Michael Chodorowski	Ergänzungen in den Funktionen 6.3 und 6.4	03.09.2009
	1.5	VBO-Services GmbH	Oliver Bandke	Ergänzungen in den Funktionen 6.1, 6.3, 6.25, 6.26 und 6.27; neue Funktionen 6.28	15.07.2010
Zusammenführung und Überarbeitung	2.0	VBO-Services GmbH	Oliver Bandke		26.07.2016
Erweiterung um die Nachrichten 6.29 ff		VBO-Services GmbH	Oliver Bandke		16.08.2018



Inhalt

1	Status des Dokuments	6
2	Zielgruppe	6
3	Terminologie	6
4	Übertragung der Nachrichten	10
4.1	Technische Aspekte	10
4.2	Automatisches Generieren von Programmcode	10
4.3	Testen von Soap-Nachrichten	10
5	Authentifizierung	11
5.1	Allgemeines zur Verschlüsselung	12
6	Nachrichten	12
6.1	Meldung eines Systempartners	12
6.1.1	CQSSRegisterMemberMsg - Anmeldung neuer Systempartner	12
6.1.2	CQSSRegisterMemberMsgReply - die Antwortdaten	13
6.1.3	CQSSRegisterMemberMsg – zusätzliche Beschreibung	13
6.2	Meldung eines Standortes	15
6.2.1	CQSSAddLocationMsg - Anmeldung eines neuen Standortes	15
6.2.2	CQSSAddLocationMsgReply - die Antwortdaten	15
6.2.3	CQSSAddLocationMsg – zusätzliche Beschreibung	16
6.3	Erweiterte Auskunft über QS-Teilnahme eines Standortes	16
6.3.1	QSExtLocationQuery - Erweiterte Abfrage von Standortdaten	16
6.3.3	QSExtLocationQueryReply - die Antwortdaten	17
6.3.4	QSExtLocationQuery - zusätzliche Beschreibung	17
6.4	Erweiterte Auskunft über QS-Teilnahme und Produktionsarten eines Standortes	17
6.4.2	QSExtLocationBtartQuery - Erweiterte Abfrage von Produktionsart und Standortdaten	18
6.4.3	QSExtLocationBtartQueryReply - die Antwortdaten	18
6.4.4	QSExtLocationBtartQueryReply - zusätzliche Beschreibung	19
6.5	Auskunft über Zusammensetzung eines Bündels	19
6.5.1	QSBdlRequestMsg - Informationen zu einzelnen Bündeln	19
6.5.2	QSBdlRequestMsgReply - die Antwortdaten	19
6.6	Adressauskunft über einen gebündelten Betrieb	20
6.6.1	QSBdlStortInfoMsg - Adressinformation zu Standorten	20
6.6.2	QSBdlStortInfoMsgReply - die Antwortdaten	20
6.7	Ummelden von Standortkennungen	21
6.7.1	QSUpdateLocationMsg - Ummelden von Standortkennungen	21
6.7.2	CQSSUpdateMsgReply - die Antwortdaten	22
6.8	Ummelden von Produktionsarten	22
6.8.1	QSUpdateBtartMsg - Ummelden von Produktionsarten	22
6.8.2	CQSSUpdateMsgReply - die Antwortdaten	22
6.9	Ummelden von Prüfinstituten	23
6.9.1	QSUpdateCertBodyMsg - Ummelden von Prüfinstituten	23
6.9.2	CQSSUpdateMsgReply - die Antwortdaten	23
6.10	Änderung von Adresdaten eines Standortes	23
6.10.1	CQSSUpdateLocAdrMsg - Änderung von Adresdaten eines Standortes	23
6.10.2	CQSSUpdateMsgReply - die Antwortdaten	24
6.11	Melden von Änderungen in den Kommunikationsdaten	24
6.11.1	CQSSUpdateLocComMsg - Änderung von Kommunikationsdaten eines Standortes	24
6.11.2	CQSSUpdateMsgReply - die Antwortdaten	25

6.12	Abmelden von Systempartnern.....	25
6.12.1	QSDeleteMemberMsg - Abmelden von allen Standorten eines Systempartners.....	25
6.12.2	CQSSUpdateMsgReply - die Antwortdaten.....	25
6.13	Abmelden von Standorten	26
6.13.1	QSDeleteLocationMsg - Abmelden von Standorten	26
6.13.2	CQSSUpdateMsgReply - die Antwortdaten.....	26
6.14	Abfragen der aktuellsten Prüfberichte	26
6.14.1	QSLastInspectionMsg - Abfragen der aktuellsten Prüfberichte.....	27
6.14.2	QSLastInspectionMsgReply - die Antwortdaten	27
6.15	Abfragen der zuletzt verschickten Prüfberichte	28
6.15.1	QSLastSentInspectionMsg - Abfragen der zuletzt verschickten Prüfberichte.....	28
6.15.2	QSLastInspectionMsgReply - die Antwortdaten	28
6.16	Erweiterte Abfrage von Prüflisten eines Standortes	28
6.16.1	QSExtInspectionMsg - Abfrage von Prüfergebnissen	28
6.16.2	QSExtInspectionMsgReply - die Antwortdaten	29
6.17	Abfragen von Prüfberichten.....	32
6.17.1	QSInspectionFromIdMsg - Abfragen von Prüfberichten	32
6.17.2	QSExtInspectionMsgReply - die Antwortdaten	32
6.18	Erweiterte Abfrage von Prüflisten eines Standortes in einem Zeitraum.....	32
6.18.1	QSExtInspectionsInPeriodMsg - Erweiterte Abfrage aller Prüfergebnisse eines bestimmten Zeitraums	32
6.18.2	QSExtInspectionsInPeriodMsgReply - die Antwortdaten	33
6.19	Abfragen von Standortinformationen	33
6.19.1	QSLocationInfoMsg - Abfragen von Standortinformationen.....	33
6.19.2	QSLocationInfoMsgReply - die Antwortdaten	33
6.20	Erstellen von Bündeln.....	34
6.20.1	QSBuildBundleMsg - Erstellen von Bündeln	34
6.20.2	QSBuildBundleMsgReply - die Antwortdaten	35
6.20.3	QSBuildBundleMsg - zusätzliche Beschreibung.....	35
6.21	Änderung von Adressdaten eines Unternehmens.....	35
6.21.1	CQSSUpdateCompAdrMsg - Änderung von Adressdaten eines Unternehmens	35
6.21.2	CQSSUpdateMsgReply - die Antwortdaten.....	36
6.22	Melden von Änderungen in den Kommunikationsdaten eines Unternehmens	36
6.22.1	CQSSUpdateCompComMsg - Änderung von Kommunikationsdaten eines Unternehmens.....	36
6.22.2	CQSSUpdateMsgReply - die Antwortdaten.....	37
6.23	Ändern von Standortdaten.....	37
6.23.1	QSChangeLocationDataMsg - Ändern von Standortdaten	37
6.23.2	CQSSUpdateMsgReply - die Antwortdaten.....	37
6.23.3	CQSSUpdateMsg - zusätzliche Beschreibung	38
6.24	Abfrage des aktuellen Prüfinstituts	38
6.24.1	QSCertBodyQueryMsg - Abfrage des aktuellen Prüfinstituts.....	38
6.24.2	QSCertBodyQueryMsgReply - die Antwortdaten	38
6.25	Meldung eines Systempartners mit anerkanntem Fremdsystem.....	38
6.25.1	CQSSRegisterCertificationSystemMsg - Anmeldung neuer anerkannter Systempartner	39
6.25.2	CQSSRegisterMemberMsgReply - die Antwortdaten.....	40
6.25.3	CQSSRegisterMemberMsg - Nachrichtenbeschreibung	40
6.26	Meldung eines Standortes mit anerkanntem Fremdsystem.....	41
6.26.1	CQSSAddCertificationSystemMsg - Anmeldung eines neuen Standortes	41
6.26.2	CQSSAddLocationMsgReply - die Antwortdaten.....	42
6.26.3	CQSSAddLocationMsg - Nachrichtenbeschreibung	42
6.27	Änderung von Standortdaten (nicht mehr Unterstützt)	43
6.27.1	QSChangeAttributeMsg - Änderung von Standortdaten	43
6.27.2	CQSSUpdateMsgReply - die Antwortdaten.....	43
6.27.3	QSChangeAttributeMsg - Nachrichtenbeschreibung	43
6.28	Änderung des Zertifizierungssystems.....	43
6.28.1	QSUpdateCertificationSystemMsg - Änderung des Zertifizierungssystems	43
6.28.2	CQSSUpdateMsgReply - die Antwortdaten.....	44
6.28.3	QSUpdateCertificationSystemMsg - Nachrichtenbeschreibung	44
6.29	Standort an Initiative Tierwohl übertragen	45
6.29.1	QSRegisterTierwohlRequest – Übertragung eines Standortes.....	45



6.29.2	registerTierwohlResponse - die Antwortdaten	45
6.29.3	QSRegisterTierwohlRequest - Nachrichtenbeschreibung	45
6.30	Ein Login für eine landwirtschaftliches Unternehmen erstellen	46
6.30.1	requestLoginRequest – Anfrage für ein Logintoken	46
6.30.2	requestLoginResponse - die Antwortdaten	46
6.30.3	requestLoginRequest - Nachrichtenbeschreibung	46
6.31	Ein Login für eine landwirtschaftliches Unternehmen erstellen und versenden	46
6.31.1	sendLoginRequest– Versand für ein Logintoken	47
6.31.2	sendLoginResponse - die Antwortdaten	47
6.31.3	sendLoginRequest - Nachrichtenbeschreibung	47
6.32	Die Liste aller landwirtschaftlichen Unternehmen ohne Login abfragen.....	47
6.32.1	noLoginRequest – Abfrage der List	47
6.32.2	noLoginResponse - die Antwortdaten.....	47
6.32.3	noLoginRequest - Nachrichtenbeschreibung	48
A.	Fehlermeldungen	48
7	Literatur.....	49
8	WSDL	50

1 Status des Dokuments

Freigabe

2 Zielgruppe

Zielgruppe dieses Dokuments sind Personen, die die hier dargelegten Schnittstellen für eigenständige Bündlerdatenbanken softwaretechnisch umsetzen bzw. realisieren wollen. Der Text enthält daher keine systematische Einführung in Verschlüsselungstechniken, Webservices, XML und SOAP sondern setzt diese Kenntnisse voraus, bzw. die Fertigkeit, sich gegebenenfalls geeignete Literatur zu beschaffen. Nützliche Bücher dafür sind beispielsweise [1], [2], [3]. Darüber hinaus finden sich im Internet viele grundsätzliche und lesenswerte Beiträge zu diesen Bereichen (z. B. bei [7], [6] oder [5]).

Ziel des Dokuments kann es auch nicht sein, umfassende Informationen zum QS-System an sich und seiner Funktionsweise zu liefern. Aus diesem Grund empfehlen sich hier eine Kooperation der jeweiligen Bündler mit ihren IT-Dienstleistern und ein Besuch auf der QS-Homepage.

Das Dokument wendet sich nicht an Bündler, die eine von einem IT-Dienstleister angebotene IT-Datenbank verwenden.

3 Terminologie

Im Folgenden werden einige Begriffe näher definiert, die für das Verständnis des QS-Systems und der Schnittstellen notwendig sind.

Teilnehmerkennung: Eindeutige Kennung eines Systempartners.

Standort: Ein Standort ist innerhalb des QS-Systems eine logische und keine geografische Größe und kennzeichnet eine logische Produktionseinheit. Standorte sind Gegenstand der QS-Kontrollen. Folglich beziehen sich Prüflisten immer auf Standorte. Ein Standort ist definiert als Kombination einer Identifikationsnummer und einer QS-Produktionsart (siehe QS-Produktionsarten).

QS-Stufen: QS unterscheidet mehrere Kategorien von Systempartnern, die im Datentyp CS840021Type verkapselt sind. In Tabelle 1 sind die definierten Stufen mit den entsprechenden Schlüsselwerten aufgeführt.



Tabelle 1: QS-Stufen (CS840021Type)

Schlüssel	Stufen
2	Futtermittelwirtschaft
3	Landwirt
4	landwirtschaftlicher Bündler
5	Schlacht-/Zerlegebetrieb Rotfleisch
6	Verarbeitungsbetrieb
7	Lebensmitteleinzelhandel
8	LEH Bündler
10	Schlacht-/Zerlegebetrieb Weißfleisch
11	Großhandel
12	Systemkoordinator
13	Fleischerhandwerk
14	Bündler Fleischerhandwerk
15	Kleinstherzeuger
16	Fahrbare Mahl- und Mischanlagen
22	Logistik
23	Landwirtschaft Brüterei

Systempartner: Ein Systempartner ist eine abstrakte Größe, der ein oder mehrere Standorte zugeordnet sind. Ein Systempartner als solches wird als Verwaltungseinheit betrachtet und ist nicht Gegenstand von QS-Kontrollen. Ein Systempartner trägt aber die Verantwortung für die bewirtschafteten Standorte gegenüber QS und ist somit Ansprechpartner im Falle von Sanktionen oder nicht bestandener Audits.

QS-Produktionsarten: Eine Produktionsart bezieht sich auf einen Standort und charakterisiert die dort betriebene Wirtschaftsform. Die Produktionsart wird durch den Datentyp CS840018Type repräsentiert. In Tabelle 2 findet sich eine Aufstellung der definierten Produktionsarten. Seit dem 23.6.2003 können die Produktionsarten landwirtschaftlicher Standorte kombiniert und entsprechend repräsentiert werden. Dies ergibt sich durch bitweise Addition der letzten drei Kennzahlen einer Produktionsartennummer. So werden aus 1001 und 1002 1003. Dies hat für das Hinzumelden von Standorten Konsequenzen. Wenn ein Standort bereits mit der Produktionsart 1012 in der Datenbank vorhanden ist, und eine Standorterweiterung mit 1004 gemeldet werden soll, wird eine Fehlermeldung erzeugt, da 1004 in 1012 (= 1008+1004) bereits enthalten ist.

Tabelle 2: QS-Produktionsarten Teil 1 (CS840018Type)

Schlüssel	Produktionsart
20	Bündler Landwirtschaft/Erzeugung
30	Schlachtung/Zerlegung Rotfleisch
31	Schlachtung Rotfleisch
32	Zerlegung Rotfleisch
33	Schlachtung/Zerlegung Weißfleisch
34	Schlachtung Weißfleisch
35	Zerlegung Weißfleisch
36	Transport Rind/Schwein eigene Fahrzeuge
37	Transport Geflügel eigene Fahrzeuge



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



38	Transport Rind/Schwein 1:1 vertraglich gebundener Spediteur
39	Transport Geflügel 1:1 vertraglich gebundener Spediteur
40	Verarbeitung
41	Zerlegung
50	Brütereier Hähnchen-Mastküken
51	Brütereier Puten-Mastküken
52	Transport Eintagsküken eigene Fahrzeuge
53	Transport Eintagsküken 1:1 vertraglich gebundener Spediteur
60	Bündler Lebensmitteleinzelhandel
61	Zentrallager (Fleisch und Fleischwaren)
70	Zusatzstoff-/Vormischungsherstellung
71	Mischfutterherstellung
72	Einzelfutterherstellung
73	Kleinsterzeuger (Einzelfuttermittelherstellung)
74	Private Labelling
75	Fahrbare Mahl- und Mischanlagen
76	Handel (Futtermittel)
77	Lagerung und Umschlag (Futtermittel)
78	Straßentransport (Futtermittel)
79	Schienen-, Binnen-, Seeschifftransport (Futtermittel)
80	Fleischgroßhandel
81	Großhandel Obst, Gemüse, Kartoffeln (Ersterfasser)
82	Großhandel Obst, Gemüse, Kartoffeln (Handelspartner)
83	Bearbeitung Obst, Gemüse, Kartoffeln
84	Logistik Obst, Gemüse, Kartoffeln
86	Lebensmitteleinzelhandelslager Fleisch und Obst, Gemüse, Kartoffeln
90	Bündler Fleischerhandwerk
91	Fleischerhandwerk: Produktion ohne Verkaufsstelle/n
92	Fleischerhandwerk: Produktion mit direkter Verkaufsstelle
93	Fleischerhandwerk: nur Verkaufsstellen
301	Elterntierhaltung für Hähnchen
304	Elterntierhaltung für Puten
450	Bearbeitung
460	Ausgegliederte Vermarktung
470	Erzeuger mit nicht selbst erzeugter Ware
701	Mischfutterherstellung (spezielle Prozesse)
702	Einzelfuttermittelherstellung (spezielle Prozesse)
703	Kleinsterzeuger von Einzelfuttermitteln (spezielle Prozesse)
801	Agenturen Obst, Gemüse, Kartoffeln (Ersterfasser)
802	Agenturen Obst, Gemüse, Kartoffeln (Handelspartner)
840	Bündler Logistik Obst, Gemüse, Kartoffeln
880	Broker (Fleisch und Fleischwaren)
904	Tiertransport (Gewerbliches Transportunternehmen)
1001	Rindermast
1002	Kälbermast
1004	Fresser-/Kälberaufzucht



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



1008	Milchviehhaltung und Kälberaufzucht
1016	Mutter-/Ammenkuhhaltung mit Kälbern
1320	Schlachtkühe (QM-Milch)
2001	Schweinemast
2002	Jungsauen-/Eberaufzucht
2004	Sauenhaltung und Ferkel bis zum Absetzen
2008	Ferkelaufzucht
3001	Hähnchenmast
3002	Putenaufzucht
3004	Putenmast
3008	Pekingentenaufzucht
3016	Pekingentenmast
3032	Legehennenhaltung
4001	Obstanbau (Freiland)
4002	Obstanbau (geschützt)
4004	Gemüseanbau (Freiland)
4008	Gemüseanbau (geschützt)
5001	Kartoffelanbau
6001	Lebensmitteleinzelhandel Fleisch und Fleischwaren
6002	Lebensmitteleinzelhandel Obst, Gemüse, Kartoffeln
6003	Lebensmitteleinzelhandel Fleisch und Obst, Gemüse und Kartoffeln
6102	Lebensmitteleinzelhandel Obst, Gemüse, Kartoffeln + Bearbeitung
6103	Lebensmitteleinzelhandel Fleisch und Obst, Gemüse und Kartoffeln + Bearbeitung
7001	Getreideanbau
7002	Ölsaatenanbau
7004	Hülsenfrüchteanbau
7008	Zuckerrübenanbau
7016	Maisanbau
7032	Kartoffelanbau (ohne Speisekartoffeln)
7064	Grünlandnutzung und Feldfutterproduktion

QS-Status: Ein QS-Betrieb kann, teilweise abhängig von der Stufe unterschiedliche Status annehmen. Der QS-Status wird durch den Datentyp CS80004 repräsentiert und ist in der Tabelle 4 definiert.

Tabelle 4: QS-Status (CS80004Type)

Schlüssel	QS-Status
1	Status I
2	Status II
3	Status III
4	Status IV
5	QS-Anwärter
6	Bündel-Mitglied
7	Bestanden
9	Zugelassen KE Stufe 2
10	Lieferberechtigt



11	IKB-Zertifiziert
13	Nicht bestanden
14	QM-Milch auditiert
15	GLOBALGAP Zertifiziert
16	IKKB Zertifiziert
17	Codiplan+ Zertifiziert
18	DKHV Mitglied

4 Übertragung der Nachrichten

Der Datenaustausch findet synchron statt. Die Daten werden mit Client- und serverseitigen Zertifikaten verschlüsselt übertragen. Die Adresse des Webservices lautet https://www.qs-plattform.de/axis/services/rpcrouter_secure

4.1 Technische Aspekte

SOAP ist eine XML-Anwendung zur Kommunikation von Daten. SOAP wird sowohl für *Remote Procedure Calls* (RPC) im Rahmen von *distributed computing* als auch im Bereich von asynchronem Nachrichtenaustausch eingesetzt. Eine SOAP-Nachricht besteht aus einem *envelope*, der aus *header* und *body* besteht. Im *header* stehen verschiedene Verarbeitungsinformationen, im *body* die eigentliche Nachricht, die im Wesentlichen einfach ein XML-Dokument darstellt.

4.2 Automatisches Generieren von Programmcode

Es sind verschiedene Werkzeuge über das Internet frei verfügbar, mit denen aus WSDL-Dokumenten Programmcodes für SOAP-Clients erzeugt werden können, mit denen wiederum die definierten Schnittstellen angesprochen werden können.

4.3 Testen von Soap-Nachrichten

Nach Fertigstellung des Clients gibt es erfahrungsgemäß Schwierigkeiten, das Kommunikationsverhalten von Client und Server nachzuvollziehen. An dieser Stelle bietet sich ein Werkzeug aus dem AXIS-Projekt an, mit dem es möglich ist, die Anfragen des Clients und Antworten des Servers durch das „Tunneln“ der Anfrage zu visualisieren. Der Tunnel wird mit drei Parametern gestartet: lokaler Port, externer Rechner, externer Port. Der Start eines Tunnels auf dem lokalen Rechner zum QS-Datenbankservers kann beispielsweise folgendermaßen aussehen:

```
java org.apache.axis.utils.tcpmon 44444 193.28.96.81 80
```

Hierzu wird die Anfrage des Clients nicht an den Server direkt, sondern an einen Proxy geschickt, der die Anfrage an den Server weiterleitet. In Abbildung 1 ist ein solcher Tunnel abgebildet.

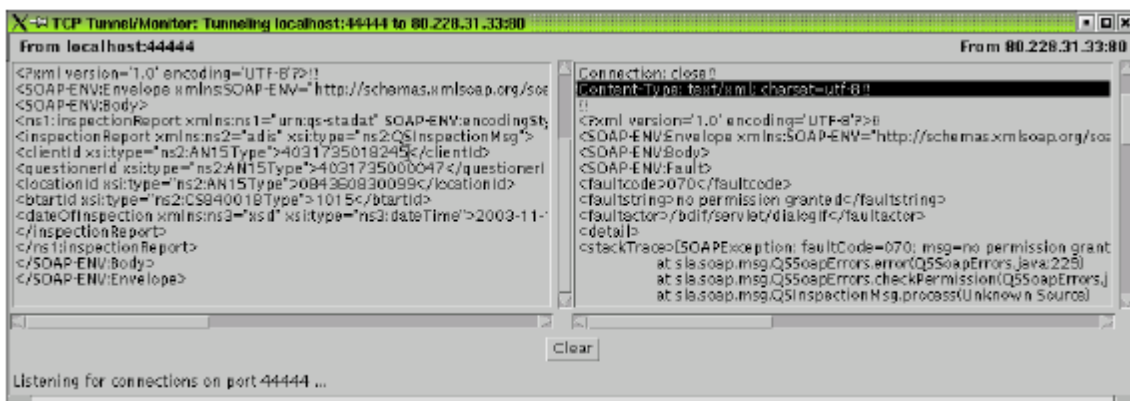


Abbildung 1: Visualisierung einer SOAP-Nachricht



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



5 Authentifizierung

Um zu gewährleisten, dass nur Berechtigte die entsprechenden Antwortnachrichten erhalten, wird die Authentifizierung des HTTP-Protokolls genutzt, d. h. beim Versenden einer SOAP-Nachricht müssen Benutzername und Passwort mitgeschickt werden, welche dem Anfragenden von der Online-Anmeldung bei der QS-Kontrolldatenbank bekannt ist. Vor der Verarbeitung der Nachricht überprüft dann der Server, ob entsprechende Berechtigungen in der Datenbank vergeben sind.

Die Übermittlung wird über das HTTP Protokoll realisiert (Basic Authentication)

```
AxisFault
faultCode: {http://xml.apache.org/axis/}HTTP
faultSubcode:
faultString: (401)Unauthorized
faultActor:
faultNode:
faultDetail:
{:return code: 401
```



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



Des Weiteren wird immer am Anfang der Funktionen überprüft, ob die in der Nachricht übermittelte Bündler-ID/Questioner-ID auch tatsächlich mit dem Bündler übereinstimmt, dessen Login verwendet worden ist. Ist dies nicht der Fall, wird die Exception 70 = „No permission granted“ geworfen (dies geschieht ebenfalls, wenn diese ID leer bzw. nicht übermittelt worden sein sollte). Dieselbe Ausnahme 70 wird auch immer dann geworfen, wenn versucht wird auf einen Standort, Systempartner oder Auditbericht zuzugreifen, der aktuell nicht diesem Bündler zugeordnet ist.

5.1 Allgemeines zur Verschlüsselung

Es muss das HTTPS Protokoll verwendet werden, um auf alle Funktionen der Bündler-Schnittstelle zugreifen zu können. (Diese Lösung ist gänzlich unabhängig von der technischen Implementierung des Webservices und basiert auf einer standardisierten, technischen Umsetzung der Verschlüsselung auf Basis eines Apache Web Servers. Etwaige, diesbezügliche Probleme sind sehr schwer von unserer Seite zu lösen und sind wahrscheinlich auf die Konfiguration und Einstellung des jeweiligen Clients zurückzuführen, damit dieser über einen HTTPS-Tunnel kommunizieren kann.)

6 Nachrichten

6.1 Meldung eines Systempartners

Vorbedingung: Der Bündler ist dem System bekannt.

Nachbedingung: Der Systempartner bekommt eine Mitgliedsnummer und einen vorläufigen Status mitgeteilt. Er ist daraufhin dem System bekannt. Im Fehlerfall wird eine entsprechende Fehlermeldung generiert und an den meldenden Teilnehmer zurückgeschickt.

6.1.1 CQSSRegisterMemberMsg - Anmeldung neuer Systempartner

Der Bündler überträgt hierzu die Stammdaten eines Systempartners sowie eines bewirtschafteten Standortes.

```
<complexType name="CQSSRegisterMemberMsg">
  <sequence>
    <element name="address" nillable="true" type="tns3:CQSSAddressType"/>
    <element name="location" nillable="true" type="tns3:CQSSLocationType"/>
    <element name="com" nillable="true" type="tns3:CQSSCommunicationType"/>
    <element name="bundlerId" nillable="true" type="tns3:AN15Type"/>
    <element name="representativeAddress" nillable="true"
      type="tns3:CQSSAddressType"/>
    <element name="groupFlag" nillable="true" type="tns3:AN99Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

address: Adresse des Systempartners.

location: Standortdaten.

com: Kommunikationseinrichtungen mit denen der Systempartner kontaktiert werden kann.

bundlerId: QS-Identifikation des Bündlers.



representativeAddress: Adresse des Ansprechpartners bzw. Verantwortlichen. Wenn Adresse des Systempartners und des Ansprechpartners identisch sind, muss hier keine weitere Adresse angegeben werden.

6.1.2 CQSSRegisterMemberMsgReply - die Antwortdaten

```
<complexType name="CQSSRegisterMemberMsgReply">  
  <sequence>  
    <element name="timeStamp" nillable="true" type="xsd:dateTime"/>  
    <element name="qsState" nillable="true" type="tns3:CS80004Type"/>  
    <element name="clientId" nillable="true" type="tns3:AN15Type"/>  
  </sequence>  
</complexType>
```

Die XML-Elemente haben folgende Semantik:

timeStamp: Zeitstempel.

qsState: QS-Status (in der Regel „5“ für noch nicht auditierten Systempartner).

clientId: Teilnehmerkennung.

6.1.3 CQSSRegisterMemberMsg – zusätzliche Beschreibung

Als erstes wird überprüft, ob unter den übermittelten Daten für LocationNumber und LocationType ein Standort existiert. Wird ein solcher Standort gefunden, erfolgt an dieser Stelle (abweichend vom sonst üblichen Procedere) KEINE Überprüfung, ob dieser Standort dem Bündler gehört (da ein Bündler auf diesem Wege einen abgemeldeten, fremden Standort für sich wieder anmelden kann) und ob der Standort abgemeldet ist.

Sollte er abgemeldet sein, wird überprüft, ob es zu einer Produktionsartenkollision kommen würde, wenn dieser Standort angemeldet wird (was zu einer „023“, "Data collision: location address is already in database" führen würde, da zur selben Zeit nicht sich überschneidende Produktionsarten angemeldet sein dürfen).

Jetzt wird für diesen abgemeldeten Standort überprüft, ob er bereits dem Bündler zugeordnet ist oder bisher einem anderen Bündler zugeordnet war. Sollte er bei einem anderen Bündler gewesen sein, so wird seine Zugehörigkeit zu dem bisherigen Bündler beendet und er wird dem neuen Bündler zugeordnet und angemeldet. Sollte der Standort aber dem Bündler bereits zugeordnet sein, so wird auf diesem Standort die Transition „Anmelden“ durchgeführt, um den Standort wieder in den Zustand „angemeldet“ zu bringen. Sollte dies nicht möglich sein, wird mit 100, "Error: please contact support@qs-platform.info geantwortet. In diesem Fall wird kein Update der Daten durchgeführt oder Werte verändert, es erfolgt nur die Anmeldung des Standortes.

Sollte der Standort angemeldet gewesen sein, geschieht gar nichts und es werden nur der aktuelle Status sowie die Client-ID zurückgegeben.

Falls der Standort NICHT gefunden werden konnte, so muss er zusammen mit dem Systempartner/ Unternehmen neu angelegt werden. Dazu wird aus den Adressdaten ein neuer Systempartner angelegt. Dabei muss aber PostCode, Town, Street und Lastname mit Text gefüllt sein, da dies Pflichtfelder sind.



Qualitätssicherung. **Vom Landwirt bis zur Ladentheke.**



Falls eines der Felder nicht ausgefüllt wird, wird die Fehlermeldung 003, "Address information is ambiguous" geworfen. Wird das Land nicht angegeben, wird mit 058, "No country code given" reagiert. Der Name wird aus dem locationAddress.Name bestimmt. Ein etwaiger, dort angegebener FirstName wird mit einem führenden Leerzeichen an den LastNamen hinten angehängt, um so den Namen des Unternehmens festzulegen (dies geschieht aus Konformitätsgründen zur früheren Anwendung). Für diesen Systempartner wird dann eine QS-ID generiert, welche auch am Ende zurückgegeben wird. Da das Feld Bundesland nun ebenfalls als ein Pflichtfeld deklariert worden ist, wenn als Land Deutschland ausgewählt wurde, dieses Bundesland aber nicht direkt in der Schnittstelle übertragen werden kann, wird in einem ersten Schritt hier vom System das Bundesland „Deutschland“ eingetragen. (Später wird dann aus der Postleitzahl das Bundesland errechnet bzw. es muss dann irgendwann über die Schnittstelle mitgeliefert werden).

Die beim Systempartner zu setzende QS-Stufe ergibt sich zwingend aus der jeweiligen QS-Stufe des bündelnden Bündlers. D.h. beim Bündler Landwirtschaft ergibt sich Stufe „3“ (Landwirt).

Falls eine RepresentativAddress vergeben worden ist, wird diese für den Anwender verwendet, der dem Systempartner als gesetzlicher Vertreter zugeordnet werden soll. Ansonsten wird die Adresse des Systempartners darauf übertragen. Falls auch diese leer sein sollte, wird auf die Adresse ausgewichen, die vielleicht in der Location übermittelt worden ist.

Es wird für diesen Systempartner nun ein neuer Standort angelegt.

Da die Zertifizierungsstelle ein Pflichtfeld ist, muss deren QS-ID übermittelt und diese Zertifizierungsstelle im System auch so gefunden werden: ansonsten wird ein 043, "No such certification body" Fehler erzeugt und zurückgegeben.

Ebenfalls zwingend notwendig ist bei einem Standort der locationType, wobei nur die locationType ausgewählt werden dürfen, die für die aktuelle QS-Stufe gültig sind, sonst wird eine Exception des Typs 033, "No such production line" geworfen.

Wie es bereits beim Systempartner beschrieben worden ist, sind auch im Standort die Felder PostCode, Town, Street und Lastname Pflichtfelder und müssen übermittelt werden, ansonsten wird eine entsprechende Exception versendet. Auch hier wird aus dem übermittelten Last und FirstName ein Standortname erzeugt.

Auch muss immer eine LocationNumber übergeben werden (sollte sie nicht übermittelt worden sein, wird eine 057, "No location number given" geworfen). Es wird dann überprüft, ob die Kombination aus locationType und LocationNumber sich nicht mit bestehenden Betriebsarten überschneidet, die für diese Standortnummer bereits angemeldet sind. Sollte bereits eine der Grundarten angemeldet sein, so wird die Exception 023, "Data collision: location address is already in database" geworfen (aus Kompatibilitätsgründen zur alten Version des WebServices wurde diese Exception gewählt).

Auch hier wird wie für den Systempartner das Bundesland mit Deutschland automatisch vorbelegt, wenn als Land Deutschland ausgewählt wurde.

Falls in dem Feld location ein Representativ mitgesendet wurde, werden seine Daten verwendet, um den Ansprechpartner des Standorts damit zu befüllen.

Bzgl. der Kommunikationsdaten (Fax, E-Mail, Telefon und Mobil) werden keine Exceptions geworfen, da sie nicht verpflichtend sind. An dieser Stelle sei nur angemerkt, dass an den Systempartner die Daten aus dem obersten Com Element angefügt werden und falls dieses leer ist, die Daten aus dem Com Element der location. An den Anwender des Standorts werden nur die Daten des location.com Elements gepackt.



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



Sollte als Standortnummer eine Nummer gewählt werden, die mit „528528“ beginnt, so bedeutet dies, dass der Bündler den Standort für das System IKB anmelden möchte. Dies ist aber nur möglich, wenn auch die gewählte Produktionsart im 2000er Bereich liegt. Ansonsten wird die Exception „033“, "No such production line" geworfen und es erfolgt keine Anmeldung, da IKB nur für dieses System gültig ist.

Falls beim Speichern des Systempartners oder des Standorts festgestellt wird, dass dessen Adresse bereits in der DB enthalten ist (also eine Dublette der Adresse dann vorliegen würde), so wird eine Exception 023, "Data collision: location address is already in database" geworfen.

Als Rückgabewert werden das aktuelle Datum und die generierte ID des Systempartners angefügt (also nicht die Standortnummer).

6.2 Meldung eines Standortes

Vorbedingung: Der Bündler und der Systempartner sind dem System bekannt.

Nachbedingung: Der neue Standort ist im System bekannt und Audits können für diesen Standort erfasst werden. Im Fehlerfall wird eine entsprechende Fehlermeldung zurückgeschickt.

6.2.1 CQSSAddLocationMsg - Anmeldung eines neuen Standortes

Ein Standort ist an dieser Stelle kein geografischer Begriff, sondern eine logische Größe, die durch eine Standortnummer (z. B. VVVO-Nummer), die Kennung des Systempartners und eine Produktionsart eindeutig bestimmt ist.

```
<complexType name="CQSSAddLocationMsg">
  <sequence>
    <element name="clientId" nillable="true" type="tns3:AN15Type"/>
    <element name="location" nillable="true" type="tns3:CQSSLocationType"/>
    <element name="bundlerId" nillable="true" type="tns3:AN15Type"/>
    <element name="groupFlag" nillable="true" type="tns3:AN99Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

clientId: Systemidentifikation des Teilnehmers, von dem ein Standort gemeldet wird.

location: nachgemeldeter Standort des Systempartners.

bundlerId: Systemidentifikation des Bündlers.

6.2.2 CQSSAddLocationMsgReply - die Antwortdaten

```
<complexType name="CQSSAddLocationMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:dateTime"/>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="qsState" nillable="true" type="tns3:CS80004Type"/>
    <element name="clientId" nillable="true" type="tns3:AN15Type"/>
  </sequence>
</complexType>
```



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



Die XML-Elemente haben folgende Semantik:

timeStamp: Zeitstempel.

locationId: Standortidentifikationsnummer (Landwirtschaft: VVVO-Nummer und ISO Länderzeichen (276 für Deutschland)).

qsState: QS-Status des neuen Standortes (in der Regel „5“ für noch nicht auditierten Systemteilnehmer).

clientId: Systemidentifikation des Teilnehmers, von dem ein Standort gemeldet wird.

6.2.3 CQSSAddLocationMsg – zusätzliche Beschreibung

Bei dieser Funktion muss der Systempartner bereits angelegt worden sein (also im System existieren), ansonsten wird eine Fehlermeldung 031", "No such participants" generiert.

Auch bei der Suche nach dem Standort wird dieser gefundene Systempartner mit einbezogen und nur seine Standorte können, wie oben beschrieben, auf diesem Wege wieder angemeldet werden. (Da hier die Funktionalität ausgelagert worden ist, kann also auch auf diesem Wege ein Bündlerwechsel und eine EurepGap- bzw. QS-Ummeldung erfolgen, wie es oben beschrieben worden ist).

Falls kein Standort angemeldet, sondern diesem Systempartner ein neuer Standort hinzugefügt werden soll, so verhält sich die Anwendung von hier an wie bereits unter 6.1 beschrieben (wobei natürlich kein Systempartner und kein Anwender für diesen Systempartner erzeugt werden).

Bei der Rückantwort an den Aufrufenden werden sowohl der aktuelle Status des Standorts, in dem sich dieser nun befindet, seine Standortnummer, als auch die ID des Systempartners mit dem Timestamp zurückgeliefert.

6.3 Erweiterte Auskunft über QS-Teilnahme eines Standortes

Vorbedingung: Jemand fragt, ob für einen Standort ein gültiger Systemvertrag vorliegt. Produktionsart und Standortkennung des abgefragten Standortes sind bekannt.

Nachbedingung: Status des Standortes ist bekannt. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.

6.3.1 QSExtLocationQuery - Erweiterte Abfrage von Standortdaten

```
<complexType name="QSExtLocationQuery">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="bstartId" nillable="true" type="tns3:CS840018Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

locationId: Die Standortnummer des Standortes.

bstartId: Der Produktionsartenschlüssel des Standortes nach Schlüsseltable.



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



6.3.3 QSExtLocationQueryReply - die Antwortdaten

Die Antwortnachricht enthält einen Schlüsselwert, der anzeigt, ob für den Standort ein gültiger Systemvertrag vorliegt (1) oder nicht (0). Bei inaktiven Standorten (geperrt wegen fehlender Salmonellenproben) wird 2 zurückgeliefert. Bei Standorten, die aus anderen Gründen gesperrt sind, wird 3 zurückgegeben. Des Weiteren wird die Antwort der Nachricht in Klartext zurückgeliefert. Bei zugelassenen Standorten „yes“, bei nicht zugelassenen Standorten „no“, bei inaktiven Standorten (gesperrt wegen fehlender Salmonellenproben) „inactive“ und bei gesperrten Standorten „locked“. In der Antwortnachricht wird zusätzlich das Datum, bis wann dieser Standort zugelassen ist `accreditedUntil`, übertragen.

```
<complexType name="QSExtLocationQueryReply">
  <sequence>
    <element name="certifiedLocation" type="xsd:int"/>
    <element name="description" nillable="true" type="tns3:AN99Type"/>
    <element name="accreditedUntil" type="xsd:dateTime"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

certifiedLocation: Schlüsselwert, der anzeigt ob für einen Standort ein gültiger Systemvertrag vorliegt oder nicht. Schlüssel: 0 = no, 1 = yes, 2 = inactive, 3 = locked

description: Beschreibung des Schlüsselwertes.

accreditedUntil: Datum, bis wann dieser Standort zugelassen ist.

6.3.4 QSExtLocationQuery - zusätzliche Beschreibung

Der Parameter „description“ in der Antwort kann noch zusätzliche Informationen enthalten.

- FIAS, für ein durchgeführtes FIAS-Audit
- FB, falls es sich um einen Familienbetrieb handelt.
- REGIO_OK, für ein bestandenes Regionalfenster-Audit
- REGIO_NOK, falls noch kein Regionalfenster-Audit durchgeführt wurde oder dieses nicht bestanden wurde
- VLOG, falls eine Teilnahme an VLOG gewünscht ist

6.4 Erweiterte Auskunft über QS-Teilnahme und Produktionsarten eines Standortes

Vorbedingung: Jemand fragt, ob für einen Standort bezüglich einer Tierart (Schwein, Rind, Geflügel) ein gültiger Systemvertrag vorliegt. Tierart und Standortkennung des abgefragten Standortes sind bekannt.

Nachbedingung: Status und Produktionsart des Standortes ist bekannt. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



6.4.2 QSExtLocationBtartQuery - Erweiterte Abfrage von Produktionsart und Standortdaten

```
<complexType name="QSExtLocationBtartQuery">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="btartId" nillable="true" type="tns3:CS840018Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

locationId: Die Standortnummer des Standortes.

btartId: Der Produktionsartenschlüssel des Standortes für maximal mögliche Kombination für eine Tierart nach Schlüsseltable (1031, 2015, 3031, 4015, 5015, 7063).

6.4.3 QSExtLocationBtartQueryReply - die Antwortdaten

Die Antwortnachricht enthält die Produktionsarten, für die der Standort eine Zulassung besitzt. Für jede Produktionsart wird angezeigt, ob für den Standort ein gültiger Systemvertrag vorliegt (1) oder nicht (0). Bei inaktiven Standorten (geperrt wegen fehlender Salmonellenproben) wird 2 zurückgeliefert. Bei Standorten, die aus anderen Gründen gesperrt sind, wird 3 zurückgegeben. Des Weiteren wird der Zulassungsstatus der Produktionsart in Klartext zurückgeliefert. Bei zugelassenen Standorten „yes“, bei nicht zugelassenen Standorten „no“, bei inaktiven Standorten (gesperrt wegen fehlender Salmonellenproben) „inactive“ und bei gesperrten Standorten „locked“. In der Antwortnachricht wird zusätzlich das Datum, bis wann dieser Standort zugelassen ist `accreditedUntil`, übertragen.

```
<complexType name="QSExtLocationBtartQueryReply">
  <sequence>
    <element name="listOfBtartId" type="xsd:apachesoap:Vector"/>
  </sequence>
</complexType>

<complexType name="QSExtLocationBtartQueryType">
  <sequence>
    <element name="btartId" type="tns3:CS840018Type"/>
    <element name="certifiedLocation" type="xsd:int"/>
    <element name="description" nillable="true" type="tns3:AN99Type"/>
    <element name="accreditedUntil" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

listOfBtartId: Liste der gefundenen Betriebsart.

btartId: Der Produktionsartenschlüssel des Standortes.

certifiedLocation: Schlüsselwert, der anzeigt ob für einen Standort ein gültiger Systemvertrag vorliegt oder nicht. Schlüssel: 0 = no, 1 = yes, 2 = inactive, 3 = locked



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



description: Beschreibung des Schlüsselwertes.

accreditedUntil: Datum, bis wann dieser Standort zugelassen ist.

6.4.4 QSExtLocationBtartQueryReply - zusätzliche Beschreibung

Diese Nachricht gleicht der Nachricht 6.3. Im Unterschied zu 6.3 wird hier aber nicht nur eine einzige Grundart zugelassen, sondern im Ergebnisreply werden alle in der übermittelten Produktionsart enthaltenen Grundarten überprüft und für jede dieser Grundarten wird das jeweilige Ergebnis übermittelt. Demnach wird z. B. bei der Übermittlung von 1005 als btartID, sowohl 1004, als auch 1001 überprüft, während bei der 1015 jede Grundart bis zur 8 enthalten ist.

6.5 Auskunft über Zusammensetzung eines Bündels

Vorbedingung: Die Nummer des entsprechenden Bündels muss bekannt sein. Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein.

Nachbedingung: Im Erfolgsfall wird ein Vektor mit Elementen des Typs `QSBdlRequestType` mit Informationen zu Teilnehmernummer, VVVO-Nummer, Produktionsart und Status zurückgeschickt. Sonst wird eine entsprechende Meldung generiert und zurückgeschickt.

6.5.1 QSBdlRequestMsg - Informationen zu einzelnen Bündeln

```
<complexType name="QSBdlRequestMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
    <element name="bundleNr" type="xsd:int"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

questionerId: Teilnehmernummer des anfragenden Bündlers.

bundleNr: Interne, vom QS-System automatisch vergebene Nummer des Bündels, nicht die pro Bündler vergebene laufende Nummer. Wenn als Bündelnummer die Zahl -1 angegeben wird, erhält man in der Antwort alle ungebündelten Betriebe.

6.5.2 QSBdlRequestMsgReply - die Antwortdaten

Die Antwortnachricht enthält einen Vektor von Objekten des Typs `QSBdlRequestType` in denen die Informationen zu den Systempartnern im Bündel verkapselt sind.

```
<complexType name="QSBdlRequestMsgReply">
  <sequence>
    <element name="bundles" nillable="true" type="apachesoap:Vector"/>
  </sequence>
</complexType>

<complexType name="QSBdlRequestType">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="btartId" nillable="true" type="tns3:CS840018Type"/>
  </sequence>
</complexType>
```



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



```
<element name="statusId" nillable="true" type="tns3:CS80004Type"/>
</sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

locationId: Standortkennung,

btartId: Produktionsart,

statusId: QS-Status des Standortes.

6.6 Adressauskunft über einen gebündelten Betrieb

Vorbedingung: Kenntnis von Teilnehmernummer, Standortnummer und Produktionsart. Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein.

Nachbedingung: Im Erfolgsfall wird eine Nachricht mit den aktuell in der Datenbank gespeicherten Adressdaten des Teilnehmers zurückgeliefert, ansonsten eine Fehlermeldung.

6.6.1 QSBdlStortInfoMsg - Adressinformation zu Standorten

```
<complexType name="QSBdlStortInfoMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="btartId" nillable="true" type="tns3:CS840018Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

questionerId: Teilnehmernummer des anfragenden Bündlers.

locationId: Standortkennung,

btartId: Produktionsart,

6.6.2 QSBdlStortInfoMsgReply - die Antwortdaten

Die Antwort besteht aus einem Objekt, das die Adressinformation und das aktuelle Datum enthält.

```
<complexType name="QSBdlStortInfoMsgReply">
  <sequence>
    <element name="stortInfo" nillable="true" type="tns3:QSBdlStortInfoType"/>
    <element name="refDate" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```

```
<complexType name="QSBdlStortInfoType">
  <sequence>
    <element name="fax" nillable="true" type="tns3:AN99Type"/>
    <element name="street" nillable="true" type="tns3:AN99Type"/>
    <element name="country" nillable="true" type="tns3:AN99Type"/>
  </sequence>
</complexType>
```



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



```
<element name="telephone" nillable="true" type="tns3:AN99Type"/>
<element name="city" nillable="true" type="tns3:AN99Type"/>
<element name="pob" nillable="true" type="tns3:AN99Type"/>
<element name="dateOfBirth" nillable="true" type="xsd:dateTime"/>
<element name="zipPob" nillable="true" type="tns3:AN10Type"/>
<element name="EMail" nillable="true" type="tns3:AN99Type"/>
<element name="firstName" nillable="true" type="tns3:AN99Type"/>
<element name="lastName" nillable="true" type="tns3:AN99Type"/>
<element name="zipStreet" nillable="true" type="tns3:AN10Type"/>
</sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

stortInfo: Objekt mit den Adressdaten.

refDate: aktuelles Tagesdatum.

6.7 Ummelden von Standortkennungen

Vorbedingung: Ein Systempartner `questionerId` meldet eine andere Standortkennung für einen vorhandenen Standort. Kennung von Standort und Produktionsart müssen bekannt sein. Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein.

Nachbedingung: Die Standortnummer wird geändert. Die Änderung wird in der QS-Datenbank protokolliert. Die anfragende Stelle erhält als Bestätigung ein Nachrichtenobjekt mit einem Zeitstempel. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.

6.7.1 QSUpdateLocationMsg - Ummelden von Standortkennungen

```
<complexType name="QSUpdateLocationMsg">
  <sequence>
    <element name="locationIdOld" nillable="true" type="tns3:AN15Type"/>
    <element name="locationIdNew" nillable="true" type="tns3:AN15Type"/>
    <element name="bstartId" nillable="true" type="tns3:CS840018Type"/>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

locationIdOld: Die bisherige Standortnummer des Standortes (die VVVO-Nummer bei landwirtschaftlichen Betrieben).

locationIdNew: Die neue Standortnummer des Standortes.

bstartId: Der Produktionsartenschlüssel des Standortes nach Schlüsseltablelle.

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



6.7.2 CQSSUpdateMsgReply - die Antwortdaten

Die Antwortdaten enthalten einen Zeitstempel als Bestätigung der durchgeführten Aktualisierung.

```
<complexType name="CQSSUpdateMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```

6.8 Ummelden von Produktionsarten

Vorbedingung: Ein Systempartner `questionerId` meldet die Produktionsart für einen bestimmten Standort um. Kennung von Standort und Produktionsart müssen bekannt sein. Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein.

Nachbedingung: Die anfragende Stelle erhält als Bestätigung ein Nachrichtenobjekt mit einem Zeitstempel. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.

6.8.1 QSUpdateBartMsg - Ummelden von Produktionsarten

```
<complexType name="QSUpdateBartMsg">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="bartIdOld" nillable="true" type="tns3:CS840018Type"/>
    <element name="bartIdNew" nillable="true" type="tns3:CS840018Type"/>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

locationId: Die Standortnummer des Standortes (die VVVO-Nummer bei landwirtschaftlichen Betrieben).

bartIdOld: Bisherige Produktionsart nach Schlüsseltabelle.

bartIdNew: Aktuelle Produktionsart des Standortes nach Schlüsseltabelle.

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.

6.8.2 CQSSUpdateMsgReply - die Antwortdaten

Die Antwortdaten enthalten einen Zeitstempel als Bestätigung der durchgeführten Aktualisierung.

```
<complexType name="CQSSUpdateMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```



6.9 Ummelden von Prüfinstituten

Vorbedingung: Ein Systempartner `questionerId` meldet für einen Standort ab einem bestimmten Datum ein neues Prüfinstitut an. Ab dem entsprechenden Zeitpunkt erlöschen die Zugriffsrechte für das bisherige Prüfinstitut.

Nachbedingung: Die anfragende Stelle erhält als Bestätigung ein Nachrichtenobjekt mit einem Zeitstempel. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.

6.9.1 QSUpdateCertBodyMsg - Ummelden von Prüfinstituten

```
<complexType name="QSUpdateCertBodyMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="btaartId" nillable="true" type="tns3:CS840018Type"/>
    <element name="validFrom" nillable="true" type="xsd:dateTime"/>
    <element name="certificationBody" nillable="true" type="tns3:AN15Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.

locationId: Die Standortnummer des Standortes.

btaartId: Der Produktionsartenschlüssel des Standortes nach Schlüsseltable.

certificationBody: QS-Teilnehmerkennung des Prüfinstituts.

validFrom: Datum ab dem die neuen Berechtigungen gelten sollen.

6.9.2 CQSSUpdateMsgReply - die Antwortdaten

Die Antwortdaten enthalten einen Zeitstempel als Bestätigung der durchgeführten Aktualisierung.

```
<complexType name="CQSSUpdateMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```

6.10 Änderung von Adressdaten eines Standortes

Vorbedingung: Ein Bündler `questionerId` meldet für einen Standort `locationId` + `btaartId` Änderungen der Adressdaten. Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein.

Nachbedingung: Die Änderung wird durchgeführt und im Erfolgsfall ein Zeitstempel zurückgeschickt. Sonst erfolgt eine entsprechende Fehlermeldung.

6.10.1 CQSSUpdateLocAdrMsg - Änderung von Adressdaten eines Standortes

```
<complexType name="CQSSUpdateLocAdrMsg">
  <sequence>
```



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



```
<element name="locationId" nillable="true" type="tns3:AN15Type" />
<element name="bstartId" nillable="true" type="tns3:CS840018Type" />
<element name="questionerId" nillable="true" type="tns3:AN15Type" />
<element name="oldValue" nillable="true" type="tns3:CQSSAddressType" />
<element name="newValue" nillable="true" type="tns3:CQSSAddressType" />
</sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

locationId: Standortkennung.

bstartId: Produktionsart.

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.

oldValue: alte Adresse. Muss nicht übermittelt werden.

newValue: neue Adresse.

6.10.2 CQSSUpdateMsgReply - die Antwortdaten

Die Antwortdaten enthalten einen Zeitstempel als Bestätigung der durchgeführten Aktualisierung.

```
<complexType name="CQSSUpdateMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:dateTime" />
  </sequence>
</complexType>
```

6.11 Melden von Änderungen in den Kommunikationsdaten

Vorbedingung: Ein Bündler `questionerId` meldet für einen von ihm gebündelten Standort `locationId` + `bstartId` Änderungen der Telefon-, Fax- oder Emailverbindung. Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein. **Nachbedingung:** Änderung wird durchgeführt und im Erfolgsfall ein Zeitstempel zurückgeschickt. Sonst erfolgt eine entsprechende Fehlermeldung.

6.11.1 CQSSUpdateLocComMsg - Änderung von Kommunikationsdaten eines Standortes

```
<complexType name="CQSSUpdateLocComMsg">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type" />
    <element name="bstartId" nillable="true" type="tns3:CS840018Type" />
    <element name="questionerId" nillable="true" type="tns3:AN15Type" />
    <element name="oldValue" nillable="true" type="tns3:CQSSCommunicationType" />
    <element name="newValue" nillable="true" type="tns3:CQSSCommunicationType" />
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

bstartId: Produktionsart des Standortes.

locationId: Produktionsart des Standortes.



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.

oldValue: alte Verbindungsdaten. Muss nicht übermittelt werden.

newValue: neue Verbindungsdaten.

6.11.2 CQSSUpdateMsgReply - die Antwortdaten

Die Antwortdaten enthalten einen Zeitstempel als Bestätigung.

```
<complexType name="CQSSUpdateMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```

6.12 Abmelden von Systempartnern

Vorbedingung: Ein Bündler `questionerId` meldet alle Standorte eines von ihm gebündelten Systempartners `clientId` zu einem bestimmten Datum ab. Beide QS-Identifikationen müssen bekannt sein. Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein.

Nachbedingung: Alle Systempartner werden als abgemeldet gekennzeichnet. Alle Zugriffsrechte erlöschen zum angegebenen datum `refDate`. Der Abmeldevorgang wird in der QS-Datenbank protokolliert. Als Bestätigung wird ein Nachrichtenobjekt mit einem Zeitstempel zurückgesendet. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.

6.12.1 QSDeleteMemberMsg - Abmelden von allen Standorten eines Systempartners

```
<complexType name="QSDeleteMemberMsg">
  <sequence>
    <element name="clientId" nillable="true" type="tns3:AN15Type"/>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
    <element name="refDate" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

clientId: QS-Teilnehmerkennung des auditierten Betriebes.

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.

refDate: Datum zu dem der Teilnehmer abgemeldet werden soll.

6.12.2 CQSSUpdateMsgReply - die Antwortdaten

Die Antwortdaten enthalten einen Zeitstempel als Bestätigung der durchgeführten Aktualisierung.

```
<complexType name="CQSSUpdateMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```



6.13 Abmelden von Standorten

Vorbedingung: Ein Bündler `questionerId` meldet einen Standort `locationId` einer bestimmten Produktionsart `btartId` zu einem bestimmten Datum ab. Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein.

Nachbedingung: Der Standort wird als abgemeldet gekennzeichnet. Alle Zugriffsrechte werden zum angegebenen Datum `refdate` entzogen. Die anfragende Stelle erhält als Bestätigung ein Nachrichtenobjekt mit einem Zeitstempel. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.

6.13.1 QSDeleteLocationMsg - Abmelden von Standorten

```
<complexType name="QSDeleteLocationMsg">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="btartId" nillable="true" type="tns3:CS840018Type"/>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
    <element name="refDate" nillable="true" type="xsd:date"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

locationId: Die Standortnummer des Standortes (die VVVO-Nummer bei landwirtschaftlichen Betrieben).

btartId: Der Produktionsartenschlüssel des Standortes nach Schlüsseltabelle.

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.

refDate: Datum zu dem der Standort abgemeldet werden soll.

6.13.2 CQSSUpdateMsgReply - die Antwortdaten

Die Antwortdaten enthalten einen Zeitstempel als Bestätigung der durchgeführten Aktualisierung.

```
<complexType name="CQSSUpdateMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:date"/>
  </sequence>
</complexType>
```

6.14 Abfragen der aktuellsten Prüfberichte

Ein Bündler `questionerId` fragt für einen bestimmten Zeitraum (`dateFrom` bis `dateTo`) die aktuellsten Prüfberichte seiner Standorte ab.

Vorbedingung: Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein.

Nachbedingung: Die für den angegebenen Zeitraum aktuellsten Prüfberichte der Standorte sind bekannt. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.



6.14.1 QSLastInspectionMsg - Abfragen der aktuellsten Prüfberichte

```
<complexType name="QSLastInspectionMsg">  
  <sequence>  
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>  
    <element name="dateFrom" nillable="true" type="xsd:dateTime"/>  
    <element name="dateTo" nillable="true" type="xsd:dateTime"/>  
  </sequence>  
</complexType>
```

Die XML-Elemente haben folgende Semantik:

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.

dateFrom: Anfang des abzufragenden Zeitraums, in dem das Auditdatum liegen kann.

dateTo: Ende des abzufragenden Zeitraums, in dem das Auditdatum liegen kann.

6.14.2 QSLastInspectionMsgReply - die Antwortdaten

Die Antwortdaten enthalten Informationen über den letzten Prüfbericht in dem angegebenen Zeitraum pro Standort.

```
<complexType name="QSLastInspectionMsgReply">  
  <sequence>  
    <element name="inspections" nillable="true" type="apachesoap:Vector"/>  
  </sequence>  
</complexType>  
  
<complexType name="QSLastInspectionType">  
  <sequence>  
    <element name="inspectionId" nillable="true" type="xsd:int"/>  
    <element name="checklistId" nillable="true" type="xsd:int"/>  
    <element name="checklistName" nillable="true" type="xsd:string"/>  
    <element name="dateOfInspection" nillable="true" type="xsd:dateTime"/>  
    <element name="auditType" nillable="true" type="xsd:int"/>  
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>  
    <element name="locationType" nillable="true" type="tns3:CS840018Type"/>  
    <element name="inspectedType" nillable="true" type="tns3:CS840018Type"/>  
    <element name="state" nillable="true" type="tns3:CS80004Type"/>  
  </sequence>  
</complexType>
```

Die XML-Elemente haben folgende Semantik:

inspections: Liste der gefundenen Prüfberichte.

inspectionId: Eindeutige Identifikation der ausgefüllten Prüfliste.

checklistId: Eindeutige Identifikation des Prüflistenformulars.



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



checklistName: Name der Prüfliste.

dateOfInspection: Datum der Prüfung.

auditType: Typ der Prüfung.

locationId: Standortnummer des geprüften Standorts.

locationType: Betriebsart des geprüften Standorts.

inspectedType: Überprüfte Betriebsart des Standorts.

state: Errechneter QS-Status.

6.15 Abfragen der zuletzt verschickten Prüfberichte

Ein Bündler `questionerId` fragt für einen bestimmten Zeitraum (`dateFrom` bis `dateTo`) die zuletzt verschickten Prüfberichte seiner Standorte ab. D. h. der angegebene Zeitraum bezieht sich auf das Datum, wann der Mängelbericht verschickt worden ist.

Vorbedingung: Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein.

Nachbedingung: Die für den angegebenen Zeitraum zuletzt freigegebenen Prüfberichte der Standorte sind bekannt. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.

6.15.1 QSLastSentInspectionMsg - Abfragen der zuletzt verschickten Prüfberichte

```
<complexType name="QSLastSentInspectionMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
    <element name="dateFrom" nillable="true" type="xsd:dateTime"/>
    <element name="dateTo" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.

dateFrom: Anfang des abzufragenden Zeitraums, in dem der Prüfbericht versandt worden sein kann..

dateTo: Ende des abzufragenden Zeitraums, in dem der Prüfbericht versandt worden sein kann.

6.15.2 QSLastInspectionMsgReply - die Antwortdaten

siehe Abschnitt 6.14.2

6.16 Erweiterte Abfrage von Prüflisten eines Standortes

Vorbedingung: Ein Systempartner `questionerId` fragt ein bestimmtes Prüfergebnis eines Standortes ab. Produktionsart und Standortkennung sind bekannt und die Zugriffsberechtigungen vorhanden.

Nachbedingung: Prüfergebnis und Bewertung aller Prüfpunkte sind bekannt. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.

6.16.1 QSExtInspectionMsg - Abfrage von Prüfergebnissen

```
<complexType name="QSExtInspectionMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
  </sequence>
</complexType>
```



```
<element name="locationId" nillable="true" type="tns3:AN15Type" />
<element name="locationType" nillable="true" type="tns3:CS840018Type" />
<element name="dateOfInspection" nillable="true" type="xsd:dateTime" />
</sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers (Bündlers).

locationId: Die Standortnummer des Standortes (die VVVO-Nummer bei landwirtschaftlichen Betrieben).

locationType: Der Produktionsartenschlüssel des Standortes nach Schlüsselstabelle.

dateOfInspection: Datum des Audits (wenn dieses nicht klar zugeordnet werden kann, wird der aktuellste Bericht ausgewählt).

6.16.2 QSExtInspectionMsgReply - die Antwortdaten

Die Antwortdaten enthalten die komplette Information über den Prüfvorgang einschließlich der einzelnen Prüfpunkte ExtCheckListItem, die in einem Vektor checklistItems verkapselt werden.

```
<complexType name="QSExtInspectionMsgReply">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type" />
    <element name="locationType" nillable="true" type="tns3:CS840018Type" />
    <element name="inspectionId" nillable="true" type="xsd:int" />
    <element name="checklistId" nillable="true" type="xsd:int" />
    <element name="checklistName" nillable="true" type="xsd:string" />
    <element name="inspectedType" nillable="true" type="tns3:CS840018Type" />
    <element name="inspectionType" nillable="true" type="xsd:int" />
    <element name="dateOfInspection" nillable="true" type="xsd:dateTime" />
    <element name="certificationBody" nillable="true" type="tns3:AN15Type" />
    <element name="auditor" nillable="true" type="tns3:AN60Type" />
    <element name="state" nillable="true" type="tns3:CS80004Type" />
    <element name="percentage" type="xsd:double" />
    <element name="fromTime" type="xsd:dateTime" />
    <element name="toTime" type="xsd:dateTime" />
    <element name="inspectionDuration" type="xsd:double" />
    <element name="informant" type="xsd:string" />
    <element name="lackReportDate" type="xsd:dateTime" />
    <element name="leftoverFeeding" type="xsd:string" />
    <element name="selfMixer" type="xsd:string" />
    <element name="betterments" nillable="true" type="xsd:string" />
    <element name="comment" nillable="true" type="xsd:string" />
    <element name="generalKo" nillable="true" type="xsd:boolean" />
    <element name="generalKoComment" nillable="true" type="xsd:string" />
  </sequence>
</complexType>
```



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



```
<element name="numberOfPlaces" nillable="true" type="apachesoap:Vector"/>
<element name="checklistItems" nillable="true" type="apachesoap:Vector"/>
<element name="addOnChecklists" nillable="true" type="apachesoap:Vector"/>
</sequence>
</complexType>
```

```
<complexType name="ExtChecklistItem">
  <sequence>
    <element name="id" nillable="true" type="xsd:int"/>
    <element name="numbering" nillable="true" type="xsd:string"/>
    <element name="name" nillable="true" type="xsd:string"/>
    <element name="knockOut" nillable="true" type="xsd:string"/>
    <element name="remark" nillable="true" type="xsd:string"/>
    <element name="mark" nillable="true" type="xsd:string"/>
    <element name="timeLimit" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```

```
<complexType name="QSPlacesType">
  <sequence>
    <element name="locationType" nillable="true" type="tns3:CS840018"/>
    <element name="stablePlaces" nillable="true" type="xsd:int"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

locationId: Die Standortnummer des Standortes (die VVVO-Nummer bei landwirtschaftlichen Betrieben).

locationType: Der Produktionsartenschlüssel des Standortes nach Schlüsseltablelle.

inspectionId: Eindeutige Identifikation der ausgefüllten Prüfliste.

checklistId: Eindeutige Identifikation des Prüflistenformulars.

checklistName: Name der Prüfliste.

inspectedType: Der Produktionsartenschlüssel des Standortes nach Schlüsseltablelle. Angegeben wird hier die tatsächlich auditierte Produktionsart, die eine Teilmenge der gemeldeteten Produktionsart sein kann.

inspectionType: Typ der Prüfung (1=Systemprüfung; 2=Stichprobe; 3=Sonder; 4=Parallel; 5=Spot; 6=FIAS; 99=unangekündigtes Systemaudit).

dateOfInspection: Datum des Audits.

certificationBody: QS-Teilnehmerkennung des Prüfinstituts.

auditor: Name des Auditors eines Prüfinstituts, der die Inspektion durchgeführt hat.



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



state: Teilnehmerstatus nach Prüfbericht.

percentage: Prozentuales Prüfergebnis.

fromTime: Uhrzeit des Auditbeginns.

toTime: Uhrzeit des Auditendes

inspectionDuration: Prüfdauer des Audits.

informant: Auskunftsperson des Audits

lackReportDate: Datum, an dem der Mängelbericht versendet worden ist.

leftoverFeeding: Speiseresteverfütterung ("j" = ja, "n" = nein)

selfMixer: Selbstmischer ("j" = ja, "n" = nein)

betterments: Abzustellende Mängel im Format:

Nummer~

Anforderung~

Antwort~

Abweichung~

Korrekturmaßnahme~

Behebungsfrist~

Umsetzung~

BehebungsDatum~

Status|

Status:

1 = offen

2 = begonnen

3 = nicht behoben

4 = fristgerecht behoben

5 = Ungeprüft erledigt

6 = nachträglich behoben

comment: Schriftliche Bemerkungen des Auditors.

generalKo: Wiederholtes D-Kriterium ("j" = ja, "n" = nein).

generalKoComment: Kommentar zum wiederholten D-Kriterium.

numberOfPlaces: Vektor, der die Anzahl der Stallplätze pro überprüfter Betriebsart verkapselt.

ChecklistItems: Vektor, der die einzelnen Prüfpunkte vom Typ QSChecklistItems verkapselt.

id: Eindeutige Identifikation des Prüfpunktes.

numbering: laufende Nummer des Prüfpunkts.

name: Bezeichnung des Prüfpunkts.



knockOut: K.O.-Kriterium (ja/nein).

mark: Ergebnis für den Prüfpunkt.

remark: Bemerkung zu dem Prüfpunkt.

timeLimit: Frist zu dem der angegebene Mangel behoben werden muss.

locationType: Betriebsartenschlüssel.

stablePlaces: Anzahl Stallplätze.

6.17 Abfragen von Prüfberichten

Vorbedingung: Ein Bündler `questionerId` fragt für eine bestimmte Prüflistennummer `inspectionId` ein Prüfbericht ab. Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein.

Nachbedingung: Der Prüfbericht für die angegebene Prüflistennummer wird zurückgegeben. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.

6.17.1 QSIinspectionFromIdMsg - Abfragen von Prüfberichten

```
<complexType name="QSIinspectionFromIdMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
    <element name="inspectionId" nillable="true" type="xsd:int"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.

inspectionId: Eindeutige Identifikation des Prüfberichts.

6.17.2 QSExtInspectionMsgReply - die Antwortdaten

Die Antwortdaten enthalten Informationen über den angefragten Prüfbericht. Die Antwortnachricht enthält die gleichen Daten wie die Antwortnachricht der QSExtInspectionMsg.

6.18 Erweiterte Abfrage von Prüflisten eines Standortes in einem Zeitraum

6.18.1 QSExtInspectionsInPeriodMsg - Erweiterte Abfrage aller Prüfergebnisse eines bestimmten Zeitraums

```
<complexType name="QSExtInspectionsInPeriodMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="locationType" nillable="true" type="tns3:CS840018Type"/>
    <element name="fromDate" nillable="true" type="xsd:dateTime"/>
    <element name="toDate" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers (Bündlers).

locationId: Die Standortnummer des Standortes (die VVVO-Nummer bei landwirtschaftlichen Betrieben).

locationType: Der Produktionsartenschlüssel des Standortes nach Schlüsseltable.

fromDate: Kleinstes mögliches Datum eines Audits.

toDate: Größtes mögliches Datum eines Audits.

6.18.2 QSExtInspectionsInPeriodMsgReply - die Antwortdaten

Die Antwortdaten enthalten die komplette Information über alle Prüfvorgänge im abgefragten Zeitraum einschließlich der einzelnen Prüfpunkte ExtChecklistItem, die in einem Vektor checklistItems verkapselt werden. Die Einzelnen Prüfberichte werden ihrerseits in einem Vektor von Einzelantworten zusammengefasst.

```
<complexType name="QSExtInspectionsInPeriodMsgReply">
  <sequence>
    <element name="inspectionMsg" nillable="true" type="apachesoap:Vector"/>
  </sequence>
</complexType>
```

Der Vector enthält die Daten von Datentyp QSExtInspectionMsgReply.

6.19 Abfragen von Standortinformationen

Vorbedingung: Ein Bündler `questionerId` fragt für einen bestimmten Standort ab, ob dieser bei ihm gebündelt ist. Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein.

Nachbedingung: Ist der Standort beim anfragenden Bündler gebündelt, wird das Anmeldedatum, das evtl. Abmeldedatum und der Bündelname zurückgeliefert. Ist dieser Standort bei einem anderen Bündler gebündelt, bekommt der anfragende Bündler nur die Information, ob dieser Abgemeldet ist oder nicht `signOffDate`. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.

6.19.1 QSLocationInfoMsg - Abfragen von Standortinformationen

```
<complexType name="QSLocationInfoMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="locationType" nillable="true" type="tns3:CS840018Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.

locationId: Standortnummer des abzufragenden Standorts.

locationType: Betriebsart des abzufragenden Standorts.

6.19.2 QSLocationInfoMsgReply - die Antwortdaten

Die Antwortdaten enthalten Informationen über den angefragten Standort.

```
<complexType name="QSLocationInfoMsgReply">
```



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



```
<sequence>
  <element name="ownLocation" nillable="true" type="xsd:boolean"/>
  <element name="pendingSince" nillable="true" type="xsd:dateTime"/>
  <element name="signOffDate" nillable="true" type="xsd:dateTime"/>
  <element name="bundleName" nillable="true" type="tns3:AN99Type"/>
  <element name="state" nillable="true" type="tns3:CS80004Type"/>
</sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

ownLocation: Kennzeichen, ob ein Standort von anfragenden Bündler gebündelt wird oder nicht.

pendingSince: Datum, seit wann dieser Standort bei dem anfragenden Bündler gebündelt wird.

signOffDate: Datum der Abmeldung, falls vorhanden.

bundleName: Name des Bündels. Das Feld bleibt leer, falls der Standort nicht aktuell beim anfragenden Bündler gebündelt ist.

state: QS-Status des Standorts. Das Feld bleibt leer, falls der Standort nicht aktuell beim anfragenden Bündler gebündelt ist.

6.20 Erstellen von Bündeln

Vorbedingung: Die mitgeschickten Standorte müssen der QS-Datenbank bekannt sein und dürfen noch nicht gebündelt sein. Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein.

Nachbedingung: Die mitgelieferten Standorte wurden in dem mitgelieferten Bündel zusammengefasst. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.

6.20.1 QSBuildBundleMsg - Erstellen von Bündeln

```
<complexType name="QSBuildBundleMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
    <element name="bundleName" nillable="true" type="tns3:AN99Type"/>
    <element name="locations" nillable="true" type="apachesoap:Vector"/>
  </sequence>
</complexType>
<complexType name="QSBuildBundleLocationType">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="locationType" nillable="true" type="tns3:CS840018Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.

bundleName: Name des Bündels.



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



locations: Liste von Standorten, die in einem Bündel zusammengefasst werden sollen.

locationId: Standortnummer des Standorts (Bei landw. Standorten die VVONr.).

locationType: Betriebsart des Standorts.

6.20.2 QSBuildBundleMsgReply - die Antwortdaten

Die Antwortdaten enthalten Informationen über das angelegte Bündel und über evtl. fehlerhafte Standorte.

```
<complexType name="QSBuildBundleMsgReply">
  <sequence>
    <element name="bundleId" nillable="true" type="xsd:int"/>
    <element name="errorLocations" nillable="true" type="apachesoap:Vector"/>
  </sequence>
</complexType>
```

```
<complexType name="QSBuildBundleLocationErrorType">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="locationType" nillable="true" type="tns3:CS840018Type"/>
    <element name="errorCode" nillable="true" type="xsd:int"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

bundleId: Eindeutige Identifikation des Bündels

errorLocations: Liste von fehlerhaften Standorten, die nicht gebündelt werden konnten.

locationId: Standortnummer des Standorts (Bei landw. Standorten die VVVO-Nr.).

locationType: Betriebsart des Standorts.

errorCode: Fehlernummer.

6.20.3 QSBuildBundleMsg - zusätzliche Beschreibung

Es können nur Standorte mit dem Zertifizierungssystem GlobalGap und AMA-Gap gebündelt werden.

6.21 Änderung von Adressdaten eines Unternehmens

Vorbedingung: Ein Bündler `questionerId` meldet für ein Unternehmen `clientId` Änderungen der Adressdaten. Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein.

Nachbedingung: Die Änderung wird durchgeführt und im Erfolgsfall ein Zeitstempel zurückgeschickt. Sonst erfolgt eine entsprechende Fehlermeldung.

6.21.1 CQSSUpdateCompAdrMsg - Änderung von Adressdaten eines Unternehmens

```
<complexType name="CQSSUpdateCompAdrMsg">
  <sequence>
    <element name="clientId" nillable="true" type="tns3:AN15Type"/>
  </sequence>
```



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



```
<element name="questionerId" nillable="true" type="tns3:AN15Type"/>
<element name="newValue" nillable="true" type="tns3:CQSSAddressType"/>
</sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

clientId: QS-ID des Unternehmens.

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.

newValue: neue Adresse.

6.21.2 CQSSUpdateMsgReply - die Antwortdaten

Die Antwortdaten enthalten einen Zeitstempel als Bestätigung der durchgeführten Aktualisierung.

```
<complexType name="CQSSUpdateMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```

6.22 Melden von Änderungen in den Kommunikationsdaten eines Unternehmens

Vorbedingung: Ein Bündler `questionerId` meldet für ein Unternehmen `clientId` Änderungen der Telefon-, Fax- oder E-Mail-Verbindung. Es müssen entsprechende Zugriffsberechtigungen in der Datenbank vergeben sein. Als zusätzliche Sicherheit müssen die ursprünglichen Kommunikationsdaten bekannt sein.

Nachbedingung: Änderung wird durchgeführt und im Erfolgsfall ein Zeitstempel zurückgeschickt. Sonst erfolgt eine entsprechende Fehlermeldung.

6.22.1 CQSSUpdateCompComMsg - Änderung von Kommunikationsdaten eines Unternehmens

```
<complexType name="CQSSUpdateCompComMsg">
  <sequence>
    <element name="clientId" nillable="true" type="tns3:AN15Type"/>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
    <element name="oldValue" nillable="true" type="tns3:CQSSCommunicationType"/>
    <element name="newValue" nillable="true" type="tns3:CQSSCommunicationType"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

clientId: QS-ID des Unternehmens.

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers.

oldValue: alte Verbindungsdaten.

newValue: neue Verbindungsdaten.



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



6.22.2 CQSSUpdateMsgReply - die Antwortdaten

Die Antwortdaten enthalten einen Zeitstempel als Bestätigung.

```
<complexType name="CQSSUpdateMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```

6.23 Ändern von Standortdaten

Vorbedingung: Ein Systempartner `questionerId` meldet eine Änderung der Standortdaten für einen vorhandenen Standort. Zurzeit kann nur die Gruppenzuordnung geändert werden. Produktionsart und Standortkennung des zu ändernden Standortes sind bekannt.

Nachbedingung: Die Standortdaten sind geändert. Im Fehlerfall wird eine entsprechende Meldung generiert und zurückgeschickt.

6.23.1 QSChangeLocationDataMsg - Ändern von Standortdaten

```
<complexType name="QSChangeLocationDataMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="locationType" nillable="true" type="tns3:CS840018Type"/>
    <element name="groupFlag" nillable="true" type="tns3:AN99Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers (Bündlers).

locationId: Die Standortnummer des Standortes.

locationType: Der Produktionsartenschlüssel des Standortes nach Schlüsseltable.

groupFlag: Gruppenzuordnung eines Standortes

6.23.2 CQSSUpdateMsgReply - die Antwortdaten

Die Antwortdaten enthalten einen Zeitstempel als Bestätigung der durchgeführten Aktualisierung.

```
<complexType name="CQSSUpdateMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```



6.23.3 CQSSUpdateMsg - zusätzliche Beschreibung

Über das groupFlag können zusätzliche Attribute des Standortes verändert werden.

- SPOTAUDIT: YES/NO Bestimmt, ob für diesen Standort zusätzliche Spotaudits durchgeführt werden sollen.
- FIAS: YES/NO Bestimmt, ob eine FIAS Teilnahme gewünscht wird.
- REGIO: YES/NO Bestimmt, ob eine Regionalfenster Teilnahme gewünscht wird.
- VLOG: YES/NO Bestimmt, ob eine VLOG Teilnahme gewünscht wird.

6.24 Abfrage des aktuellen Prüfinstituts

Ein Systempartner `questionerId` fragt das aktuelle Prüfinstitut für einen von ihm gebündelten Standort ab.

Vorbedingung: Produktionsart und Standortkennung des zu ändernden Standortes sind bekannt. Dem Standort ist aktuell genau ein Prüfinstitut zugeordnet.

Nachbedingung: Das aktuelle Prüfinstitut wird zurückgegeben. Sollte dem Standort kein oder mehrere Prüfinstitute zugeordnet sein, so wird eine entsprechende Fehlermeldung zurückgegeben.

6.24.1 QSCertBodyQueryMsg - Abfrage des aktuellen Prüfinstituts

```
<complexType name="QSCertBodyQueryMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type"/>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="locationType" nillable="true" type="tns3:CS840018Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers (Bündlers).

locationId: Die Standortnummer des Standortes.

locationType: Der Produktionsartenschlüssel des Standortes nach Schlüsseltabelle.

6.24.2 QSCertBodyQueryMsgReply - die Antwortdaten

Die Antwortdaten enthalten die QS-ID und die Adresse des aktuellen Prüfinstituts.

```
<complexType name="QSCertBodyQueryMsgReply">
  <sequence>
    <element name="certificationBody" nillable="true" type="tns3:AN15Type"/>
    <element name="address" nillable="true" type="tns3:CQSSAddressType"/>
  </sequence>
</complexType>
```

6.25 Meldung eines Systempartners mit anerkanntem Fremdsystem

Vorbedingung: Der Bündler ist dem System bekannt.

Nachbedingung: Der Systempartner bekommt eine Mitgliedsnummer und wird für die Laufzeit der Zertifizierung als QS-zugelassen geführt. Er ist daraufhin dem System bekannt. Im Fehlerfall wird eine entsprechende Fehlermeldung generiert und an den meldenden Teilnehmer zurückgeschickt.



6.25.1 CQSSRegisterCertificationSystemMsg - Anmeldung neuer anerkannter Systempartner

Der Bündler überträgt hierzu die Stammdaten eines Systempartners sowie eines bewirtschafteten Standortes mit den dazugehörigen Zertifikatsdaten des Fremdsystems.

```
<complexType name=" CQSSRegisterCertificationSystemMsg ">
  <sequence>
    <element name="address" nillable="true" type="tns3:CQSSAddressType"/>
    <element name="location" nillable="true" type="tns3:CQSSLocationType"/>
    <element name="com" nillable="true" type="tns3:CQSSCommunicationType"/>
    <element name="bundlerId" nillable="true" type="tns3:AN15Type"/>
    <element name="representativeAddress" nillable="true"
type="tns3:CQSSAddressType"/>
    <element name="groupFlag" nillable="true" type="tns3:AN99Type"/>
    <element name="auditScheme" nillable="true" type="xsd:int"/>
    <element name="certNr" nillable="true" type="tns3:AN99Type"/>
    <element name="accreditedUntil" nillable="true" type="xsd:dateTime"/>
    <element name="certificatedProducts" nillable="true" type="apachesoap:Vector" />
  </sequence>
</complexType>

<complexType name="QSCerificadedProductType">
  <sequence>
    <element name="productId" nillable="false" type="xsd:string" />
    <element name="area" nillable="true" type="xsd:double" />
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

address: Adresse des Systempartners.

location: Standortdaten.

com: Kommunikationseinrichtungen mit denen der Systempartner kontaktiert werden kann.

bundlerId: QS-Identifikation des Bündlers.

representativeAddress: Adresse des Ansprechpartners bzw. Verantwortlichen. Wenn Adresse des Systempartners und des Ansprechpartners identisch sind, muss hier keine weitere Adresse angegeben werden.

groupFlag: Kennzeichen, mit dem Standorte zu Gruppen zusammengefasst werden können.

auditScheme: Zertifizierungssystem.

certNr: Zertifikatsnummer.

accreditedUntil: Gültigkeit des Zertifikats.



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



certificatedProducts: Liste von angebauten Kulturen.

productId: Kulturnummer.

area: Anbaufläche.

6.25.2 CQSSRegisterMemberMsgReply - die Antwortdaten

```
<complexType name="CQSSRegisterMemberMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:dateTime"/>
    <element name="qsState" nillable="true" type="tns3:CS80004Type"/>
    <element name="clientId" nillable="true" type="tns3:AN15Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

timeStamp: Zeitstempel.

qsState: QS-Status.

clientId: Teilnehmerkennung.

6.25.3 CQSSRegisterMemberMsg - Nachrichtenbeschreibung

Diese Funktionen unterscheiden sich in ihrer Funktionalität nicht von den entsprechenden Funktionen 6.1. Zusätzlich kann hier noch ein Zertifizierungssystem angegeben werden.

Mögliche Zertifizierungssysteme sind:

1. QS
2. QS-Gap
3. QM-Milch
4. IKB
5. GlobalGap
6. Codiplan
7. IKKB
8. AMA-GAP
9. Verpflichtungserklärer Salmonellenmonitoring
10. Verpflichtungserklärer Antibiotikamonitoring
11. Verpflichtungserklärer Salmonellenmonitoring und Antibiotikamonitoring



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



Für manche Zertifizierungssysteme, muss auch für AccreditedUntil ein Wert gesetzt sein, der nicht in der Vergangenheit liegt (sonst wird eine 045, "Date collision" Exception abgesetzt) und auch in CertNr muss eventuell ein Text eingetragen sein (sonst erfolgt eine 081, "No distinct certification body found" Exception). Für GlobalGap muss zusätzlich eine Liste der zertifizierten Kulturen übertragen werden.

Sollte die aktuelle Produktionsart NICHT zum angegebenen Zertifizierungssystem passen, wird die Fehlermeldung 034, "Production line does not match with stored values" geworfen.

6.26 Meldung eines Standortes mit anerkanntem Fremdsystem

Vorbedingung: Der Bündler und der Systempartner sind dem System bekannt.

Nachbedingung: Der neue Standort ist im System bekannt und Audits können für diesen Standort erfasst werden. Im Fehlerfall wird eine entsprechende Fehlermeldung zurückgeschickt.

6.26.1 CQSSAddCertificationSystemMsg - Anmeldung eines neuen Standortes

Ein Standort ist an dieser Stelle kein geographischer Begriff, sondern eine logische Größe, die durch eine Standortnummer (z.B. VVVO-Nummer), die Kennung des Systempartners und eine Produktionsart eindeutig bestimmt ist.

```
<complexType name=" CQSSAddCertificationSystemMsg ">
  <sequence>
    <element name="clientId" nillable="true" type="tns3:AN15Type"/>
    <element name="location" nillable="true" type="tns3:CQSSLocationType"/>
    <element name="bundlerId" nillable="true" type="tns3:AN15Type"/>
    <element name="groupFlag" nillable="true" type="tns3:AN99Type"/>
    <element name="auditScheme" nillable="true" type="xsd:int"/>
    <element name="certNr" nillable="true" type="tns3:AN99Type"/>
    <element name="accreditedUntil" nillable="true" type="xsd:dateTime"/>
    <element name="certificatedProducts" nillable="true" type="apachesoap:Vector" />
  </sequence>
</complexType>
```

```
<complexType name="QSCerificadedProductType">
  <sequence>
    <element name="productId" nillable="false" type="xsd:string" />
    <element name="area" nillable="true" type="xsd:double" />
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

clientId: Systemidentifikation des Teilnehmers, von dem ein Standort gemeldet wird.

location: nachgemeldeter Standort des Systempartners.

bundlerId: Systemidentifikation des Bündlers.

groupFlag: Kennzeichen, mit dem Standorte zu Gruppen zusammengefasst werden können.

auditScheme: Zertifizierungssystem.



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



certNr: Zertifikatsnummer.

accreditedUntil: Gültigkeit des Zertifikats.

certificatedProducts: Liste von angebauten Kulturen.

productId: Kulturnummer.

area: Anbaufläche.

6.26.2 CQSSAddLocationMsgReply - die Antwortdaten

```
<complexType name="CQSSAddLocationMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:dateTime"/>
    <element name="locationId" nillable="true" type="tns3:AN15Type"/>
    <element name="qsState" nillable="true" type="tns3:CS80004Type"/>
    <element name="clientId" nillable="true" type="tns3:AN15Type"/>
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

timeStamp: Zeitstempel.

locationId: Standortidentifikationsnummer (Landwirtschaft: VVVO-Nummer und ISO Länderzeichen (276 für Deutschland)).

qsState: QS-Status des neuen.

clientId: Systemidentifikation des Teilnehmers, von dem ein Standort gemeldet wird.

6.26.3 CQSSAddLocationMsg - Nachrichtenbeschreibung

Diese Funktionen unterscheiden sich in ihrer Funktionalität nicht von den entsprechenden Funktionen 6.2. Zusätzlich kann hier noch ein Zertifizierungssystem angegeben werden.

Mögliche Zertifizierungssysteme sind:

12. QS
13. QS-Gap
14. QM-Milch
15. IKB
16. GlobalGap
17. Codiplan
18. IKKB
19. AMA-GAP
20. Verpflichtungserklärer Salmonellenmonitoring
21. Verpflichtungserklärer Antibiotikamonitoring
22. Verpflichtungserklärer Salmonellenmonitoring und Antibiotikamonitoring



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



Für manche Zertifizierungssysteme, muss auch für AccreditedUntil ein Wert gesetzt sein, der nicht in der Vergangenheit liegt (sonst wird eine 045, "Date collision" Exception abgesetzt) und auch in CertNr muss eventuell ein Text eingetragen sein (sonst erfolgt eine 081, "No distinct certification body found" Exception). Für GlobalGap muss zusätzlich eine Liste der zertifizierten Kulturen übertragen werden.

Sollte die aktuelle Produktionsart NICHT zum angegebenen Zertifizierungssystem passen, wird die Fehlermeldung 034, "Production line does not match with stored values" geworfen.

6.27 Änderung von Standortdaten (nicht mehr Unterstützt)

6.27.1 QSChangeAttributeMsg - Änderung von Standortdaten

6.27.2 CQSSUpdateMsgReply - die Antwortdaten

6.27.3 QSChangeAttributeMsg - Nachrichtenbeschreibung

Diese Nachricht wird nicht mehr unterstützt und erzeugt aus diesem Grund eine „NOT_IMPLEMENTED_EXCEPTION“

6.28 Änderung des Zertifizierungssystems

Vorbedingung: Der Bündler und der Standort sind dem System bekannt.

Nachbedingung: Das Zertifizierungssystem ist geändert. Im Fehlerfall wird eine entsprechende Fehlermeldung zurückgeschickt.

6.28.1 QSUpdateCertificationSystemMsg - Änderung des Zertifizierungssystems

Ein Standort ist an dieser Stelle kein geographischer Begriff, sondern eine logische Größe, die durch eine Standortnummer (z. B. VVVO-Nummer), die Kennung des Systempartners und eine Produktionsart eindeutig bestimmt ist.

```
<complexType name="QSUpdateCertificationSystemMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type" />
    <element name="locationId" nillable="true" type="tns3:AN15Type" />
    <element name="locationType" nillable="true" type="tns3:CS840018Type" />
    <element name="auditScheme" nillable="true" type="xsd:int" />
    <element name="certificationBody" nillable="true" type="tns3:AN15Type" />
    <element name="locationTypeNew" nillable="true" type="tns3:CS840018Type" />
    <element name="certNr" nillable="true" type="tns3:AN99Type" />
    <element name="accreditedUntil" nillable="true" type="xsd:dateTime" />
    <element name="certificatedProducts" nillable="true" type="apachesoap:Vector" />
  </sequence>
</complexType>
```

```
<complexType name="QSCerificadedProductType">
  <sequence>
    <element name="productId" nillable="false" type="xsd:string" />
  </sequence>
</complexType>
```



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



```
<element name="area" nillable="true" type="xsd:double" />
</sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

questionerId: QS-Teilnehmerkennung des anfragenden Teilnehmers (Bündlers).

locationId: Die Standortnummer des Standortes.

locationType: Der Produktionsartenschlüssel des Standortes nach Schlüsseltabelle.

auditScheme: Zertifizierungssystem.

certificationBody: QS-Teilnehmerkennung des Prüfinstituts.

locationTypeNew: Der neue Produktionsartenschlüssel des Standortes nach Schlüsseltabelle.

certNr: Zertifikatsnummer (EurepGAP Registriernummer).

accreditedUntil: Gültigkeit des Zertifikats.

certificatedProducts: Liste von angebauten Kulturen.

productId: Kulturnummer.

area: Anbaufläche.

6.28.2 CQSSUpdateMsgReply - die Antwortdaten

Die Antwortdaten enthalten einen Zeitstempel als Bestätigung der durchgeführten Aktualisierung.

```
<complexType name="CQSSUpdateMsgReply">
  <sequence>
    <element name="timeStamp" nillable="true" type="xsd:dateTime"/>
  </sequence>
</complexType>
```

6.28.3 QSUpdateCertificationSystemMsg - Nachrichtenbeschreibung

Mit dieser Nachricht kann das Zertifizierungssystem geändert werden.

Mögliche Zertifizierungssysteme sind:

23. QS
24. QS-Gap
25. QM-Milch
26. IKB
27. GlobalGap
28. Codiplan
29. IKKB
30. AMA-GAP
31. Verpflichtungserklärer Salmonellenmonitoring



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



32. Verpflichtungserklärer Antibiotikamonitoring

33. Verpflichtungserklärer Salmonellenmonitoring und Antibiotikamonitoring

Für manche Zertifizierungssysteme, muss auch für AccreditedUntil ein Wert gesetzt sein, der nicht in der Vergangenheit liegt (sonst wird eine 045, "Date collision" Exception abgesetzt) und auch in CertNr muss eventuell ein Text eingetragen sein (sonst erfolgt eine 081, "No distinct certification body found" Exception). Für GlobalGap muss zusätzlich eine Liste der zertifizierten Kulturen übertragen werden.

Auch die Produktionsart muss je nach Zertifizierungssystem geändert werden, wenn die aktuelle Produktionsart nicht zum gewünschten Zertifizierungssystem passen.

6.29 Standort an Initiative Tierwohl übertragen

Vorbedingung: Der Bündler und der Standort sind dem System bekannt.

Nachbedingung: Der Standort wurde an die Initiative Tierwohl übertragen. Im Fehlerfall wird eine entsprechende Fehlermeldung zurückgeschickt.

6.29.1 QSRegisterTierwohlRequest – Übertragung eines Standortes

Ein Standort ist an dieser Stelle kein geographischer Begriff, sondern eine logische Größe, die durch eine Standortnummer (z. B. VVVO-Nummer) und eine Produktionsart eindeutig bestimmt ist.

```
<complexType name="QSRegisterTierwohlRequest">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type" />
    <element name="bstartId" nillable="true" type="tns3:CS840018Type" />
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

locationId: Die Standortnummer des Standortes.

bstartId: Der Produktionsartenschlüssel des Standortes nach Schlüsseltabelle.

6.29.2 registerTierwohlResponse - die Antwortdaten

Die Antwortdaten enthalten die übertragene Produktionsart.

```
<wsdl:message name="registerTierwohlResponse">
  <wsdl:part name="registerTierwohlReturn" type="tns3:CS840018Type" />
</wsdl:message>
```

6.29.3 QSRegisterTierwohlRequest - Nachrichtenbeschreibung

Mit dieser Nachricht kann ein Standort an die Datenbank der Initiative Tierwohl übertragen werden.

Die Standortnummer und Produktionsart bestimmen, welche Standortdatensätze an die Initiative Tierwohl übertragen werden.

Mögliche Produktionsarten sind:

2001, 2004, 2008, 2005, 2009, 2012, 2013



3001, 3004, 3005

Die Antwort enthält die Produktionsart, welche tatsächlich übertragen wurde.

Beispiel: Im Request ist die Produktionsart 2013 angegeben. Es existiert aber nur ein Standort mit der Produktionsart 2001 und einer mit der Produktionsart 2004. Somit wird der Standort mit der Produktionsart 2005 übertragen.

6.30 Ein Login für ein landwirtschaftliches Unternehmen erstellen

Vorbedingung: Der Bündler und das Unternehmen sind dem System bekannt.

Nachbedingung: Es wurde ein Logintoken erstellt. Im Fehlerfall wird eine entsprechende Fehlermeldung zurückgeschickt.

6.30.1 requestLoginRequest – Anfrage für ein Logintoken

```
<wsdl:message name="requestLoginRequest">  
  <wsdl:part name="in0" type="tns3:AN15Type" />  
</wsdl:message>
```

Die XML-Elemente haben folgende Semantik:

In0: Die QS ID des landwirtschaftlichen Unternehmens

6.30.2 requestLoginResponse - die Antwortdaten

Die Antwortdaten enthalten das Login und das Token.

```
<complexType name="QSLoginResponse">  
  <sequence>  
    <element name="clientId" nillable="false" type="tns3:AN15Type" />  
    <element name="token" nillable="false" type="tns3:AN99Type" />  
  </sequence>  
</complexType>
```

Die XML-Elemente haben folgende Semantik:

clientId: Der Benutzername (Login) ist die QS-ID des Unternehmens

token: Das Logintoken (Passwort) welches nur für den ersten Login genutzt werden kann.

6.30.3 requestLoginRequest - Nachrichtenbeschreibung

Mit dieser Nachricht kann ein Login für einen landwirtschaftlichen Betrieb erstellt werden. Das Login (QS-ID) und das Einmaltoken (Passwort) werden zurück gegeben. Der Landwirt muss beim ersten Login ein neues Passwort vergeben. Das Token hat eine Gültigkeit von 6 Monaten.

6.31 Ein Login für ein landwirtschaftliches Unternehmen erstellen und versenden

Vorbedingung: Der Bündler und das Unternehmen sind dem System bekannt. Beim Vertreter des landwirtschaftlichen Unternehmens ist eine Email Adresse hinterlegt.



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



Nachbedingung: Es wurde ein Logintoken erstellt. Im Fehlerfall wird eine entsprechende Fehlermeldung zurückgeschickt.

6.31.1 sendLoginRequest– Versand für ein Logintoken

```
<wsdl:message name="sendLoginRequest">
  <wsdl:part name="in0" type="tns3:AN15Type" />
</wsdl:message>
```

Die XML-Elemente haben folgende Semantik:

In0: Die QS ID des landwirtschaftlichen Unternehmens

6.31.2 sendLoginResponse - die Antwortdaten

Die Antwortdaten enthalten das Login und das Token.

```
<complexType name="QSLoginResponse">
  <sequence>
    <element name="clientId" nillable="false" type="tns3:AN15Type" />
    <element name="token" nillable="false" type="tns3:AN99Type" />
  </sequence>
</complexType>
```

Die XML-Elemente haben folgende Semantik:

clientId: Der Benutzername (Login) ist die QS-ID des Unternehmens

token: Das Logintoken (Passwort) welches nur für den ersten login genutzt werden kann.

6.31.3 sendLoginRequest - Nachrichtenbeschreibung

Mit dieser Nachricht kann ein Login für einen landwirtschaftlichen Betrieb erstellt werden. Das Login (QS-ID) und das Einmaltoken (Passwort) werden zurück gegeben. Der Landwirt muss beim ersten Login ein neues Passwort vergeben. Das Token hat eine Gültigkeit von 6 Monaten. Der Benutzername und das Token werden dem Vertreter des Unternehmens per Email mitgeteilt.

6.32 Die Liste aller landwirtschaftlichen Unternehmen ohne Login abfragen

Vorbedingung: Der Bündler ist dem System bekannt.

Nachbedingung: Es wurde ein Liste aller Unternehmen ohne Datenbankzugang erstellt. Im Fehlerfall wird eine entsprechende Fehlermeldung zurückgeschickt.

6.32.1 noLoginRequest – Abfrage der List

Keine Eingabeparameter notwendig

6.32.2 noLoginResponse - die Antwortdaten

Die Antwortdaten enthalten die QS IDs der Unternehmen ohne Datenbanklogin.

```
<complexType name="QSNoLoginResponse">
  <sequence>
```



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



```
<element name="clientId" maxOccurs="unbounded" minOccurs="0" nillable="true"
type="tns3:AN15Type" />
```

```
</sequence>
```

</complexType>Die XML-Elemente haben folgende Semantik:

clientId: Die QS-ID des Unternehmens

6.32.3 noLoginRequest - Nachrichtenbeschreibung

Mit dieser Nachricht kann eine Liste mit QS-IDs abgefragt werden, für die noch kein Datenbanklogin erstellt wurde.

A. Fehlermeldungen

Tabelle 5: Fehlermeldungen

Nummer	Fehlermeldung
003	Address information is ambiguous
005	Address could not be retrieved
007	Bundle could not be retrieved
008	Can't build bundle
013	ID length does not refer to a valid system member
014	No valid system member
015	No valid country code
016	No valid location number
023	Data collision: address is already in database
031	No such participants
032	Production line already in database
033	No such production line
034	Production line does not match with stored values
043	No such certification body
045	Date collision
050	No such verification report
053	No such location
055	No location adress given
057	No location number given
058	No country code given
059	Uncorrect length of location number
060	Location already deleted
061	Member already deleted
062	Certification body already attributive
063	Location is already bundled
070	No permission granted
071	No unique permission granted
080	Duplicate location identification found
081	No distinct certification body found
082	No valid animal number in GroupFlag
100	Error: please contact support@qs-platform.info



Abbildung 4: XML/SOAP-Fehlermeldung nach Standard

```
<?xml version="1.0" encoding="utf-8" ?>
- <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
- <soapenv:Body>
  - <soapenv:Fault>
    <faultcode>soapenv:Server.userException</faultcode>
    <faultstring>org.apache.soap.SOAPException: Bundle could not be retrieved</faultstring>
  - <detail>
    - <faultData xsi:type="ns1:SOAPException" xmlns:ns1="http://soap.apache.org">
      <faultCode xsi:type="xsd:string">007</faultCode>
      <message xsi:type="xsd:string">Bundle could not be retrieved</message>
      <targetException xsi:type="xsd:anyType" xsi:nil="true" />
      <rootException xsi:type="xsd:anyType" xsi:nil="true" />
      <cause xsi:nil="true" />
    </faultData>
    <ns2:hostname xmlns:ns2="http://xml.apache.org/axis/">Servername</ns2:hostname>
  </detail>
</soapenv:Fault>
</soapenv:Body>
</soapenv:Envelope>
```

7 Literatur

- [1] Harold, Means: XML in a Nutshell, (O'Reilly), 2015.
- [2] Seely, Sharkey: SOAP: Cross Platform Web Service Development using XML (Prentice Hall), 2002.
- [3] Snell et al.: Programming Web Services with SOAP (O'Reilly), 2002.
- [4] <http://www.gnupp.de>.
- [5] <http://www.apache.org>.
- [6] <http://www.xml.org>.
- [7] <http://www.w3.org>.
- [8] <http://www.gnupg.org>.
- [9] <http://www.w3.org/TR/SOAP-dsig>.



8 WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="urn:qs-stadat"
  xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:apacheSOAP="http://xml.apache.org/xml-soap"
  xmlns:impl="urn:qs-stadat" xmlns:intf="urn:qs-stadat"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns2="http://msg.soap.sla"
  xmlns:tns3="http://dtyp.soap.sla" xmlns:tns4="http://soap.apache.org"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:wsdlSOAP="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <wsdl:types>
    <schema targetNamespace="http://dtyp.soap.sla" xmlns="http://www.w3.org/2001/XMLSchema">
      <import namespace="http://schemas.xmlsoap.org/soap/encoding/" />

      <simpleType name="AN10Type">
        <restriction base="xsd:string">
          <maxLength value="10" />
        </restriction>
      </simpleType>

      <simpleType name="AN15Type">
        <restriction base="xsd:string">
          <maxLength value="15" />
        </restriction>
      </simpleType>

      <simpleType name="AN20Type">
        <restriction base="xsd:string">
          <maxLength value="20" />
        </restriction>
      </simpleType>

      <simpleType name="AN25Type">
        <restriction base="xsd:string">
          <maxLength value="25" />
        </restriction>
      </simpleType>

      <simpleType name="AN30Type">
        <restriction base="xsd:string">
          <maxLength value="30" />
        </restriction>
      </simpleType>

      <simpleType name="AN50Type">
        <restriction base="xsd:string">
          <maxLength value="50" />
        </restriction>
      </simpleType>

      <simpleType name="AN60Type">
        <restriction base="xsd:string">
          <maxLength value="60" />
        </restriction>
      </simpleType>

      <simpleType name="AN99Type">
        <restriction base="xsd:string">
          <maxLength value="99" />
        </restriction>
      </simpleType>

      <simpleType name="CS80004Type">
        <restriction base="xsd:string">
          <enumeration value="1" />
          <enumeration value="2" />
          <enumeration value="3" />
          <enumeration value="4" />
          <enumeration value="5" />
          <enumeration value="6" />
          <enumeration value="7" />
          <enumeration value="9" />
          <enumeration value="10" />
          <enumeration value="11" />
          <enumeration value="13" />
          <enumeration value="14" />
          <enumeration value="15" />
          <enumeration value="16" />
          <enumeration value="17" />
          <enumeration value="18" />
        </restriction>
    </schema>
  </wsdl:types>
</wsdl:definitions>
```



```
</simpleType>  
  
<simpleType name="CS840018Type">  
  <restriction base="xsd:string">  
    <enumeration value="20"/>  
    <enumeration value="30"/>  
    <enumeration value="31"/>  
    <enumeration value="32"/>  
    <enumeration value="33"/>  
    <enumeration value="34"/>  
    <enumeration value="35"/>  
    <enumeration value="36"/>  
    <enumeration value="37"/>  
    <enumeration value="38"/>  
    <enumeration value="39"/>  
    <enumeration value="40"/>  
    <enumeration value="41"/>  
    <enumeration value="50"/>  
    <enumeration value="51"/>  
    <enumeration value="52"/>  
    <enumeration value="53"/>  
    <enumeration value="60"/>  
    <enumeration value="61"/>  
    <enumeration value="70"/>  
    <enumeration value="71"/>  
    <enumeration value="72"/>  
    <enumeration value="73"/>  
    <enumeration value="74"/>  
    <enumeration value="75"/>  
    <enumeration value="76"/>  
    <enumeration value="77"/>  
    <enumeration value="78"/>  
    <enumeration value="79"/>  
    <enumeration value="80"/>  
    <enumeration value="81"/>  
    <enumeration value="82"/>  
    <enumeration value="83"/>  
    <enumeration value="84"/>  
    <enumeration value="86"/>  
    <enumeration value="90"/>  
    <enumeration value="91"/>  
    <enumeration value="92"/>  
    <enumeration value="93"/>  
    <enumeration value="100"/>  
    <enumeration value="301"/>  
    <enumeration value="304"/>  
    <enumeration value="450"/>  
    <enumeration value="460"/>  
    <enumeration value="470"/>  
    <enumeration value="701"/>  
    <enumeration value="702"/>  
    <enumeration value="703"/>  
    <enumeration value="801"/>  
    <enumeration value="802"/>  
    <enumeration value="840"/>  
    <enumeration value="880"/>  
    <enumeration value="901"/>  
    <enumeration value="902"/>  
    <enumeration value="904"/>  
    <enumeration value="1001"/>  
    <enumeration value="1002"/>  
    <enumeration value="1003"/>  
    <enumeration value="1004"/>  
    <enumeration value="1005"/>  
    <enumeration value="1006"/>  
    <enumeration value="1007"/>  
    <enumeration value="1008"/>  
    <enumeration value="1009"/>  
    <enumeration value="1010"/>  
    <enumeration value="1011"/>  
    <enumeration value="1012"/>  
    <enumeration value="1013"/>  
    <enumeration value="1014"/>  
    <enumeration value="1015"/>  
    <enumeration value="1016"/>  
    <enumeration value="1017"/>  
    <enumeration value="1018"/>  
    <enumeration value="1019"/>  
    <enumeration value="1020"/>  
    <enumeration value="1021"/>  
    <enumeration value="1022"/>  
    <enumeration value="1023"/>  
    <enumeration value="1024"/>
```



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



```
<enumeration value="1025"/>
<enumeration value="1026"/>
<enumeration value="1027"/>
<enumeration value="1028"/>
<enumeration value="1029"/>
<enumeration value="1030"/>
<enumeration value="1031"/>
<enumeration value="1320"/>
<enumeration value="2001"/>
<enumeration value="2002"/>
<enumeration value="2003"/>
<enumeration value="2004"/>
<enumeration value="2005"/>
<enumeration value="2006"/>
<enumeration value="2007"/>
<enumeration value="2008"/>
<enumeration value="2009"/>
<enumeration value="2010"/>
<enumeration value="2011"/>
<enumeration value="2012"/>
<enumeration value="2013"/>
<enumeration value="2014"/>
<enumeration value="2015"/>
<enumeration value="3001"/>
<enumeration value="3002"/>
<enumeration value="3003"/>
<enumeration value="3004"/>
<enumeration value="3005"/>
<enumeration value="3006"/>
<enumeration value="3007"/>
<enumeration value="3008"/>
<enumeration value="3009"/>
<enumeration value="3010"/>
<enumeration value="3011"/>
<enumeration value="3012"/>
<enumeration value="3013"/>
<enumeration value="3014"/>
<enumeration value="3015"/>
<enumeration value="3016"/>
<enumeration value="3017"/>
<enumeration value="3018"/>
<enumeration value="3019"/>
<enumeration value="3020"/>
<enumeration value="3021"/>
<enumeration value="3022"/>
<enumeration value="3023"/>
<enumeration value="3024"/>
<enumeration value="3025"/>
<enumeration value="3026"/>
<enumeration value="3027"/>
<enumeration value="3028"/>
<enumeration value="3029"/>
<enumeration value="3030"/>
<enumeration value="3031"/>
<enumeration value="3032"/>
<enumeration value="4001"/>
<enumeration value="4002"/>
<enumeration value="4003"/>
<enumeration value="4004"/>
<enumeration value="4005"/>
<enumeration value="4006"/>
<enumeration value="4007"/>
<enumeration value="4008"/>
<enumeration value="4009"/>
<enumeration value="4010"/>
<enumeration value="4011"/>
<enumeration value="4012"/>
<enumeration value="4013"/>
<enumeration value="4014"/>
<enumeration value="4015"/>
<enumeration value="5001"/>
<enumeration value="6001"/>
<enumeration value="6002"/>
<enumeration value="6003"/>
<enumeration value="6102"/>
<enumeration value="6103"/>
<enumeration value="7001"/>
<enumeration value="7002"/>
<enumeration value="7003"/>
<enumeration value="7004"/>
<enumeration value="7005"/>
<enumeration value="7006"/>
<enumeration value="7007"/>
```



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



```
<enumeration value="7008"/>
<enumeration value="7009"/>
<enumeration value="7010"/>
<enumeration value="7011"/>
<enumeration value="7012"/>
<enumeration value="7013"/>
<enumeration value="7014"/>
<enumeration value="7015"/>
<enumeration value="7016"/>
<enumeration value="7017"/>
<enumeration value="7018"/>
<enumeration value="7019"/>
<enumeration value="7020"/>
<enumeration value="7021"/>
<enumeration value="7022"/>
<enumeration value="7023"/>
<enumeration value="7024"/>
<enumeration value="7025"/>
<enumeration value="7026"/>
<enumeration value="7027"/>
<enumeration value="7028"/>
<enumeration value="7029"/>
<enumeration value="7030"/>
<enumeration value="7031"/>
<enumeration value="7032"/>
<enumeration value="7033"/>
<enumeration value="7034"/>
<enumeration value="7035"/>
<enumeration value="7036"/>
<enumeration value="7037"/>
<enumeration value="7038"/>
<enumeration value="7039"/>
<enumeration value="7040"/>
<enumeration value="7041"/>
<enumeration value="7042"/>
<enumeration value="7043"/>
<enumeration value="7044"/>
<enumeration value="7045"/>
<enumeration value="7046"/>
<enumeration value="7047"/>
<enumeration value="7048"/>
<enumeration value="7049"/>
<enumeration value="7050"/>
<enumeration value="7051"/>
<enumeration value="7052"/>
<enumeration value="7053"/>
<enumeration value="7054"/>
<enumeration value="7055"/>
<enumeration value="7056"/>
<enumeration value="7057"/>
<enumeration value="7058"/>
<enumeration value="7059"/>
<enumeration value="7060"/>
<enumeration value="7061"/>
<enumeration value="7062"/>
<enumeration value="7063"/>
<enumeration value="7064"/>
<enumeration value="7065"/>
<enumeration value="7066"/>
<enumeration value="7067"/>
<enumeration value="7068"/>
<enumeration value="7069"/>
<enumeration value="7070"/>
<enumeration value="7071"/>
<enumeration value="7072"/>
<enumeration value="7073"/>
<enumeration value="7074"/>
<enumeration value="7075"/>
<enumeration value="7076"/>
<enumeration value="7077"/>
<enumeration value="7078"/>
<enumeration value="7079"/>
<enumeration value="7080"/>
<enumeration value="7081"/>
<enumeration value="7082"/>
<enumeration value="7083"/>
<enumeration value="7084"/>
<enumeration value="7085"/>
<enumeration value="7086"/>
<enumeration value="7087"/>
<enumeration value="7088"/>
<enumeration value="7089"/>
<enumeration value="7090"/>
```

```

<enumeration value="7091"/>
<enumeration value="7092"/>
<enumeration value="7093"/>
<enumeration value="7094"/>
<enumeration value="7095"/>
<enumeration value="7096"/>
<enumeration value="7097"/>
<enumeration value="7098"/>
<enumeration value="7099"/>
<enumeration value="7100"/>
<enumeration value="7101"/>
<enumeration value="7102"/>
<enumeration value="7103"/>
<enumeration value="7104"/>
<enumeration value="7105"/>
<enumeration value="7106"/>
<enumeration value="7107"/>
<enumeration value="7108"/>
<enumeration value="7109"/>
<enumeration value="7110"/>
<enumeration value="7111"/>
<enumeration value="7112"/>
<enumeration value="7113"/>
<enumeration value="7114"/>
<enumeration value="7115"/>
<enumeration value="7116"/>
<enumeration value="7117"/>
<enumeration value="7118"/>
<enumeration value="7119"/>
<enumeration value="7120"/>
<enumeration value="7121"/>
<enumeration value="7122"/>
<enumeration value="7123"/>
<enumeration value="7124"/>
<enumeration value="7125"/>
<enumeration value="7126"/>
<enumeration value="7127"/>
</restriction>
</simpleType>

<simpleType name="CS840021Type">
  <restriction base="xsd:string">
    <enumeration value="2" />
    <enumeration value="3" />
    <enumeration value="4" />
    <enumeration value="5" />
    <enumeration value="6" />
    <enumeration value="7" />
    <enumeration value="8" />
    <enumeration value="9" />
    <enumeration value="10" />
    <enumeration value="11" />
    <enumeration value="12" />
    <enumeration value="13" />
    <enumeration value="14" />
    <enumeration value="15" />
    <enumeration value="16" />
    <enumeration value="22" />
    <enumeration value="23" />
  </restriction>
</simpleType>

<simpleType name="CS82009Type">
  <restriction base="xsd:string">
    <enumeration value="008" />
    <enumeration value="012" />
    <enumeration value="020" />
    <enumeration value="032" />
    <enumeration value="036" />
    <enumeration value="040" />
    <enumeration value="056" />
    <enumeration value="070" />
    <enumeration value="072" />
    <enumeration value="076" />
    <enumeration value="100" />
    <enumeration value="124" />
    <enumeration value="191" />
    <enumeration value="203" />
    <enumeration value="208" />
    <enumeration value="232" />
    <enumeration value="246" />
    <enumeration value="250" />
    <enumeration value="276" />
  </restriction>
</simpleType>

```

```

<enumeration value="300" />
<enumeration value="348" />
<enumeration value="352" />
<enumeration value="372" />
<enumeration value="376" />
<enumeration value="380" />
<enumeration value="392" />
<enumeration value="428" />
<enumeration value="434" />
<enumeration value="438" />
<enumeration value="440" />
<enumeration value="442" />
<enumeration value="492" />
<enumeration value="504" />
<enumeration value="516" />
<enumeration value="528" />
<enumeration value="554" />
<enumeration value="578" />
<enumeration value="616" />
<enumeration value="620" />
<enumeration value="642" />
<enumeration value="643" />
<enumeration value="703" />
<enumeration value="705" />
<enumeration value="710" />
<enumeration value="724" />
<enumeration value="752" />
<enumeration value="756" />
<enumeration value="788" />
<enumeration value="792" />
<enumeration value="804" />
<enumeration value="818" />
<enumeration value="826" />
<enumeration value="840" />
<enumeration value="862" />
<enumeration value="891" />
</restriction>
</simpleType>

<complexType name="CQSSAddressType">
  <sequence>
    <element name="postCode" nillable="true" type="tns3:AN10Type" />
    <element name="name" nillable="true" type="tns3:CQSSNameType" />
    <element name="town" nillable="true" type="tns3:AN50Type" />
    <element name="country" nillable="true" type="tns3:CS820009Type" />
    <element name="street" nillable="true" type="tns3:AN50Type" />
  </sequence>
</complexType>

<complexType name="CQSSNameType">
  <sequence>
    <element name="lastName" nillable="true" type="tns3:AN99Type" />
    <element name="dateOfBirth" nillable="true" type="xsd:date" />
    <element name="firstName" nillable="true" type="tns3:AN99Type" />
    <element name="salutation" nillable="true" type="tns3:AN10Type" />
    <element name="title" nillable="true" type="tns3:AN30Type" />
    <element name="role" nillable="true" type="tns3:AN99Type" />
  </sequence>
</complexType>

<complexType name="CQSSLocationType">
  <sequence>
    <element name="LocationCharacteristics" nillable="true"
      type="apachesoap:Vector" />
    <element name="com" nillable="true" type="tns3:CQSSCommunicationType" />
    <element name="representative" nillable="true" type="tns3:AN30Type" />
    <element name="locatedSince" nillable="true" type="xsd:date" />
    <element name="locationNumber" nillable="true" type="tns3:AN15Type" />
    <element name="certificationBody" nillable="true" type="tns3:AN15Type" />
    <element name="locationType" nillable="true" type="tns3:CS840018Type" />
    <element name="address" nillable="true" type="tns3:CQSSAddressType" />
    <element name="issuedBy" nillable="true" type="tns3:AN99Type" />
  </sequence>
</complexType>

<complexType name="CQSSCommunicationType">
  <sequence>
    <element name="phone" nillable="true" type="tns3:AN20Type" />
    <element name="mobile" nillable="true" type="tns3:AN20Type" />
    <element name="email" nillable="true" type="tns3:AN99Type" />
    <element name="fax" nillable="true" type="tns3:AN25Type" />
  </sequence>
</complexType>

```

```

<complexType name="QSBdLStortInfoType">
  <sequence>
    <element name="fax" nillable="true" type="tns3:AN99Type" />
    <element name="street" nillable="true" type="tns3:AN99Type" />
    <element name="country" nillable="true" type="tns3:AN99Type" />
    <element name="telephone" nillable="true" type="tns3:AN99Type" />
    <element name="city" nillable="true" type="tns3:AN99Type" />
    <element name="pob" nillable="true" type="tns3:AN99Type" />
    <element name="dateOfBirth" nillable="true" type="xsd:dateTime" />
    <element name="zipPob" nillable="true" type="tns3:AN10Type" />
    <element name="EMail" nillable="true" type="tns3:AN99Type" />
    <element name="firstName" nillable="true" type="tns3:AN99Type" />
    <element name="lastName" nillable="true" type="tns3:AN99Type" />
    <element name="zipStreet" nillable="true" type="tns3:AN10Type" />
  </sequence>
</complexType>

<complexType name="QSCheckListItem">
  <sequence>
    <element name="numbering" nillable="true" type="xsd:string" />
    <element name="name" nillable="true" type="xsd:string" />
    <element name="knockOut" nillable="true" type="xsd:string" />
    <element name="remark" nillable="true" type="xsd:string" />
    <element name="mark" nillable="true" type="xsd:string" />
    <element name="timeLimit" nillable="true" type="xsd:dateTime" />
  </sequence>
</complexType>

<complexType name="ExtCheckListItem">
  <sequence>
    <element name="id" nillable="true" type="xsd:int" />
    <element name="numbering" nillable="true" type="xsd:string" />
    <element name="name" nillable="true" type="xsd:string" />
    <element name="knockOut" nillable="true" type="xsd:string" />
    <element name="remark" nillable="true" type="xsd:string" />
    <element name="mark" nillable="true" type="xsd:string" />
    <element name="timeLimit" nillable="true" type="xsd:dateTime" />
  </sequence>
</complexType>

<complexType name="QSExtLocationBtartQueryType">
  <sequence>
    <element name="btartId" nillable="true" type="tns3:CS840018Type" />
    <element name="certifiedLocation" type="xsd:int" />
    <element name="description" nillable="true" type="tns3:AN99Type" />
    <element name="accreditedUntil" nillable="true" type="xsd:dateTime" />
  </sequence>
</complexType>

<complexType name="QSLastInspectionType">
  <sequence>
    <element name="inspectionId" nillable="true" type="xsd:int" />
    <element name="checkListId" nillable="true" type="xsd:int" />
    <element name="checkListName" nillable="true" type="xsd:string" />
    <element name="dateOfInspection" nillable="true" type="xsd:dateTime" />
    <element name="auditType" nillable="true" type="xsd:int" />
    <element name="LocationId" nillable="true" type="tns3:AN15Type" />
    <element name="LocationType" nillable="true" type="tns3:CS840018Type" />
    <element name="inspectedType" nillable="true" type="tns3:CS840018Type" />
    <element name="state" nillable="true" type="tns3:CS80004Type" />
  </sequence>
</complexType>

<complexType name="QSBtart">
  <sequence>
    <element name="btartId" nillable="true" type="tns3:CS840018Type" />
  </sequence>
</complexType>

<complexType name="QSPLacesType">
  <sequence>
    <element name="LocationType" nillable="true" type="tns3:CS840018Type" />
    <element name="stablePlaces" nillable="true" type="xsd:int" />
  </sequence>
</complexType>

<complexType name="QSBuildBundleLocationType">
  <sequence>
    <element name="LocationId" nillable="true" type="tns3:AN15Type" />
    <element name="LocationType" nillable="true" type="tns3:CS840018Type" />
  </sequence>
</complexType>

```




Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



```
<complexType name="QSBuILDBundleLocationErrorType">
  <sequence>
    <element name="LocationId" nillable="true" type="tns3:AN15Type" />
    <element name="LocationType" nillable="true" type="tns3:CS840018Type" />
    <element name="errorCode" nillable="true" type="xsd:int" />
  </sequence>
</complexType>

<complexType name="QSBdlRequestType">
  <sequence>
    <element name="LocationId" nillable="true" type="tns3:AN15Type" />
    <element name="bstartId" nillable="true" type="tns3:CS840018Type" />
    <element name="statusId" nillable="true" type="tns3:CS80004Type" />
  </sequence>
</complexType>

<complexType name="QSCerificadedProductType">
  <sequence>
    <element name="productId" nillable="false" type="xsd:string" />
    <element name="area" nillable="true" type="xsd:double" />
  </sequence>
</complexType>

</schema>

<schema targetNamespace="http://soap.apache.org" xmlns="http://www.w3.org/2001/XMLSchema">
  <import namespace="http://schemas.xmlsoap.org/soap/encoding/" />
  <complexType name="SOAPException">
    <sequence>
      <element name="faultCode" nillable="true" type="xsd:string" />
      <element name="message" nillable="true" type="xsd:string" />
      <element name="targetException" nillable="true" type="xsd:anyType" />
      <element name="rootException" nillable="true" type="xsd:anyType" />
    </sequence>
  </complexType>
  <element name="SOAPException" nillable="true" type="tns4:SOAPException" />
</schema>

<schema targetNamespace="http://xml.apache.org/xml-soap" xmlns="http://www.w3.org/2001/XMLSchema">
  <import namespace="http://schemas.xmlsoap.org/soap/encoding/" />
  <complexType name="Vector">
    <sequence>
      <element maxOccurs="unbounded" minOccurs="0" name="item"
        type="xsd:anyType" />
    </sequence>
  </complexType>
</schema>

<schema targetNamespace="http://msg.soap.sLa" xmlns="http://www.w3.org/2001/XMLSchema">
  <import namespace="http://schemas.xmlsoap.org/soap/encoding/" />
  <complexType name="QSDeleteLocationMsg">
    <sequence>
      <element name="questionerId" nillable="true" type="tns3:AN15Type" />
      <element name="LocationId" nillable="true" type="tns3:AN15Type" />
      <element name="bstartId" nillable="true" type="tns3:CS840018Type" />
      <element name="refDate" nillable="true" type="xsd:dateTime" />
    </sequence>
  </complexType>

  <element name="QSDeleteLocationMsg" nillable="true"
    type="tns2:QSDeleteLocationMsg" />

  <complexType name="QSLastInspectionMsg">
    <sequence>
      <element name="questionerId" nillable="true" type="tns3:AN15Type" />
      <element name="dateFrom" nillable="true" type="xsd:dateTime" />
      <element name="dateTo" nillable="true" type="xsd:dateTime" />
    </sequence>
  </complexType>

  <element name="QSLastInspectionMsg" nillable="true"
    type="tns2:QSLastInspectionMsg" />

  <complexType name="QSLastSentInspectionMsg">
    <sequence>
      <element name="questionerId" nillable="true" type="tns3:AN15Type" />
      <element name="dateFrom" nillable="true" type="xsd:dateTime" />
      <element name="dateTo" nillable="true" type="xsd:dateTime" />
    </sequence>
  </complexType>

  <element name="QSLastSentInspectionMsg" nillable="true"
    type="tns2:QSLastSentInspectionMsg" />
</schema>
```

```

        type="tns2:QSLastSentInspectionMsg" />
<complexType name="QSLastInspectionMsgReply">
  <sequence>
    <element name="inspections" nillable="true" type="apachesoap:Vector" />
  </sequence>
</complexType>

<element name="QSLastInspectionMsgReply" nillable="true"
  type="tns2:QSLastInspectionMsgReply" />

<complexType name="QSInspectionFromIdMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type" />
    <element name="inspectionId" nillable="true" type="xsd:int" />
  </sequence>
</complexType>

<element name="QSInspectionFromIdMsg" nillable="true"
  type="tns2:QSInspectionFromIdMsg" />

<complexType name="QSChangeLocationDataMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type" />
    <element name="LocationId" nillable="true" type="tns3:AN15Type" />
    <element name="LocationType" nillable="true" type="tns3:CS840018Type" />
    <element name="groupFlag" nillable="true" type="tns3:AN99Type" />
  </sequence>
</complexType>

<element name="QSChangeLocationDataMsg" nillable="true"
  type="tns2:QSChangeLocationDataMsg" />

<complexType name="QSChangeAttributeMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type" />
    <element name="LocationId" nillable="true" type="tns3:AN15Type" />
    <element name="LocationType" nillable="true" type="tns3:CS840018Type" />
    <element name="attribute" nillable="true" type="tns3:AN99Type" />
    <element name="value" nillable="true" type="tns3:AN99Type" />
  </sequence>
</complexType>

<element name="QSChangeAttributeMsg" nillable="true"
  type="tns2:QSChangeAttributeMsg" />

<complexType name="QSUpdateCertificationSystemMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type" />
    <element name="LocationId" nillable="true" type="tns3:AN15Type" />
    <element name="LocationType" nillable="true" type="tns3:CS840018Type" />
    <element name="auditScheme" nillable="true" type="xsd:int" />
    <element name="certificationBody" nillable="true" type="tns3:AN15Type" />
    <element name="LocationTypeNew" nillable="true" type="tns3:CS840018Type" />
    <element name="certNr" nillable="true" type="tns3:AN99Type" />
    <element name="accreditedUntil" nillable="true" type="xsd:dateTime" />
    <element name="certificatedProducts" nillable="true" type="apachesoap:Vector" />
  </sequence>
</complexType>

<element name="QSUpdateCertificationSystemMsg" nillable="true"
  type="tns2:QSUpdateCertificationSystemMsg" />

<complexType name="QSExtInspectionsInPeriodMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type" />
    <element name="LocationId" nillable="true" type="tns3:AN15Type" />
    <element name="LocationType" nillable="true" type="tns3:CS840018Type" />
    <element name="fromDate" nillable="true" type="xsd:dateTime" />
    <element name="toDate" nillable="true" type="xsd:dateTime" />
  </sequence>
</complexType>

<element name="QSExtInspectionsInPeriodMsg" nillable="true"
  type="tns2:QSExtInspectionsInPeriodMsg" />

<complexType name="QSExtInspectionsInPeriodMsgReply">
  <sequence>
    <element name="inspectionMsg" nillable="true" type="apachesoap:Vector" />
  </sequence>
</complexType>

<element name="QSExtInspectionsInPeriodMsgReply" nillable="true"

```

```

        type="tns2:QSExtInspectionsInPeriodMsgReply" />

<complexType name="QSExtInspectionMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type" />
    <element name="locationId" nillable="true" type="tns3:AN15Type" />
    <element name="locationType" nillable="true" type="tns3:CS840018Type" />
    <element name="dateOfInspection" nillable="true" type="xsd:dateTime" />
  </sequence>
</complexType>

<element name="QSExtInspectionMsg" nillable="true"
  type="tns2:QSExtInspectionMsg" />

<complexType name="QSExtInspectionMsgReply">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type" />
    <element name="locationType" nillable="true" type="tns3:CS840018Type" />
    <element name="inspectionId" nillable="true" type="xsd:int" />
    <element name="checkListId" nillable="true" type="xsd:int" />
    <element name="checkListName" nillable="true" type="xsd:string" />
    <element name="inspectedType" nillable="true" type="tns3:CS840018Type" />
    <element name="inspectionType" nillable="true" type="xsd:int" />
    <element name="dateOfInspection" nillable="true" type="xsd:dateTime" />
    <element name="certificationBody" nillable="true" type="tns3:AN15Type" />
    <element name="auditor" nillable="true" type="tns3:AN60Type" />
    <element name="state" nillable="true" type="tns3:CS80004Type" />
    <element name="percentage" type="xsd:double" />
    <element name="fromTime" type="xsd:dateTime" />
    <element name="toTime" type="xsd:dateTime" />
    <element name="inspectionDuration" type="xsd:double" />
    <element name="informant" type="xsd:string" />
    <element name="LackReportDate" nillable="true" type="xsd:dateTime" />
    <element name="LeftoverFeeding" nillable="true" type="xsd:string" />
    <element name="selfMixer" type="xsd:string" />
    <element name="betterments" nillable="true" type="xsd:string" />
    <element name="comment" nillable="true" type="xsd:string" />
    <element name="generalKo" nillable="true" type="xsd:boolean" />
    <element name="generalKoComment" nillable="true" type="xsd:string" />
    <element name="numberOfPlaces" nillable="true" type="apachesoap:Vector" />
    <element name="checkListItems" nillable="true" type="apachesoap:Vector" />
    <element name="addOnCheckLists" nillable="true"
      type="apachesoap:Vector" />
  </sequence>
</complexType>

<element name="QSExtInspectionMsgReply" nillable="true"
  type="tns2:QSExtInspectionMsgReply" />

<complexType name="QSLocationInfoMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type" />
    <element name="locationId" nillable="true" type="tns3:AN15Type" />
    <element name="locationType" nillable="true" type="tns3:CS840018Type" />
  </sequence>
</complexType>

<element name="QSLocationInfoMsg" nillable="true"
  type="tns2:QSLocationInfoMsg" />

<complexType name="QSCertBodyQueryMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type" />
    <element name="locationId" nillable="true" type="tns3:AN15Type" />
    <element name="locationType" nillable="true" type="tns3:CS840018Type" />
  </sequence>
</complexType>

<element name="QSCertBodyQueryMsg" nillable="true"
  type="tns2:QSCertBodyQueryMsg" />

<complexType name="QSCertBodyQueryMsgReply">
  <sequence>
    <element name="certificationBody" nillable="true" type="tns3:AN15Type" />
    <element name="address" nillable="true" type="tns3:CQSSAddressType" />
  </sequence>
</complexType>

<element name="QSCertBodyQueryMsgReply" nillable="true"
  type="tns2:QSCertBodyQueryMsgReply" />

<complexType name="QSLocationInfoMsgReply">
  <sequence>

```

```

        <element name="ownLocation" type="xsd:boolean" />
        <element name="pendingSince" nillable="true" type="dateTime" />
        <element name="signOffDate" nillable="true" type="dateTime" />
        <element name="bundleName" nillable="true" type="tns3:AN99Type" />
        <element name="state" nillable="true" type="tns3:CS80004Type" />
    </sequence>
</complexType>

<element name="QSLocationInfoMsgReply" nillable="true"
    type="tns2:QSLocationInfoMsgReply" />

<complexType name="QSBUILDBundleMsg">
    <sequence>
        <element name="questionerId" nillable="true" type="tns3:AN15Type" />
        <element name="bundleName" nillable="true" type="tns3:AN99Type" />
        <element name="Locations" nillable="true" type="apachesoap:Vector" />
    </sequence>
</complexType>

<element name="QSBUILDBundleMsg" nillable="true"
    type="tns2:QSBUILDBundleMsg" />

<complexType name="QSBUILDBundleMsgReply">
    <sequence>
        <element name="bundleId" nillable="true" type="xsd:int" />
        <element name="errorLocations" nillable="true" type="apachesoap:Vector" />
    </sequence>
</complexType>

<element name="QSBUILDBundleMsgReply" nillable="true"
    type="tns2:QSBUILDBundleMsgReply" />

<complexType name="CQSSUpdateCompAdrMsg">
    <sequence>
        <element name="clientId" nillable="true" type="tns3:AN15Type" />
        <element name="questionerId" nillable="true" type="tns3:AN15Type" />
        <element name="newValue" nillable="true" type="tns3:CQSSAddressType" />
    </sequence>
</complexType>

<element name="CQSSUpdateCompAdrMsg" nillable="true"
    type="tns2:CQSSUpdateCompAdrMsg" />

<complexType name="CQSSUpdateCompComMsg">
    <sequence>
        <element name="clientId" nillable="true" type="tns3:AN15Type" />
        <element name="questionerId" nillable="true" type="tns3:AN15Type" />
        <element name="oldValue" nillable="true"
            type="tns3:CQSSCommunicationType" />
        <element name="newValue" nillable="true"
            type="tns3:CQSSCommunicationType" />
    </sequence>
</complexType>

<element name="CQSSUpdateCompComMsg" nillable="true"
    type="tns2:CQSSUpdateCompComMsg" />

<complexType name="CQSSUpdateMsgReply">
    <sequence>
        <element name="timeStamp" nillable="true" type="xsd:dateTime" />
    </sequence>
</complexType>

<element name="CQSSUpdateMsgReply" nillable="true"
    type="tns2:CQSSUpdateMsgReply" />

<complexType name="QSDeLeteMemberMsg">
    <sequence>
        <element name="clientId" nillable="true" type="tns3:AN15Type" />
        <element name="questionerId" nillable="true" type="tns3:AN15Type" />
        <element name="refDate" nillable="true" type="xsd:dateTime" />
    </sequence>
</complexType>

<element name="QSDeLeteMemberMsg" nillable="true"
    type="tns2:QSDeLeteMemberMsg" />

<complexType name="CQSSRegisterMemberMsg">
    <sequence>
        <element name="address" nillable="true" type="tns3:CQSSAddressType" />
        <element name="Location" nillable="true" type="tns3:CQSSLocationType" />
        <element name="com" nillable="true" type="tns3:CQSSCommunicationType" />
        <element name="bundleId" nillable="true" type="tns3:AN15Type" />
    </sequence>

```

```

        <element name="representativeAddress" nillable="true"
            type="tns3:CQSSAddressType" />
        <element name="groupFlag" nillable="true" type="tns3:AN99Type" />
    </sequence>
</complexType>

<element name="CQSSRegisterMemberMsg" nillable="true"
    type="tns2:CQSSRegisterMemberMsg" />

<complexType name="CQSSRegisterMemberMsgReply">
    <sequence>
        <element name="qsState" nillable="true" type="tns3:CS80004Type" />
        <element name="clientId" nillable="true" type="tns3:AN15Type" />
        <element name="timeStamp" nillable="true" type="xsd:date" />
    </sequence>
</complexType>

<element name="CQSSRegisterMemberMsgReply" nillable="true"
    type="tns2:CQSSRegisterMemberMsgReply" />

<complexType name="CQSSRegisterCertificationSystemMsg">
    <sequence>
        <element name="address" nillable="true" type="tns3:CQSSAddressType" />
        <element name="location" nillable="true" type="tns3:CQSSLocationType" />
        <element name="com" nillable="true" type="tns3:CQSSCommunicationType" />
        <element name="bundleId" nillable="true" type="tns3:AN15Type" />
        <element name="representativeAddress" nillable="true"
            type="tns3:CQSSAddressType" />
        <element name="groupFlag" nillable="true" type="tns3:AN99Type" />
        <element name="auditScheme" nillable="true" type="xsd:int" />
        <element name="certNr" nillable="true" type="tns3:AN99Type" />
        <element name="accreditedUntil" nillable="true" type="xsd:date" />
        <element name="certificatedProducts" nillable="true" type="apachesoap:Vector" />
    </sequence>
</complexType>

<element name="CQSSRegisterCertificationSystemMsg" nillable="true"
    type="tns2:CQSSRegisterCertificationSystemMsg" />

<complexType name="CQSSAddLocationMsg">
    <sequence>
        <element name="clientId" nillable="true" type="tns3:AN15Type" />
        <element name="location" nillable="true" type="tns3:CQSSLocationType" />
        <element name="bundleId" nillable="true" type="tns3:AN15Type" />
        <element name="groupFlag" nillable="true" type="tns3:AN99Type" />
    </sequence>
</complexType>

<element name="CQSSAddLocationMsg" nillable="true"
    type="tns2:CQSSAddLocationMsg" />

<complexType name="CQSSAddLocationMsgReply">
    <sequence>
        <element name="qsState" nillable="true" type="tns3:CS80004Type" />
        <element name="clientId" nillable="true" type="tns3:AN15Type" />
        <element name="timeStamp" nillable="true" type="xsd:date" />
        <element name="locationId" nillable="true" type="tns3:AN15Type" />
    </sequence>
</complexType>

<element name="CQSSAddLocationMsgReply" nillable="true"
    type="tns2:CQSSAddLocationMsgReply" />

<complexType name="CQSSAddCertificationSystemMsg">
    <sequence>
        <element name="clientId" nillable="true" type="tns3:AN15Type" />
        <element name="location" nillable="true" type="tns3:CQSSLocationType" />
        <element name="bundleId" nillable="true" type="tns3:AN15Type" />
        <element name="groupFlag" nillable="true" type="tns3:AN99Type" />
        <element name="auditScheme" nillable="true" type="xsd:int" />
        <element name="certNr" nillable="true" type="tns3:AN99Type" />
        <element name="accreditedUntil" nillable="true" type="xsd:date" />
        <element name="certificatedProducts" nillable="true" type="apachesoap:Vector" />
    </sequence>
</complexType>

<element name="CQSSAddCertificationSystemMsg" nillable="true"
    type="tns2:CQSSAddCertificationSystemMsg" />

<complexType name="QSUpdateCertBodyMsg">
    <sequence>
        <element name="questionerId" nillable="true" type="tns3:AN15Type" />
        <element name="locationId" nillable="true" type="tns3:AN15Type" />
    </sequence>
</complexType>

```

```

        <element name="bstartId" nillable="true" type="tns3:CS840018Type" />
        <element name="validFrom" nillable="true" type="xsd:dateTime" />
        <element name="certificationBody" nillable="true" type="tns3:AN15Type" />
    </sequence>
</complexType>

<element name="QSUpdateCertBodyMsg" nillable="true"
    type="tns2:QSUpdateCertBodyMsg" />

<complexType name="CQSSUpdateLocAdrMsg">
    <sequence>
        <element name="questionerId" nillable="true" type="tns3:AN15Type" />
        <element name="oldValue" nillable="true" type="tns3:CQSSAddressType" />
        <element name="locationId" nillable="true" type="tns3:AN15Type" />
        <element name="bstartId" nillable="true" type="tns3:CS840018Type" />
        <element name="newValue" nillable="true" type="tns3:CQSSAddressType" />
    </sequence>
</complexType>

<element name="CQSSUpdateLocAdrMsg" nillable="true"
    type="tns2:CQSSUpdateLocAdrMsg" />

<complexType name="CQSSUpdateLocComMsg">
    <sequence>
        <element name="questionerId" nillable="true" type="tns3:AN15Type" />
        <element name="oldValue" nillable="true"
            type="tns3:CQSSCommunicationType" />
        <element name="locationId" nillable="true" type="tns3:AN15Type" />
        <element name="bstartId" nillable="true" type="tns3:CS840018Type" />
        <element name="newValue" nillable="true"
            type="tns3:CQSSCommunicationType" />
    </sequence>
</complexType>

<element name="CQSSUpdateLocComMsg" nillable="true"
    type="tns2:CQSSUpdateLocComMsg" />

<complexType name="QSUpdateBstartMsg">
    <sequence>
        <element name="questionerId" nillable="true" type="tns3:AN15Type" />
        <element name="bstartIdNew" nillable="true" type="tns3:CS840018Type" />
        <element name="locationId" nillable="true" type="tns3:AN15Type" />
        <element name="bstartIdOld" nillable="true" type="tns3:CS840018Type" />
    </sequence>
</complexType>

<element name="QSUpdateBstartMsg" nillable="true"
    type="tns2:QSUpdateBstartMsg" />

<complexType name="QSUpdateLocationMsg">
    <sequence>
        <element name="locationIdOld" nillable="true" type="tns3:AN15Type" />
        <element name="questionerId" nillable="true" type="tns3:AN15Type" />
        <element name="locationIdNew" nillable="true" type="tns3:AN15Type" />
        <element name="bstartId" nillable="true" type="tns3:CS840018Type" />
    </sequence>
</complexType>

<element name="QSUpdateLocationMsg" nillable="true"
    type="tns2:QSUpdateLocationMsg" />

<complexType name="QSBdLStortInfoMsg">
    <sequence>
        <element name="questionerId" nillable="true" type="tns3:AN15Type" />
        <element name="locationId" nillable="true" type="tns3:AN15Type" />
        <element name="bstartId" nillable="true" type="tns3:CS840018Type" />
    </sequence>
</complexType>

<element name="QSBdLStortInfoMsg" nillable="true"
    type="tns2:QSBdLStortInfoMsg" />

<complexType name="QSBdLStortInfoMsgReply">
    <sequence>
        <element name="stortInfo" nillable="true"
            type="tns3:QSBdLStortInfoType" />
        <element name="refDate" nillable="true" type="xsd:dateTime" />
    </sequence>
</complexType>

<element name="QSBdLStortInfoMsgReply" nillable="true"
    type="tns2:QSBdLStortInfoMsgReply" />

```



```
<complexType name="QSBdlRequestMsg">
  <sequence>
    <element name="bundleNr" type="xsd:int" />
    <element name="questionerId" nillable="true" type="tns3:AN15Type" />
  </sequence>
</complexType>

<element name="QSBdlRequestMsg" nillable="true"
  type="tns2:QSBdlRequestMsg" />

<complexType name="QSBdlRequestMsgReply">
  <sequence>
    <element name="bundles" nillable="true" type="apachesoap:Vector" />
  </sequence>
</complexType>

<element name="QSBdlRequestMsgReply" nillable="true"
  type="tns2:QSBdlRequestMsgReply" />

<complexType name="QSExtLocationQuery">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type" />
    <element name="btartId" nillable="true" type="tns3:CS840018Type" />
  </sequence>
</complexType>

<element name="QSExtLocationQuery" nillable="true"
  type="tns2:QSExtLocationQuery" />

<complexType name="QSExtLocationQueryReply">
  <sequence>
    <element name="certifiedLocation" type="xsd:int" />
    <element name="description" nillable="true" type="tns3:AN99Type" />
    <element name="accreditedUntil" type="xsd:dateTime" />
  </sequence>
</complexType>

<element name="QSExtLocationQueryReply" nillable="true"
  type="tns2:QSExtLocationQueryReply" />

<complexType name="QSExtLocationBtartQuery">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type" />
    <element name="btartId" nillable="true" type="tns3:CS840018Type" />
  </sequence>
</complexType>

<element name="QSExtLocationBtartQuery" nillable="true"
  type="tns2:QSExtLocationBtartQuery" />

<complexType name="QSExtLocationBtartQueryReply">
  <sequence>
    <element name="listOfBtartId" nillable="true" type="apachesoap:Vector" />
  </sequence>
</complexType>

<element name="QSExtLocationBtartQueryReply" nillable="true"
  type="tns2:QSExtLocationBtartQueryReply" />

<complexType name="QSCertificatedProductsQueryMsg">
  <sequence>
    <element name="questionerId" nillable="true" type="tns3:AN15Type" />
    <element name="locationId" nillable="true" type="tns3:AN15Type" />
    <element name="btartId" nillable="true" type="tns3:CS840018Type" />
  </sequence>
</complexType>

<element name="QSCertificatedProductsQueryMsg" nillable="true"
  type="tns2:QSCertificatedProductsQueryMsg" />

<complexType name="QSCertificatedProductsQueryMsgReply">
  <sequence>
    <element name="certificatedProducts" nillable="true" type="apachesoap:Vector" />
  </sequence>
</complexType>

<element name="QSCertificatedProductsQueryMsgReply" nillable="true"
  type="tns2:QSCertificatedProductsQueryMsgReply" />

<complexType name="QSRegisterTierwohlRequest">
  <sequence>
    <element name="locationId" nillable="true" type="tns3:AN15Type" />
  </sequence>
</complexType>
```

```

        <element name="bstartId" nillable="true" type="tns3:CS840018Type" />
    </sequence>
</complexType>

<element name="QSRegisterTierwohlRequest" nillable="true"
    type="tns2:QSRegisterTierwohlRequest" />

<complexType name="QSLoginResponse">
    <sequence>
        <element name="clientId" nillable="false" type="tns3:AN15Type" />
        <element name="token" nillable="false" type="tns3:AN99Type" />
    </sequence>
</complexType>

<element name="QSLoginResponse" nillable="true"
    type="tns2:QSLoginResponse" />

<complexType name="QSNoLoginResponse">
    <sequence>
        <element name="clientId" maxOccurs="unbounded" minOccurs="0" nillable="true" type="tns3:AN15Type" />
    </sequence>
</complexType>

<element name="QSNoLoginResponse" nillable="true"
    type="tns2:QSNoLoginResponse" />
</schema>
</wsdl:types>

<wsdl:message name="insertRequest">
    <wsdl:part name="in0" type="tns2:CQSSRegisterMemberMsg" />
</wsdl:message>

<wsdl:message name="insertRequest1">
    <wsdl:part name="in0" type="tns2:CQSSAddLocationMsg" />
</wsdl:message>

<wsdl:message name="insertRequest2">
    <wsdl:part name="in0" type="tns2:CQSSRegisterCertificationSystemMsg" />
</wsdl:message>

<wsdl:message name="insertRequest3">
    <wsdl:part name="in0" type="tns2:CQSSAddCertificationSystemMsg" />
</wsdl:message>

<wsdl:message name="insertResponse">
    <wsdl:part name="insertReturn" type="tns2:CQSSRegisterMemberMsgReply" />
</wsdl:message>

<wsdl:message name="insertResponse1">
    <wsdl:part name="insertReturn" type="tns2:CQSSAddLocationMsgReply" />
</wsdl:message>

<wsdl:message name="updateRequest">
    <wsdl:part name="in0" type="tns2:QSUpdateCertBodyMsg" />
</wsdl:message>

<wsdl:message name="updateRequest1">
    <wsdl:part name="in0" type="tns2:CQSSUpdateLocAdrMsg" />
</wsdl:message>

<wsdl:message name="updateRequest2">
    <wsdl:part name="in0" type="tns2:CQSSUpdateLocComMsg" />
</wsdl:message>

<wsdl:message name="updateRequest3">
    <wsdl:part name="in0" type="tns2:QSUpdateBstartMsg" />
</wsdl:message>

<wsdl:message name="updateRequest4">
    <wsdl:part name="in0" type="tns2:QSUpdateLocationMsg" />
</wsdl:message>

<wsdl:message name="updateRequest5">
    <wsdl:part name="in0" type="tns2:QSBUILDBundleMsg" />
</wsdl:message>

<wsdl:message name="updateRequest6">
    <wsdl:part name="in0" type="tns2:CQSSUpdateCompAdrMsg" />
</wsdl:message>

<wsdl:message name="updateRequest7">
    <wsdl:part name="in0" type="tns2:CQSSUpdateCompComMsg" />

```



```

</wsdl:message>

<wsdl:message name="updateRequest8">
  <wsdl:part name="in0" type="tns2:QSChangeLocationDataMsg" />
</wsdl:message>

<wsdl:message name="updateRequest9">
  <wsdl:part name="in0" type="tns2:QSChangeAttributeMsg" />
</wsdl:message>

<wsdl:message name="updateRequest10">
  <wsdl:part name="in0" type="tns2:QSUpdateCertificationSystemMsg" />
</wsdl:message>

<wsdl:message name="updateResponse">
  <wsdl:part name="updateReturn" type="tns2:CQSSUpdateMsgReply" />
</wsdl:message>

<wsdl:message name="updateResponse1">
  <wsdl:part name="updateReturn" type="tns2:CQSSUpdateMsgReply" />
</wsdl:message>

<wsdl:message name="updateResponse2">
  <wsdl:part name="updateReturn" type="tns2:CQSSUpdateMsgReply" />
</wsdl:message>

<wsdl:message name="updateResponse3">
  <wsdl:part name="updateReturn" type="tns2:CQSSUpdateMsgReply" />
</wsdl:message>

<wsdl:message name="updateResponse4">
  <wsdl:part name="updateReturn" type="tns2:CQSSUpdateMsgReply" />
</wsdl:message>

<wsdl:message name="updateResponse5">
  <wsdl:part name="updateReturn" type="tns2:QSBUILDBundleMsgReply" />
</wsdl:message>

<wsdl:message name="updateResponse6">
  <wsdl:part name="updateReturn" type="tns2:CQSSUpdateMsgReply" />
</wsdl:message>

<wsdl:message name="updateResponse7">
  <wsdl:part name="updateReturn" type="tns2:CQSSUpdateMsgReply" />
</wsdl:message>

<wsdl:message name="updateResponse8">
  <wsdl:part name="updateReturn" type="tns2:CQSSUpdateMsgReply" />
</wsdl:message>

<wsdl:message name="updateResponse9">
  <wsdl:part name="updateReturn" type="tns2:CQSSUpdateMsgReply" />
</wsdl:message>

<wsdl:message name="updateResponse10">
  <wsdl:part name="updateReturn" type="tns2:CQSSUpdateMsgReply" />
</wsdl:message>

<wsdl:message name="deLeteRequest">
  <wsdl:part name="in0" type="tns2:QSDeLeteLocationMsg" />
</wsdl:message>

<wsdl:message name="deLeteRequest1">
  <wsdl:part name="in0" type="tns2:QSDeLeteMemberMsg" />
</wsdl:message>

<wsdl:message name="deLeteResponse">
  <wsdl:part name="deLeteReturn" type="tns2:CQSSUpdateMsgReply" />
</wsdl:message>

<wsdl:message name="deLeteResponse1">
  <wsdl:part name="deLeteReturn" type="tns2:CQSSUpdateMsgReply" />
</wsdl:message>

<wsdl:message name="selectRequest">
  <wsdl:part name="in0" type="tns2:QSBdLStortInfoMsg" />
</wsdl:message>

<wsdl:message name="selectRequest1">
  <wsdl:part name="in0" type="tns2:QSBdLRequestMsg" />
</wsdl:message>

<wsdl:message name="selectRequest4">

```

```

    <wsdl:part name="in0" type="tns2:QSExtLocationBtartQuery" />
</wsdl:message>

<wsdl:message name="selectRequest5">
  <wsdl:part name="in0" type="tns2:QSExtLocationQuery" />
</wsdl:message>

<wsdl:message name="selectRequest6">
  <wsdl:part name="in0" type="tns2:QSLastInspectionMsg" />
</wsdl:message>

<wsdl:message name="selectRequest7">
  <wsdl:part name="in0" type="tns2:QSInspectionFromIdMsg" />
</wsdl:message>

<wsdl:message name="selectRequest8">
  <wsdl:part name="in0" type="tns2:QSLocationInfoMsg" />
</wsdl:message>

<wsdl:message name="selectRequest10">
  <wsdl:part name="in0" type="tns2:QSExtInspectionsInPeriodMsg" />
</wsdl:message>

<wsdl:message name="selectRequest11">
  <wsdl:part name="in0" type="tns2:QSLastSentInspectionMsg" />
</wsdl:message>

<wsdl:message name="selectRequest12">
  <wsdl:part name="in0" type="tns2:QSCertBodyQueryMsg" />
</wsdl:message>

<wsdl:message name="selectRequest13">
  <wsdl:part name="in0" type="tns2:QSCertificatedProductsQueryMsg" />
</wsdl:message>

<wsdl:message name="selectResponse">
  <wsdl:part name="selectReturn" type="tns2:QSBdLStortInfoMsgReply" />
</wsdl:message>

<wsdl:message name="selectResponse1">
  <wsdl:part name="selectReturn" type="tns2:QSBdLRequestMsgReply" />
</wsdl:message>

<wsdl:message name="selectResponse4">
  <wsdl:part name="selectReturn" type="tns2:QSExtLocationBtartQueryReply" />
</wsdl:message>

<wsdl:message name="selectResponse5">
  <wsdl:part name="selectReturn" type="tns2:QSExtLocationQueryReply" />
</wsdl:message>

<wsdl:message name="selectResponse6">
  <wsdl:part name="selectReturn" type="tns2:QSLastInspectionMsgReply" />
</wsdl:message>

<wsdl:message name="selectResponse7">
  <wsdl:part name="selectReturn" type="tns2:QSExtInspectionMsgReply" />
</wsdl:message>

<wsdl:message name="selectResponse8">
  <wsdl:part name="selectReturn" type="tns2:QSLocationInfoMsgReply" />
</wsdl:message>

<wsdl:message name="selectResponse10">
  <wsdl:part name="selectReturn" type="tns2:QSExtInspectionsInPeriodMsgReply" />
</wsdl:message>

<wsdl:message name="selectResponse11">
  <wsdl:part name="selectReturn" type="tns2:QSLastInspectionMsgReply" />
</wsdl:message>

<wsdl:message name="selectResponse12">
  <wsdl:part name="selectReturn" type="tns2:QSCertBodyQueryMsgReply" />
</wsdl:message>

<wsdl:message name="selectResponse13">
  <wsdl:part name="selectReturn" type="tns2:QSCertificatedProductsQueryMsgReply" />
</wsdl:message>

<wsdl:message name="inspectionReportRequest3">
  <wsdl:part name="in0" type="tns2:QSExtInspectionMsg" />
</wsdl:message>

```

```

<wsdl:message name="inspectionReportResponse3">
  <wsdl:part name="inspectionReportReturn" type="tns2:QSExtInspectionMsgReply" />
</wsdl:message>

<wsdl:message name="registerTierwohlRequest">
  <wsdl:part name="in0" type="tns2:QSRegisterTierwohlRequest" />
</wsdl:message>

<wsdl:message name="registerTierwohlResponse">
  <wsdl:part name="registerTierwohlReturn" type="tns3:CS840018Type" />
</wsdl:message>

<wsdl:message name="requestLoginRequest">
  <wsdl:part name="in0" type="tns3:AN15Type" />
</wsdl:message>

<wsdl:message name="requestLoginResponse">
  <wsdl:part name="requestLoginResponse" type="tns2:QSLoginResponse" />
</wsdl:message>

<wsdl:message name="sendLoginRequest">
  <wsdl:part name="in0" type="tns3:AN15Type" />
</wsdl:message>

<wsdl:message name="sendLoginResponse">
  <wsdl:part name="sendLoginResponse" type="tns2:QSLoginResponse" />
</wsdl:message>

<wsdl:message name="noLoginRequest">
</wsdl:message>

<wsdl:message name="noLoginResponse">
  <wsdl:part name="noLoginResponse" type="tns2:QSNologinResponse" />
</wsdl:message>

<wsdl:message name="SOAPException">
  <wsdl:part name="fault" type="tns4:SOAPException" />
</wsdl:message>

<wsdl:portType name="QSV2MsgProcessor">
  <wsdl:operation name="delete" parameterOrder="in0">
    <wsdl:input message="impl:deleteRequest" name="deleteRequest" />
    <wsdl:output message="impl:deleteResponse" name="deleteResponse" />
    <wsdl:fault message="impl:SOAPException" name="SOAPException" />
  </wsdl:operation>

  <wsdl:operation name="delete" parameterOrder="in0">
    <wsdl:input message="impl:deleteRequest1" name="deleteRequest1" />
    <wsdl:output message="impl:deleteResponse1" name="deleteResponse1" />
    <wsdl:fault message="impl:SOAPException" name="SOAPException" />
  </wsdl:operation>

  <wsdl:operation name="insert" parameterOrder="in0">
    <wsdl:input message="impl:insertRequest" name="insertRequest" />
    <wsdl:output message="impl:insertResponse" name="insertResponse" />
    <wsdl:fault message="impl:SOAPException" name="SOAPException" />
  </wsdl:operation>

  <wsdl:operation name="insert" parameterOrder="in0">
    <wsdl:input message="impl:insertRequest1" name="insertRequest1" />
    <wsdl:output message="impl:insertResponse1" name="insertResponse1" />
    <wsdl:fault message="impl:SOAPException" name="SOAPException" />
  </wsdl:operation>

  <wsdl:operation name="insert" parameterOrder="in0">
    <wsdl:input message="impl:insertRequest2" name="insertRequest2" />
    <wsdl:output message="impl:insertResponse" name="insertResponse" />
    <wsdl:fault message="impl:SOAPException" name="SOAPException" />
  </wsdl:operation>

  <wsdl:operation name="insert" parameterOrder="in0">
    <wsdl:input message="impl:insertRequest3" name="insertRequest3" />
    <wsdl:output message="impl:insertResponse1" name="insertResponse1" />
    <wsdl:fault message="impl:SOAPException" name="SOAPException" />
  </wsdl:operation>

  <wsdl:operation name="select" parameterOrder="in0">
    <wsdl:input message="impl:selectRequest6" name="selectRequest6" />
    <wsdl:output message="impl:selectResponse6" name="selectResponse6" />
    <wsdl:fault message="impl:SOAPException" name="SOAPException" />
  </wsdl:operation>

```

```

<wsdl:operation name="select" parameterOrder="in0">
  <wsdl:input message="impl:selectRequest7" name="selectRequest7" />
  <wsdl:output message="impl:selectResponse7" name="selectResponse7" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="select" parameterOrder="in0">
  <wsdl:input message="impl:selectRequest8" name="selectRequest8" />
  <wsdl:output message="impl:selectResponse8" name="selectResponse8" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="select" parameterOrder="in0">
  <wsdl:input message="impl:selectRequest10" name="selectRequest10" />
  <wsdl:output message="impl:selectResponse10" name="selectResponse10" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="select" parameterOrder="in0">
  <wsdl:input message="impl:selectRequest11" name="selectRequest11" />
  <wsdl:output message="impl:selectResponse11" name="selectResponse11" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="select" parameterOrder="in0">
  <wsdl:input message="impl:selectRequest12" name="selectRequest12" />
  <wsdl:output message="impl:selectResponse12" name="selectResponse12" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="update" parameterOrder="in0">
  <wsdl:input message="impl:updateRequest" name="updateRequest" />
  <wsdl:output message="impl:updateResponse" name="updateResponse" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="update" parameterOrder="in0">
  <wsdl:input message="impl:updateRequest1" name="updateRequest1" />
  <wsdl:output message="impl:updateResponse1" name="updateResponse1" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="update" parameterOrder="in0">
  <wsdl:input message="impl:updateRequest5" name="updateRequest5" />
  <wsdl:output message="impl:updateResponse5" name="updateResponse5" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="update" parameterOrder="in0">
  <wsdl:input message="impl:updateRequest6" name="updateRequest6" />
  <wsdl:output message="impl:updateResponse6" name="updateResponse6" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="update" parameterOrder="in0">
  <wsdl:input message="impl:updateRequest7" name="updateRequest7" />
  <wsdl:output message="impl:updateResponse7" name="updateResponse7" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="update" parameterOrder="in0">
  <wsdl:input message="impl:updateRequest8" name="updateRequest8" />
  <wsdl:output message="impl:updateResponse8" name="updateResponse8" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="update" parameterOrder="in0">
  <wsdl:input message="impl:updateRequest9" name="updateRequest9" />
  <wsdl:output message="impl:updateResponse9" name="updateResponse9" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="update" parameterOrder="in0">
  <wsdl:input message="impl:updateRequest10" name="updateRequest10" />
  <wsdl:output message="impl:updateResponse10" name="updateResponse10" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="update" parameterOrder="in0">
  <wsdl:input message="impl:updateRequest2" name="updateRequest2" />
  <wsdl:output message="impl:updateResponse2" name="updateResponse2" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

```



```
<wsdl:operation name="update" parameterOrder="in0">
  <wsdl:input message="impl:updateRequest3" name="updateRequest3" />
  <wsdl:output message="impl:updateResponse3" name="updateResponse3" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="update" parameterOrder="in0">
  <wsdl:input message="impl:updateRequest4" name="updateRequest4" />
  <wsdl:output message="impl:updateResponse4" name="updateResponse4" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="select" parameterOrder="in0">
  <wsdl:input message="impl:selectRequest" name="selectRequest" />
  <wsdl:output message="impl:selectResponse" name="selectResponse" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="select" parameterOrder="in0">
  <wsdl:input message="impl:selectRequest1" name="selectRequest1" />
  <wsdl:output message="impl:selectResponse1" name="selectResponse1" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="select" parameterOrder="in0">
  <wsdl:input message="impl:selectRequest4" name="selectRequest4" />
  <wsdl:output message="impl:selectResponse4" name="selectResponse4" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="select" parameterOrder="in0">
  <wsdl:input message="impl:selectRequest5" name="selectRequest5" />
  <wsdl:output message="impl:selectResponse5" name="selectResponse5" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="select" parameterOrder="in0">
  <wsdl:input message="impl:selectRequest13" name="selectRequest13" />
  <wsdl:output message="impl:selectResponse13" name="selectResponse13" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="inspectionReport" parameterOrder="in0">
  <wsdl:input message="impl:inspectionReportRequest3" name="inspectionReportRequest3" />
  <wsdl:output message="impl:inspectionReportResponse3" name="inspectionReportResponse3" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="registerTierwohl" parameterOrder="in0">
  <wsdl:input message="impl:registerTierwohlRequest" name="registerTierwohlRequest" />
  <wsdl:output message="impl:registerTierwohlResponse" name="registerTierwohlResponse" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="requestLogin" parameterOrder="in0">
  <wsdl:input message="impl:requestLoginRequest" name="requestLoginRequest" />
  <wsdl:output message="impl:requestLoginResponse" name="requestLoginResponse" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="sendLogin" parameterOrder="in0">
  <wsdl:input message="impl:sendLoginRequest" name="sendLoginRequest" />
  <wsdl:output message="impl:sendLoginResponse" name="sendLoginResponse" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>

<wsdl:operation name="noLogin">
  <wsdl:input message="impl:noLoginRequest" name="noLoginRequest" />
  <wsdl:output message="impl:noLoginResponse" name="noLoginResponse" />
  <wsdl:fault message="impl:SOAPException" name="SOAPException" />
</wsdl:operation>
</wsdl:portType>

<wsdl:binding name="rpcrouterSoapBinding" type="impl:QSV2MsgProcessor">
  <wsdlsoap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http" />

  <wsdl:operation name="delete">
    <wsdlsoap:operation soapAction="" />
    <wsdl:input name="deleteRequest">
      <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
        namespace="urn:qs-stadat" use="encoded" />
    </wsdl:input>
  </wsdl:operation>
</wsdl:binding>
```

```

<wsdl:output name="deLeteResponse">
  <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:qs-stadat" use="encoded" />
</wsdl:output>
</wsdl:operation>

<wsdl:operation name="deLete">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="deLeteRequest1">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="deLeteResponse1">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="insert">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="insertRequest">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="insertResponse">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="insert">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="insertRequest1">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="insertResponse1">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="insert">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="insertRequest2">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="insertResponse">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="insert">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="insertRequest3">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="insertResponse1">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="update">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="updateRequest">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="updateResponse">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="update">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="updateRequest1">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="updateResponse">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

```

```

</wsdl:input>
<wsdl:output name="updateResponse1">
  <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:qs-stadat" use="encoded" />
</wsdl:output>
</wsdl:operation>

<wsdl:operation name="update">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="updateRequest2">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="updateResponse2">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="update">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="updateRequest3">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="updateResponse3">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="update">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="updateRequest4">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="updateResponse4">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="update">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="updateRequest5">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="updateResponse5">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="update">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="updateRequest6">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="updateResponse6">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="update">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="updateRequest7">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="updateResponse7">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="update">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="updateRequest8">

```

```

        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:input>
    <wsdl:output name="updateResponse8">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:output>
</wsdl:operation>

<wsdl:operation name="update">
    <wsdlsoap:operation soapAction="" />
    <wsdl:input name="updateRequest9">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:input>
    <wsdl:output name="updateResponse9">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:output>
</wsdl:operation>

<wsdl:operation name="update">
    <wsdlsoap:operation soapAction="" />
    <wsdl:input name="updateRequest10">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:input>
    <wsdl:output name="updateResponse10">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:output>
</wsdl:operation>

<wsdl:operation name="select">
    <wsdlsoap:operation soapAction="" />
    <wsdl:input name="selectRequest">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:input>
    <wsdl:output name="selectResponse">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:output>
</wsdl:operation>

<wsdl:operation name="select">
    <wsdlsoap:operation soapAction="" />
    <wsdl:input name="selectRequest1">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:input>
    <wsdl:output name="selectResponse1">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:output>
</wsdl:operation>

<wsdl:operation name="select">
    <wsdlsoap:operation soapAction="" />
    <wsdl:input name="selectRequest4">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:input>
    <wsdl:output name="selectResponse4">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:output>
</wsdl:operation>

<wsdl:operation name="select">
    <wsdlsoap:operation soapAction="" />
    <wsdl:input name="selectRequest5">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:input>
    <wsdl:output name="selectResponse5">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
            namespace="urn:qs-stadat" use="encoded" />
    </wsdl:output>
</wsdl:operation>

<wsdl:operation name="select">
    <wsdlsoap:operation soapAction="" />

```



```

<wsdl:input name="selectRequest6">
  <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:qs-stadat" use="encoded" />
</wsdl:input>
<wsdl:output name="selectResponse6">
  <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:qs-stadat" use="encoded" />
</wsdl:output>
</wsdl:operation>

<wsdl:operation name="select">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="selectRequest7">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="selectResponse7">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="select">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="selectRequest8">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="selectResponse8">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="select">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="selectRequest10">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="selectResponse10">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="select">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="selectRequest11">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="selectResponse11">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="select">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="selectRequest12">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="selectResponse12">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="select">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="selectRequest13">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="selectResponse13">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="inspectionReport">

```

```

<wsdlsoap:operation soapAction="" />
<wsdl:input name="inspectionReportRequest3">
  <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:qs-stadat" use="encoded" />
</wsdl:input>
<wsdl:output name="inspectionReportResponse3">
  <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:qs-stadat" use="encoded" />
</wsdl:output>
</wsdl:operation>

<wsdl:operation name="registerTierwohl">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="registerTierwohlRequest">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="registerTierwohlResponse">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="requestLogin">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="requestLoginRequest">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="requestLoginResponse">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="sendLogin">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="sendLoginRequest">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="sendLoginResponse">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>

<wsdl:operation name="noLogin">
  <wsdlsoap:operation soapAction="" />
  <wsdl:input name="noLoginRequest">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:input>
  <wsdl:output name="noLoginResponse">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:qs-stadat" use="encoded" />
  </wsdl:output>
</wsdl:operation>
</wsdl:binding>

<wsdl:service name="QSV2MsgProcessorService">
  <wsdl:port binding="impl:rpcrouterSoapBinding" name="rpcrouter_secure">
    <wsdlsoap:address location="https://qs-plattform.de/axis/services/rpcrouter_secure"/>
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>

```



Qualitätssicherung. Vom Landwirt bis zur Ladentheke.



QS Qualität und Sicherheit GmbH

Geschäftsführer
Dr. Hermann-Josef Nienhoff

Schedestraße 1-3
53113 Bonn

Tel +49 228 35068-0
Fax +49 228 35068-10

info@q-s.de
www.q-s.de

Fotos: Fotolia