



Data Management
& Customer Care

Schnittstellendokumentation Web Service AVSMeldeschein

Stand: 01.10.2018

Inhaltsverzeichnis

1	Allgemeines.....	3
2	Sicherheit	4
2.1	TAG Erzeugung.....	4
2.2	Schnittstelle	4
2.3	Kurverwaltung	4
2.4	Benutzerid	4
2.5	Sessionid.....	5
2.6	Verarbeitung.....	5
2.6.1	Aktuell gültige Verarbeitungskennzeichen	5
2.7	Version	5
3	Schnittstellenbeschreibung.....	6
3.1	Schnittstellenformat.....	6
3.2	Operation holeMeldeschein:.....	8
3.2.1	Objekt holeMeldeschein	8
3.2.2	Bedeutung der Felder von Objekt „holeMeldeschein“	8
3.2.3	Objekt meldeschein.....	9
3.2.4	Bedeutung der Felder von Objekt „meldeschein“	11
3.3	Operation bucheMeldeschein.....	25
3.4	Operation druckeMeldeschein.....	26
3.4.1	druckeMeldescheineRequest	26
3.4.2	Bedeutung der Felder von Objekt „meldeschein“	26
3.4.3	druckeMeldescheineResponse	27
3.4.4	Bedeutung des Feldes von Objekt „ms-pdf“	27
3.5	Methode getKonfigurationsListe	27
3.5.1	Methode configuration-lists	28
3.5.2	Bedeutung der Felder von Objekt „configuration-lists“	29
4	Verhalten im Fehlerfall.....	30
5	Anwendungsfälle	32
6	SOAP-Request für Funktion „bucheMeldeschein“	39
7	SOAP-Request „holeMeldeschein“	42
7.1	Request holeMeldeschein mit Buchungsnummer	43
7.2	Request holeMeldeschein mit Meldescheinnummer	43
7.3	SOAP-Response „holeMeldeschein“	44
8	SOAP-Request „getKonfigurationsListe“	46
9	Beispiele.....	48
9.1	Request bucheMeldeschein Gruppenmeldeschein	48
9.2	Response Fehlerhafter Login	49
10	Support Anfragen	49
11	Dazu zugehörige Quellen:.....	49

1 Allgemeines

Dieses Dokument beschreibt den Datenaustausch über den Webservice „JMeldescheinWebservices“ des Meldescheinsystems der AVS GmbH in Bayreuth. Die Schnittstelle wird von der AVS als Webservice bereitgestellt. Er stellt den Punkt dar, an dem alle Transaktionen im AVSMeldeschein-System ausgelöst werden. Die Anforderungen an dieses Webservice werden üblicherweise durch ein technisches System gestellt.

Der Webservice erwartet ein vordefiniertes Request-Objekt, welches laut vorgegebener Schema-Definition mit Attributen gefüllt sein muss. Das Objekt wird nach allgemeiner Partner-Authorisierung validiert, welches über eine http Basic-Authentification erfolgt und zur Weiterverarbeitung an das Hintergrundsystem übergeben wird. Das Ergebnis wird dem aufrufenden technischen System in Form eines Response-Objekts geliefert. Schlägt eine Transaktion fehl, so wird dem Response-Objekt eine Fehlernachricht hinzugefügt.

Die Schnittstellenspezifikation wird dem Partner seitens AVS in Form einer WSDL-Datei übergeben.

Links

WS-URL

<https://meldeschein.avs.de/meldeschein-ws-prod/JMeldescheinWebservices>

WSDL

<https://meldeschein.avs.de/meldeschein-ws-prod/JMeldescheinWebservices?wsdl>

Schnittstellen-Links ohne Authentifizierung

<https://meldeschein.avs.de/meldeschein-ws-prod/wsdl/IJMeldescheinWebservices.wsdl>

https://meldeschein.avs.de/meldeschein-ws-prod/wsdl/meldeschein_webservices.xsd

https://meldeschein.avs.de/meldeschein-ws-prod/wsdl/meldeschein_buchen.xsd

https://meldeschein.avs.de/meldeschein-ws-prod/wsdl/meldeschein_datentypen.xsd

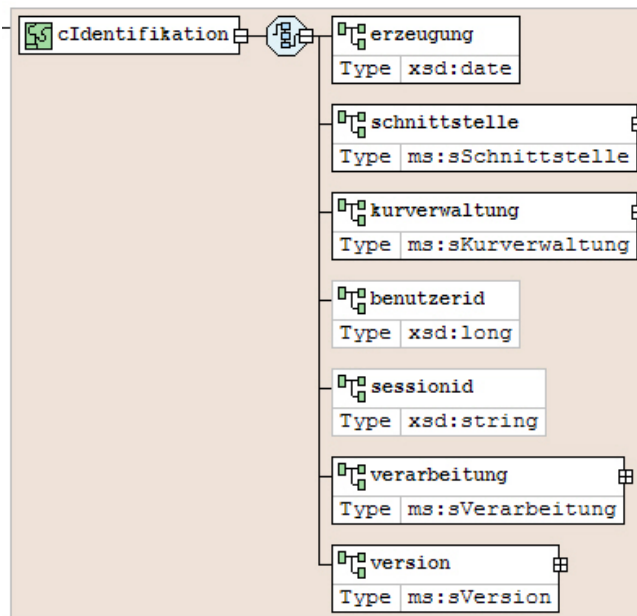
2 Sicherheit

Der Webservice ist via HTTP-Authentifikation gesichert. Die Zugangsdaten werden dem Partner durch die AVS mitgeteilt.

Die Authentifizierung erfolgt mittels http Basic-Authentication.

Die Autorisierung erfolgt durch den Identifikationsteil des SOAP Bodys.

Aufbau des Identifikationsteils:



Beispiel:

```
<identifikation>
  <erzeugung>2016-07-01</erzeugung>
  <schnittstelle>XXX</schnittstelle>
  <kurverwaltung>1</kurverwaltung>
  <benutzerid>1</benutzerid>
  <sessionid>string</sessionid>
  <verarbeitung>MS-HOLEN</verarbeitung>
  <version>string</version>
</identifikation>
```

2.1 TAG Erzeugung

Dieses Feld beinhaltet das aktuelle Datum.

Format gemäß XML-Schema-Typ **xsd:date** → Beispiel: **2016-07-12**

2.2 Schnittstelle

Hier ist der Benutzername des WS-Users zur Autorisierung einzutragen.

Der Benutzername wird im Rahmen der Erstanbindung mit dem jeweiligen Schnittstellenbediener vereinbart.

Der Benutzername entscheidet, welche Funktionen und Gemeinden aufgerufen werden können und welche nicht (Fehlercode 11900).

2.3 Kurverwaltung

Dieses Feld enthält die entsprechende ID der Kurverwaltung (Mandanten / OrtId) für die die folgende Aktion ausgeführt wird. Sie ist der interne Identifikationsschlüssel zum zugehörigen Ort.

2.4 Benutzerid

Dieses Attribut enthält optional eine Benutzerid. Hier kann der Benutzer / User des Hotelreservierungssystems übergeben werden.

2.5 Sessionid

Dieses Feld enthält einen vom Partner eindeutig generierten Sessionschlüssel, der zur Recherchezwecken verwendet werden kann. (HTTP-SessionId bei Webanwendungen).

2.6 Verarbeitung

Mit Hilfe dieses Attributes erfolgt die Differenzierung, welche Funktion über das entsprechende Verarbeitungskennzeichen angestoßen werden soll.

2.6.1 Aktuell gültige Verarbeitungskennzeichen

- **MS-HOLEN** (Abrufen eines kompletten Meldescheins)
- **BUCHEN** (Buchen eines Meldescheins)
- **MS-DRUCKEN** (Drucken eines Meldescheins)
- **CONFIG-L-*<bezeichner>*** (Abrufen von diversen Konfigurationslisten)

2.7 Version

Enthält die Versionsnummer des Webservices, die aktuelle Version wird von AVS im jeweiligen Response zurückgeliefert.

3 Schnittstellenbeschreibung

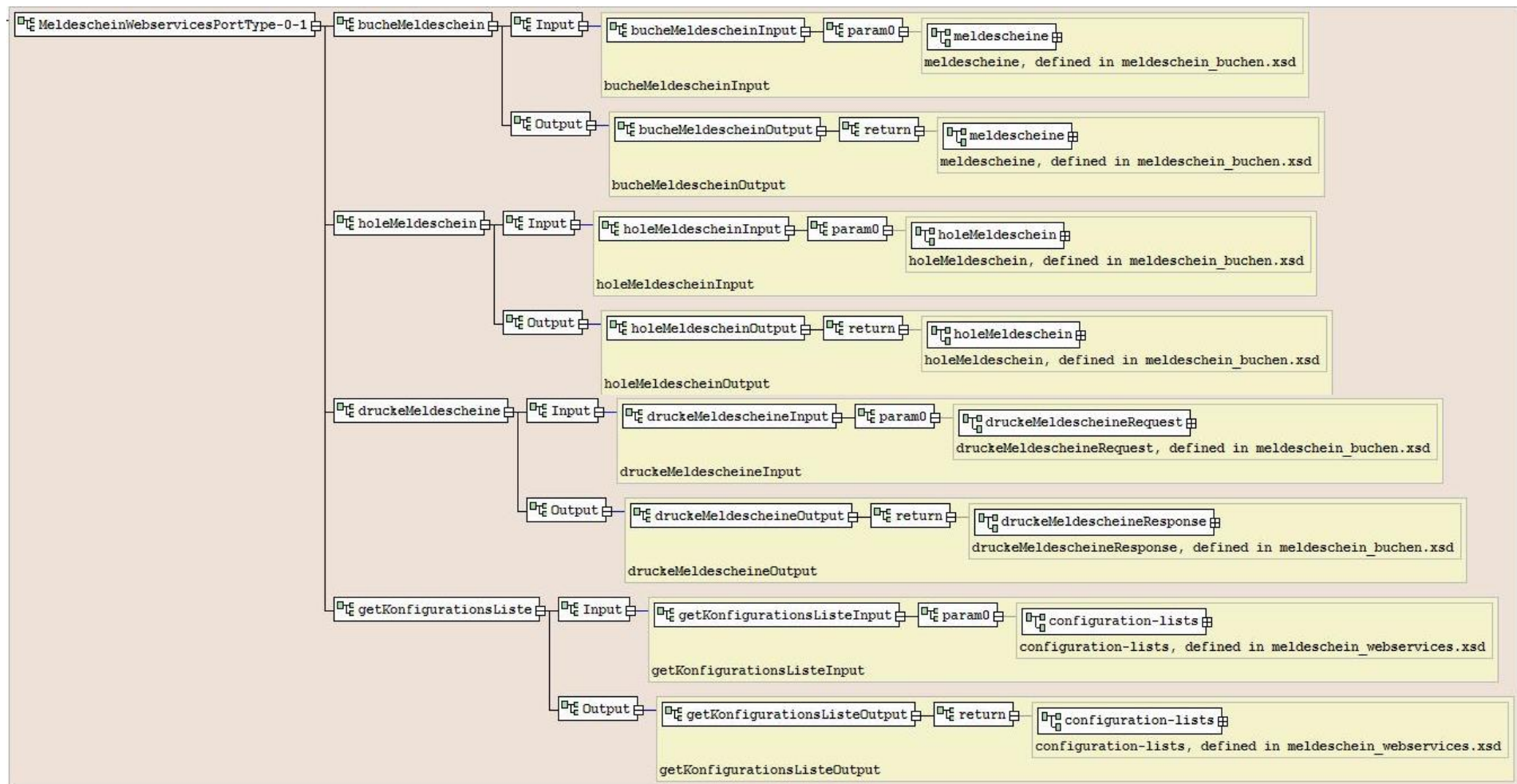
3.1 Schnittstellenformat

Die Schnittstelle wird von der AVS als Webservice bereitgestellt.

Die Schnittstelle umfasst folgende Funktionen

- **holeMeldeschein**
Damit können die akt. Meldescheindaten geholt werden. Erleichtert auch den Update-Prozess (siehe [bucheMeldeschein](#))
- **bucheMeldeschein**
Erzeugt oder ändert einen Meldeschein
- **druckeMeldeschein**
Erzeugt das Meldeschein-PDF
- **getKonfigurationsListe**
Ruft diverse Konfigurationslisten ab, um die anderen Webservice-Funktionen mit den korrekten ID's füllen zu können (Anrede-Id, Firma-Id etc.)

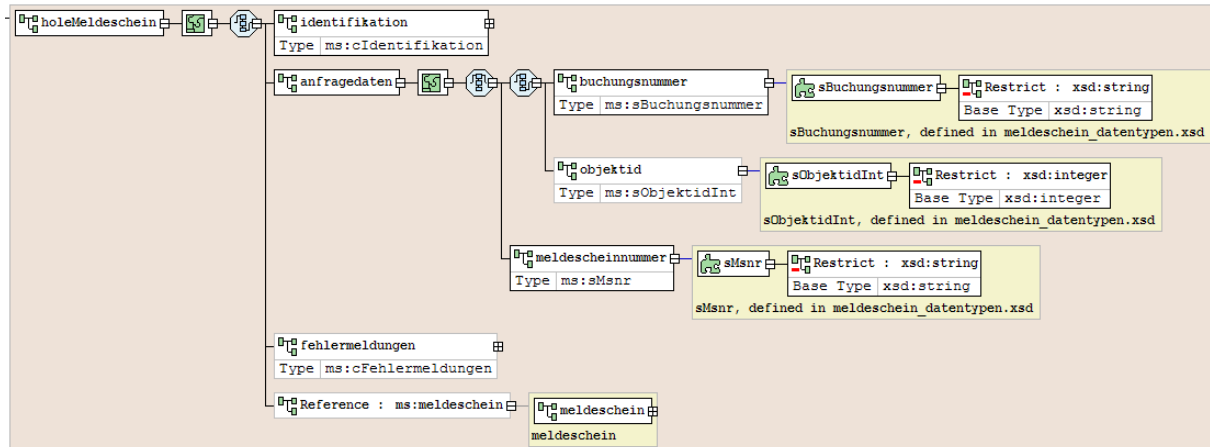
Schemabild der Webservice-Funktionen (WSDL-Porttypes):



3.2 Operation holeMeldeschein:

Eingang	Ausgang
holeMeldeschein	holeMeldeschein → meldeschein

3.2.1 Objekt holeMeldeschein



3.2.2 Bedeutung der Felder von Objekt „holeMeldeschein“

In diesem Objekt werden in einem Anfrageblock die Suchparameter gefüllt. Im Response-Objekt wird das Request-Element um die Elemente „fehlermeldungen“ und „meldeschein“ erweitert.

3.2.2.1 Anfragedaten

In den Anfragedaten kann der Meldeschein entweder anhand der Buchungsnummer + ObjektId **oder** der Meldescheinnummer geholt werden.

3.2.2.2 Buchungsnummer

Die vom Reservierungssystem vergebene Buchungsnummer (objektweit eindeutig!). Suchanfragen immer in Verbindung mit <objektid> .

3.2.2.3 ObjektId

Der eindeutige vom AVSMeldeschein vergebene Identifikationsschlüssel des Beherbungsobjektes. Suchanfragen immer in Verbindung mit <buchungsnummer> .

3.2.2.4 Meldescheinnummer

Die evtl. bekannte von AVSMeldeschein vergebene Meldescheinnummer.

3.2.3 Objekt meldeschein

Im Rückgabeelement bei erfolgreichen Anfrage-Request's wird das Element meldeschein geliefert. Schwarz umrahmte Elemente sind für die korrekte Buchung absolut notwendige Felder (Pflichtfelder), die restlichen Felder sind optional.



3.2.4 Bedeutung der Felder von Objekt „meldeschein“

3.2.4.1 Buchungsnummer

Enthält die pro „Mandant + Objekt“ eindeutige Buchungsnummer des Reservierungssystems.

Sonderfall manueller Meldeschein:

Als „Sonderprozess“ können manuelle Meldescheine nacherfasst werden. Dies ist üblicherweise nur durch über den Mandanten (Kurverwaltung, Tourist-Info) sinnvoll. Wenn Sie an diesem Prozess Interesse haben, wenden Sie sich bitte zunächst direkt an AVS.

Wird auf eine manuelle MS-Firma ein Meldeschein angelegt, muss hier die im AVS-System gepflegte manuelle MS-Nummer eingetragen werden.

3.2.4.2 kz_manuelle_ms

Wird, wie im Punkt 3.2.3.1 beim „Sonderfall“ beschrieben, eine manuelle MS-Nummer gebucht, kann über dieses *boolean*-Flag dem System dies direkt mitgeteilt werden. In diesem Fall muss nicht zwingend Firma- und ObjektId übergeben werden, diese Werte werden dann über die Konfiguration im AVSMeldeschein-System ermittelt.

3.2.4.3 Meldescheinnummer

Enthält die vom AVSMeldescheinsystem vergebene pro Mandant eindeutige Meldescheinnummer.

3.2.4.4 Firmald

Der eindeutige vom AVSMeldeschein vergebene Identifikationsschlüssel der Firma.

3.2.4.5 ObjektId

Der eindeutige vom AVSMeldeschein vergebene Identifikationsschlüssel des Beherbungsobjekts (Eine Firma kann ein oder mehrere Objekte besitzen (eigene oder vermittelte)).

3.2.4.6 Anreise

Das Anreisedatum der Hauptperson.

3.2.4.7 Abreise

Das Abreisedatum der Hauptperson.

3.2.4.8 Kategoriend

Enthält den eindeutigen, vom AVSMeldeschein pro Mandant vergebenen Identifikationsschlüssel der Personenkategorie (z.B. Erwachsener, Kind, Geschäftsreisender, Behindert). Die Konfigurationsliste (Schnittstellenkennung) erhalten Sie von der Kurverwaltung.

3.2.4.9 Anredeld

Enthält den eindeutigen vom AVSMeldeschein pro Mandant vergebenen Identifikationsschlüssel der Anrede.

3.2.4.10 Name

Name der Hauptperson.

3.2.4.11 Vorname

Vorname der Hauptperson.

3.2.4.12 Strasse

Strasse des Wohnsitzes der Hauptperson.

3.2.4.13 Hausnummer

Hausnummer des Wohnsitzes der Hauptperson.

3.2.4.14 Plz

PLZ des Wohnsitzes der Hauptperson.

3.2.4.15 Ort

Ort des Wohnsitzes der Hauptperson.

3.2.4.16 Land

Für die Hauptperson auf dem Meldeschein ist es möglich, abweichend von der Grundeinstellung ein Land (Wohnsitz der Hauptperson) zu übergeben:

Wird innerhalb des parent-TAG`s <land> der child-TAG <land-id> oder der child-TAG <land-iso> übergeben, dann wird geprüft, ob ein Wert innerhalb dieser child-TAG`s übergeben worden ist und ob dieser Wert gültig ist. Ist der Wert gültig, dann wird dieser übernommen.

Wird innerhalb des parent-TAG`s <land> der child-TAG <land-id> und der child-TAG <land-iso> übergeben, dann wird zuerst der child-TAG <land-id> geprüft. Ist der übergebene Wert gültig, dann wird dieser Wert übernommen und gespeichert. Wenn der übergebene Wert nicht korrekt bzw. leer übergeben wird, dann wird anschließend der child-TAG <land-iso> geprüft. Ist der übergebene Wert gültig, dann wird dieser Wert übernommen und gespeichert.

Sind beide child-TAG`s mit ungültigen Werten gefüllt oder werden leer übergeben, dann wird geprüft, ob der einzelne TAG <landid> vorhanden und gefüllt ist. Ist der Wert im TAG <landid> gültig, dann wird dieser übernommen, ansonsten wird automatisch der voreingestellte Default-Wert übernommen und gespeichert.

Die in den TAG`s <land> und <landid> zu übergebenden ID`s können über die Methode L-CONFIG-LAND abgerufen werden.

Beispiele für die Übergabe:

1)

```
<land>  
  <land-id>1</land-id>  
</land>
```

Die in der DB (und über die CONFIG-L-LAND abrufbare) Land-ID wird übernommen.

2)

```
<land>  
  <land-iso>DE</land-iso>  
</land>
```

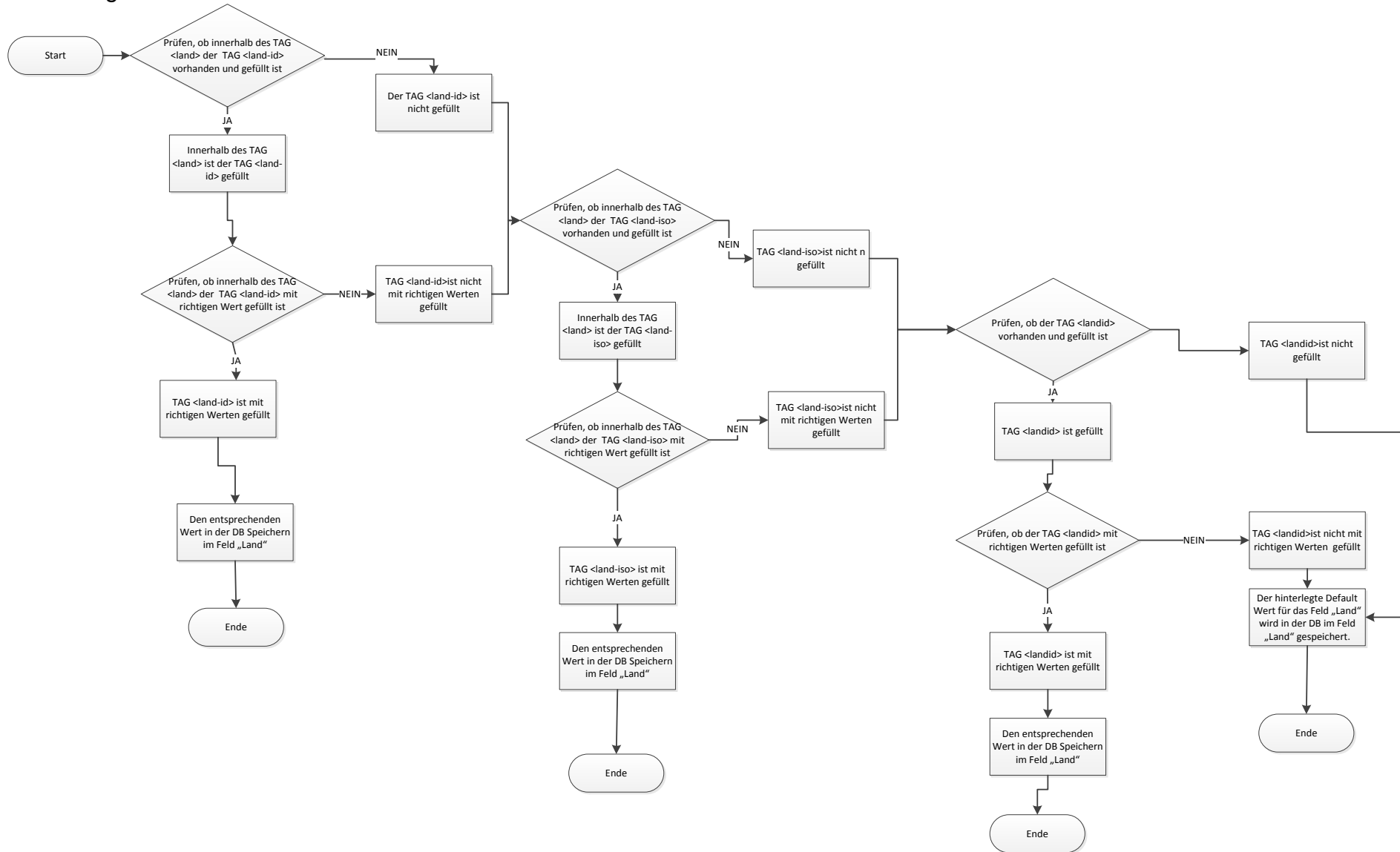
3)

```
<land>  
  <land-id>1</land-id>  
  <land-iso>DE</land-iso>  
</land>
```

4)

```
<landid>1</landid>
```

Ablaufdiagramm:



3.2.4.17 Staatsangehörigkeit

Wird innerhalb des **parent-TAG`s** **<staatsangehoerigkeit>** der **child-TAG <land-id>** **oder** der **child-TAG <land-iso>** übergeben, dann wird geprüft, ob ein Wert innerhalb dieser **child-TAG`s** übergeben worden ist und ob dieser Wert gültig ist. Ist der Wert gültig, dann wird dieser übernommen.

Wird innerhalb des **parent-TAG`s** **<staatsangehoerigkeit>** der **child-TAG <land-id>** **und** der **child-TAG <land-iso>** übergeben, dann wird zuerst der **child-TAG <land-id>** geprüft. Ist der übergebene Wert gültig, dann wird dieser Wert übernommen und gespeichert. Wenn der übergebene Wert nicht korrekt bzw. leer übergeben wird, dann wird anschließend der **child-TAG <land-iso>** geprüft. Ist der übergebene Wert gültig, dann wird dieser Wert übernommen und gespeichert.

Sind beide **child-TAG`s** mit ungültigen Werten gefüllt oder werden leer übergeben, dann wird geprüft, ob der einzelne **TAG <staatsangehoerigkeitid>** vorhanden und gefüllt ist. Ist der Wert im **TAG <staatsangehoerigkeitid>** gültig, dann wird dieser übernommen, ansonsten wird automatisch der voreingestellte Default-Wert übernommen und gespeichert.

Die in den **TAG`s <land>** **und <landid>** zu übergebenden ID`s können über die Methode L-CONFIG-LAND abgerufen werden.

Beispiele für die Übergabe::

1)

```
<staatsangehoerigkeit>  
  <land-id>1</land-id>  
</staatsangehoerigkeit>
```

2)

```
<staatsangehoerigkeit>  
  <land-iso>DE</land-iso>  
</staatsangehoerigkeit>
```

3)

```
<staatsangehoerigkeit>  
  <land-id>1</landid>  
  <land-iso>DE</land-iso>  
</staatsangehoerigkeit>
```

4)

```
<staatsangehoerigkeitid>1<staatsangehoerigkeitid>.
```

Copyright AVS GmbH_2018



3.2.4.18 Ausweisnummer (Personalausweisnummer)

Für den Hauptreisenden und die Begleitperson(en) kann die Ausweisnummer / Personalausweisnummer übergeben werden.

3.2.4.19 Weitere Angaben

Feld für weitere Angaben.

3.2.4.20 Kfzkennzeichen

KFZ-Kennzeichen des Anreisenden.

3.2.4.21 Geburtsdatum

Geburtsdatum der Hauptperson.

3.2.4.22 Betrag

Der vom AVSMeldeschein errechnete Betrag des Meldescheins.

Wenn der Betrag im TAG <meldeschein> steht, dann bezieht sich der Betrag auf den gesamten Meldeschein, nicht nur auf die Hauptperson.

Wenn der Betrag bei der Begleitperson oder Begleitgruppe steht, so ist es der Betrag nur von der jeweiligen Begleitperson bzw. Begleitgruppe.

3.2.4.23 AbrechnungstatusId

Der vom AVSMeldeschein vergebene Status des Meldescheins.

Folgende Werte / Status sind möglich:

- | | |
|---|-------------|
| 1 | Angelegt |
| 2 | Gedruckt |
| 3 | Abgerechnet |
| 4 | Storniert |
| 5 | Bezahlt |
| 6 | Ungültig |

3.2.4.24 Kategoriebedingte Zusatzfelder

Die Gemeinden / Mandanten können bei einzelnen Kategorien speziell vordefinierte Zusatzfelder eingerichtet haben, die auch über den Webservice gefüllt werden können.

Die jeweilige individuelle Konfiguration können Sie bei der Gemeinde / Mandant erfragen. Die jeweils eingerichteten Kategorien bzw. Felder sind auch in den Konfigurationsübersichten unter dem Punkt „Kategoriebedingte Zusatzfelder“ aufgelistet. Näheres dazu im PDF-Dokument „AVS Informationsschreiben - Schnittstellen Hotelprogramme“.

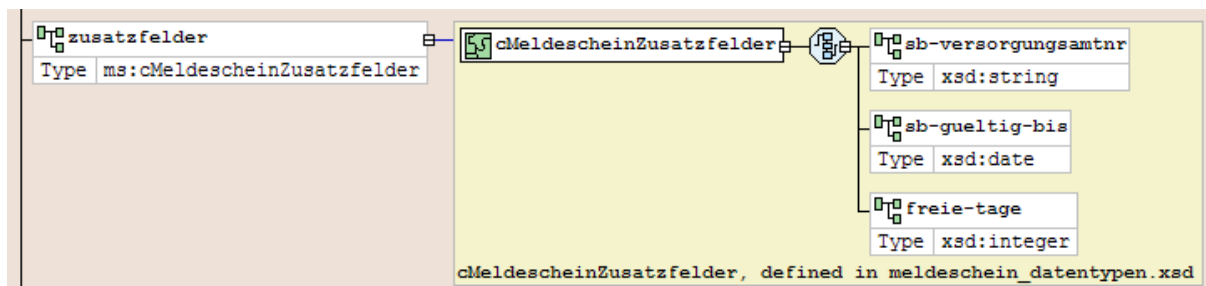
Kategoriebedingte Zusatzfelder sind grundsätzlich nur bei der Hauptperson möglich. Für Begleitpersonen und Begleitgruppen gibt es keine kategoriebedingten Zusatzfelder.

Es kann kein Meldeschein angelegt werden, in dem der Begleitperson oder der Begleitgruppe eine Kategorie mit einem kategoriebedingten Zusatzfeld zugeordnet wird. Dieser Meldeschein wird vom System abgelehnt bzw. eine Fehlermeldung zurückgeliefert.

Die Zusatzfelder werden innerhalb des **parent-TAG's** **<zusatzfelder>** übergeben. Innerhalb dieses **parent-TAG's** kann entweder ein oder mehrere Zusatzfelder übergeben werden. Wird ein **child-TAG** leer oder mit falschen Werten übergeben, dann wird der Wert nicht in den AVS-Meldeschein übernommen und das Feld dementsprechend leer angezeigt.

Folgende kategoriebedingte Zusatzfelder sind grundsätzlich möglich:

TAG	Beispiel
<sb-gueltig-bis>	<sb-gueltig-bis>2017-01-15</sb-gueltig-bis>
<sb-versorgungsamtnr>	<sb-versorgungsamtnr>ABC123</sb-versorgungsamtnr>
<firmenname>	<firmenname>Firma</firmenname>
<freie-tage>	<freie-tage>5</freie-tage>



Ein Beispiel-Request mit „allen“ Zusatzfeldern:

```
<?xml version="1.0"?>
<ms:meldescheine xmlns:ms="http://www.av.s.meldeschein.de/ns/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <identifikation>
    <erzeugung>2016-07-01</erzeugung>
    <schnittstelle>xxx</schnittstelle>
    <kurverwaltung>1</kurverwaltung>
    <benutzerid>1</benutzerid>
    <sessionid>string</sessionid>
    <verarbeitung>BUCHEN</verarbeitung>
  </identifikation>
  <zusatzfelder>
    <sb-gueltig-bis>2017-01-15</sb-gueltig-bis>
    <sb-versorgungsamtnr>ABC123</sb-versorgungsamtnr>
    <firmenname>Firma</firmenname>
    <freie-tage>5</freie-tage>
  </zusatzfelder>
</ms:meldescheine>
```

```

    <version>string</version>
</identifikation>
<ms:meldeschein>
  <buchungsnummer>30501</buchungsnummer>
  <firmaid>222</firmaid>
  <objektid>222</objektid>
  <anreise>2016-07-01</anreise>
  <abreise>2016-07-02</abreise>
  <kategorieid>2</kategorieid>
  <anredeid>1</anredeid>
  <name>Test-Name</name>
  <vorname>Test-Vorname</vorname>
  <strasse>Teststr.</strasse>
  <hausnummer>1</hausnummer>
  <plz>12345</plz>
  <ort>Teststadt</ort>
  <landid>1</landid>
  <staatsangehoerigkeitid>1</staatsangehoerigkeitid>
  <weitere_angaben>12345678</weitere_angaben>
  <ausweisnr>12345678ABC</ausweisnr>
  <kfzkennzeichen>BT TB-44</kfzkennzeichen>
  <geburtsdatum>1972-01-21</geburtsdatum>
  <zusatzfelder>
    <firmenname>Testfirma</firmenname>
    <freie-tage>5</freie-tage>
    <sb-gueltig-bis>2017-01-21</sb-gueltig-bis>
    <sb-versorgungsamtnr>ABC123</sb-versorgungsamtnr>
  </zusatzfelder>
</ms:meldeschein>
</ms:meldescheine>

```

Beispiel der Übergabe einer nicht gültigen Kategorie (kategoriebedingtes Zusatzfeld) bei der Begleitperson:

Request:

```

...
<begleitperson>
  <name>Test-Name</name>
  <vorname>Test-Vorname</vorname>
  <kategorieid>17</kategorieid>
  <anreise>2016-07-01</anreise>
  <abreise>2016-07-02</abreise>
</begleitperson>
...

```

Response:

...

```

<fehlermeldungen>
  <fehler>
    <code>10122</code>
    <beschreibung>Kategorie ungültig. Nur für den Hauptreisenden
      erlaubt</beschreibung>
    <bezug> Meldeschein-Buchungsnummer: BA1234422425 - Begleitperson: Test-
      Name</bezug>
  </fehler>
</fehlermeldungen>
...

```

Dieser Meldeschein wird vom System abgelehnt bzw. nicht gebucht.

3.2.4.25 Individuelle Zusatzfelder

Die Gemeinden / Mandanten können individuelle Zusatzfelder eingerichtet haben, die auch über den Webservice gefüllt werden können.

Die jeweilige individuelle Konfiguration dieser Zusatzfelder können Sie bei der Gemeinde / Mandant erfragen. Die jeweils eingerichteten Kategorien bzw. Felder sind auch in den Konfigurationsübersichten unter dem Punkt „Individuelle Zusatzfelder“ aufgelistet.

Nachfolgend ein Auszug aus dem PDF-Dokument mit beispielhaften Feldern:

Individuelle Felder Übersicht:		
Name	Typ	Max. Länge der Eingabe
Reisegrund	Combobox	Vorgegebene Werte
Vermerk	Text-Field	30

Vorgegebene Werte für das Feld Reisegrund

Wert
Messe
Kongress
Business
Freizeit

Näheres dazu ist auch im PDF-Dokument „AVS Informationsschreiben - Schnittstellen Hotelprogramme“ beschrieben.

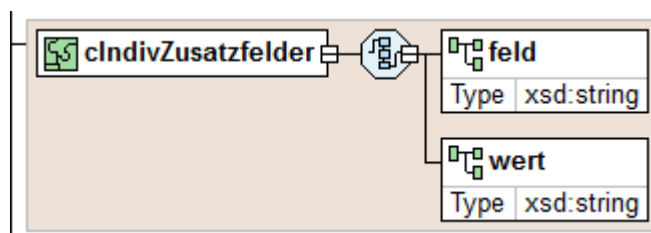
Individuelle Zusatzfelder sind grundsätzlich nur bei der Hauptperson möglich. Für Begleitpersonen gibt es keine individuellen Zusatzfelder. Es kann somit kein Meldeschein angelegt werden, in dem der Begleitperson eine Kategorie mit einem individuellen Zusatzfeld zugeordnet wird.

Die individuellen Zusatzfelder werden jeweils innerhalb des parent-TAG's `<indiv-zusatzfelder>` übergeben. Innerhalb dieses parent-TAG's kann jeweils immer nur ein individuelles Zusatzfeld übergeben werden. Eine Übergabe von mehreren individuellen Zusatzfeldern ist möglich, dies dann muss dann aber immer jeweils innerhalb von parent-TAG's erfolgen.

Bei den individuellen Zusatzfeldern gibt es zwei unterschiedliche „Typen“:

- Comboboxen; Eingabe / Übergabe von vorgegebenen / fest definierten Werten
- Text-Field; Eingabefeld „freier Text“

Typ	Feld (Beispiel)	Werte (Beispiele)	TAG's (Beispiele)
Combobox	Reisegrund	<ul style="list-style-type: none"> • Leer • Messe • Kongress • Freizeit 	<code><indiv-zusatzfelder></code> <code><feld>Reisegrund</feld></code> <code><wert>Kongress</wert></code> <code></indiv-zusatzfelder></code>
Text-Field	Vermerk	Stammgast	<code><indiv-zusatzfelder></code> <code><feld>Vermerk</feld></code> <code><wert>Stammgast</wert></code> <code></indiv-zusatzfelder></code>



Beim Typ „Comboboxen“ müssen die vorgegebenen / fest definierten Werte exakt so übergeben werden, wie sie im AVSMeldeschein angelegt bzw. auch in den „Konfigurationsübersichten“ dargestellt sind. Bei Abweichungen werden keine Meldescheine gebucht und eine Fehlermeldung zurückgeliefert. Es kann hier auch „Leer“ als Wert eingegeben bzw. übergeben werden.

Die Übergabe der Bezeichner in „Feld“ und „Werte“ sind „case-sensitive“, die Groß- und Kleinschreibung muss somit zwingend beachtet werden. Die Ausnahme ist der Wert im Text-Field. Dieser ist natürlich frei wählbar.

Im Response werden die gebuchten Felder und Werte jeweils wieder mit zurückgeliefert. Bei einem Update / Änderung des Meldescheines ist zu beachten, dass immer die bereits gebuchten Werte mit übergeben werden müssen.

Beispiel der Übergabe von Wert „Leer“ bei Typ „Combobox“:

```

...
<indiv-zusatzfelder>
  <feld>Reisgrund</feld>
  <wert></wert>
</indiv-zusatzfelder>
...

```

Beispiel der Übergabe von mehreren individuellen Zusatzfeldern:

```

...
<indiv-zusatzfelder>
  <feld>Reisgrund</feld>
  <wert>Kongress</wert>
</indiv-zusatzfelder>
<indiv-zusatzfelder>
  <feld>Vermerk</feld>
  <wert>Stammgast</wert>
</indiv-zusatzfelder>
...

```

Beispiele der Nichtbeachtung von „case-sensitive“:

```

...
<indiv-zusatzfelder>
  <feld>Reisgrund</feld>
  <wert>Kongress</wert>
</indiv-zusatzfelder>
...

```

```

...
<indiv-zusatzfelder>
  <feld>Reisgrund</feld>
  <wert>kongress</wert>
</indiv-zusatzfelder>
...

```

Bei der Nichtbeachtung von „case-sensitive“ kann das Feld / der Wert nicht erkannt werden. Es erfolgt keine Buchung des Meldescheines, es wird im Response eine entsprechende Fehlermeldung zurückgegeben:

```

...
<fehlermeldungen>
  <fehler>
    <code>10128</code>
    <beschreibung>Zusatzfelder falsch übergeben.</beschreibung>
    <bezug>Meldeschein-Buchungsnummer: BN123456</bezug>
  </fehler>
</fehlermeldungen>

```

Beispiel von fehlerhaftem Wert bei Typ „Combobox“:

```
...
<indiv-zusatzfelder>
  <feld>Reisegrund</feld>
  <wert>Irgendwas</wert>
</indiv-zusatzfelder>
...
```

„Irgendwas“ ist nicht als gültiger Wert definiert / hinterlegt. Der Wert kann nicht erkannt werden. Es erfolgt keine Buchung des Meldescheines, es wird im Response eine entsprechende Fehlermeldung zurückgegeben:

```
...
<fehlermeldungen>
  <fehler>
    <code>10128</code>
    <beschreibung>Zusatzfelder falsch übergeben.</beschreibung>
    <bezug>Meldeschein-Buchungsnummer: BN123456</bezug>
  </fehler>
</fehlermeldungen>
```

Ein (kompletter) Beispiel-Request mit einem individuellen Zusatzfeld:

```
<?xml version="1.0"?>
<ms:meldeschein xmlns:ms="http://www.av.s.meldeschein.de/ns/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <identifikation>
    <erzeugung>2018-09-01</erzeugung>
    <schnittstelle>xxx</schnittstelle>
    <kurverwaltung>1</kurverwaltung>
    <benutzerid>1</benutzerid>
    <sessionid>string</sessionid>
    <verarbeitung>BUCHEN</verarbeitung>
    <version>string</version>
  </identifikation>
  <ms:meldeschein>
    <buchungsnummer>BN0501</buchungsnummer>
    <firmaid>123</firmaid>
    <objektid>123</objektid>
    <anreise>2018-10-01</anreise>
    <abreise>2018-10-02</abreise>
    <kategorieid>2</kategorieid>
    <anredeid>1</anredeid>
    <name>Test-Name</name>
    <vorname>Test-Vorname</vorname>
    <strasse>Teststr.</strasse>
    <hausnummer>1</hausnummer>
    <plz>12345</plz>
```

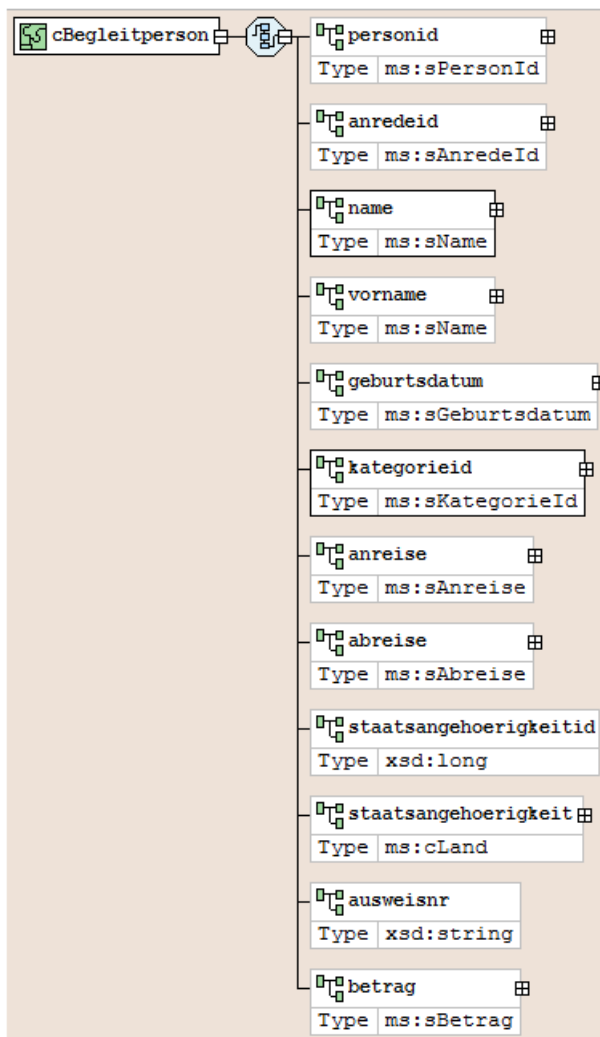
```

<ort>Teststadt</ort>
<landid>1</landid>
<staatsangehoerigkeitid>1</staatsangehoerigkeitid>
<weitere_angaben>12345678</weitere_angaben>
<ausweisnr>12345678ABC</ausweisnr>
<kfzkennzeichen>XX XX-44</kfzkennzeichen>
<geburtsdatum>1972-01-21</geburtsdatum>
<indiv-zusatzfelder>
  <feld>Reisegrund</feld>
  <wert>Kongress</wert>
</indiv-zusatzfelder>
</ms:meldeschein>
</ms:meldescheine>

```

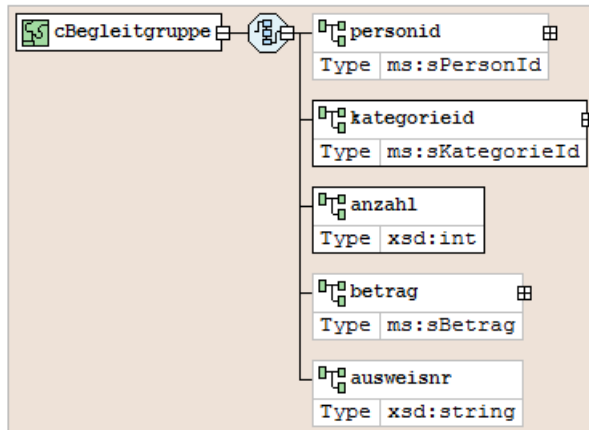
3.2.4.26 Begleitperson

Objekt vom Datentyp cBegleitperson, welches die Informationen zu Begleitpersonen enthält.



3.2.4.27 Begleitgruppe (Gruppenmeldeschein)

Objekt vom Datentyp cBegleitgruppe, welches die Informationen zu Begleitgruppen enthält.



Wichtig:

Die Anlage (Buchen) eines Gruppenmeldescheins ist nur möglich, wenn dieser Meldescheintyp bei den jeweiligen Mandaten freigeschaltet ist.

Bei der Anlage eines Gruppenmeldescheines ist zu beachten, dass die Mindestanzahl an Personen eingehalten werden muss, sonst ist keine Meldescheinanlage möglich bzw. es wird eine Fehlermeldung zurückgeliefert.

3.2.4.28 Übernachtungsentgelt / Bettensteuer

Die Gemeinden, die das Zusatzmodul „Übernachtungsentgelt / Bettensteuer“ aktiviert haben, kann das "Übernachtungsentgelt" über die Schnittstelle übergeben werden.

Tag	Beispiel
<code><ue-e-gelt></code>	<code><ue-e-gelt>130.50</ue-e-gelt></code>

Es ist ein Numerisches Feld, Eingabe eines „Betrages / Währung“, bis zu zwei Nachkommastellen.

Es gibt eine Limitierung auf maximal 5-stellige Beträge ("Vorkommastellen"), um „Fehleingaben abzufedern“.

3.3 Operation bucheMeldeschein

Request Element	Response Element
-----------------	------------------

meldescheine	meldescheine
--------------	--------------

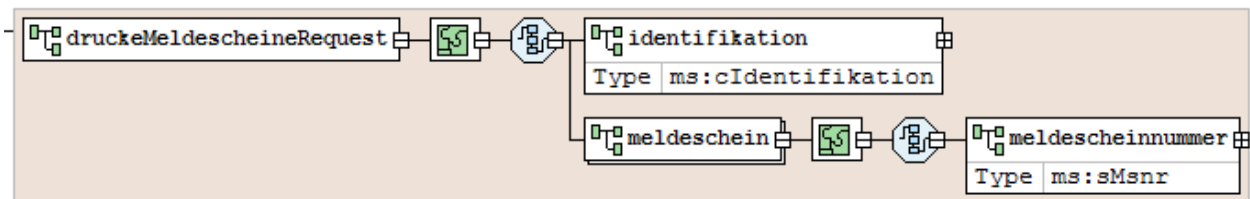
Objekt meldeschein

Siehe unter der Methode [holeMeldeschein](#).

3.4 Operation druckeMeldeschein

Request Element	Response Element
druckeMeldescheinRequest	druckeMeldescheinResponse

3.4.1 druckeMeldescheinRequest



Beispiel:

```

<p1:druckeMeldescheinRequest
  xmlns:p1="http://www.avsmeldeschein.de/ns/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <identifikation>
    <erzeugung>2016-07-01</erzeugung>
    <schnittstelle>xxx</schnittstelle>
    <kurverwaltung>1</kurverwaltung>
    <benutzerid>1</benutzerid>
    <sessionid>string</sessionid>
    <verarbeitung>MS-DRUCKEN</verarbeitung>
    <version>string</version>
  </identifikation>
  <!--Element meldeschein, maxOccurs=unbounded-->
  <meldeschein>
    <meldescheinnummer>100</meldescheinnummer>
  </meldeschein>
  <meldeschein>
    <meldescheinnummer>101</meldescheinnummer>
  </meldeschein>
</p1:druckeMeldescheinRequest>

```

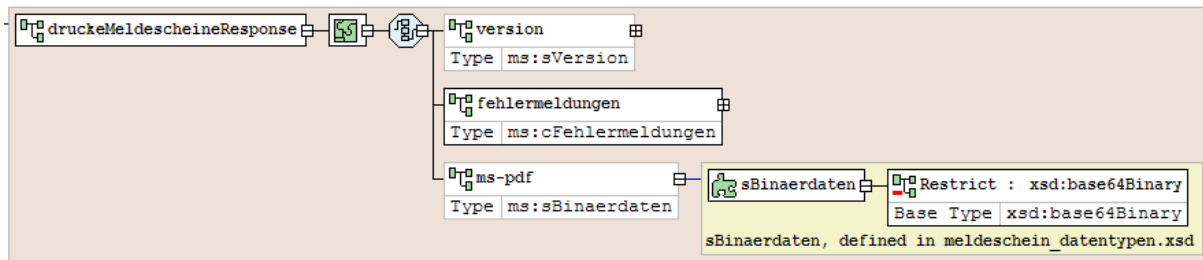
3.4.2 Bedeutung der Felder von Objekt „meldeschein“

In diesem Objekt wird der oder die Meldeschein/-e anhand der vom AVSMeldeschein vergebenen Meldescheinnummer identifiziert.

3.4.2.1 Meldescheinnummer

Die vom AVSMeldeschein vergebene Meldescheinnummer.

3.4.3 druckeMeldescheineResponse



Beispiel:

```
<?xml version="1.0"?>
<SOAP-ENV:Envelope xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Body>
    <druckeMeldescheineResponse xmlns="http://www.avs.meldeschein.de/ns/"
    <version xmlns="">1.2.06-H18</version>
    <fehlermeldungen xmlns="">
      <fehler>
        <code>10001</code>
        <beschreibung>Meldescheine-Drucken erfolgreich</beschreibung>
      </fehler>
    </fehlermeldungen>
    <ms-pdf xmlns="">JVBERi0xLjMKJaqrK0KNCAwIG9iago8PCA.....</ms-pdf>
  </druckeMeldescheineResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

3.4.4 Bedeutung des Feldes von Objekt „ms-pdf“

Im Tag `<ms-pdf>` vom Typ `xsd:base64Binary` wird im Response die serialisierte Form des PDFs zurückgegeben.

Ein Java-Webservice-Client würde für diesen Typ z.B. ein byte-Array erhalten, welches einfach als Datei abgespeichert werden kann.

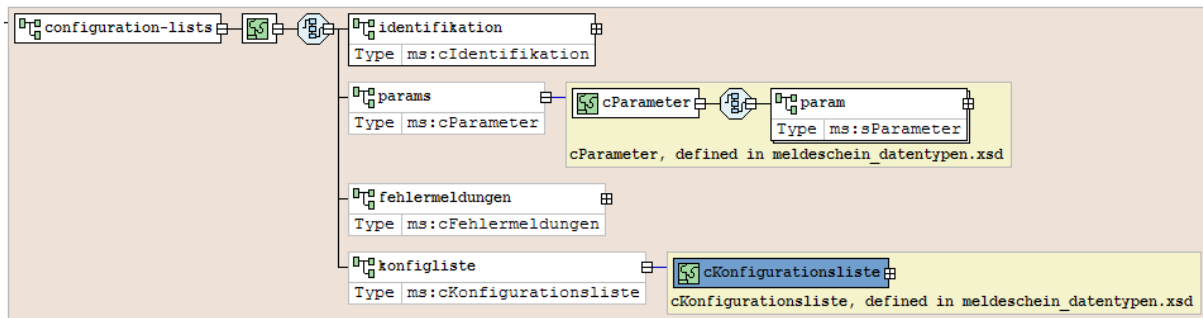
3.5 Methode getKonfigurationsListe

Request Element	Response Element
configuration-lists	configuration-lists

Das Tag `<verarbeitung>` bestimmt, welche Liste angefordert wird:

- CONFIG-L-ANREDE Anreden
- CONFIG-L-LAND Länder
- CONFIG-L-KAT Meldeschein-Kategorien

3.5.1 Methode configuration-lists



Im Response wir die jeweilige Konfigurationsliste zurückgegeben.

Beispiel Konfigurationsliste „Land“ Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns="http://www.avs.meldeschein.de/ns/">
  <soapenv:Header/>
  <soapenv:Body>
    <ns:configuration-lists>
      <identifikation>
        <erzeugung>2016-07-01</erzeugung>
        <schnittstelle>xxx</schnittstelle>
        <kurverwaltung>1</kurverwaltung>
        <benutzerid>1</benutzerid>
        <sessionid>string</sessionid>
        <verarbeitung>CONFIG-L-LAND</verarbeitung>
      </identifikation>
    </ns:configuration-lists>
  </soapenv:Body>
</soapenv:Envelope>
```

Beispiel Response (gekürzte Version):

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:configuration-lists xmlns:ns2="http://www.avs.meldeschein.de/ns/">
      <identifikation>
        <erzeugung>2016-07-01</erzeugung>
        <schnittstelle>xxx</schnittstelle>
        <kurverwaltung>1</kurverwaltung>
        <benutzerid>1</benutzerid>
        <sessionid>string</sessionid>
        <verarbeitung>CONFIG-L-LAND</verarbeitung>
      </identifikation>
    </ns2:configuration-lists>
  </S:Body>
</S:Envelope>
```

```

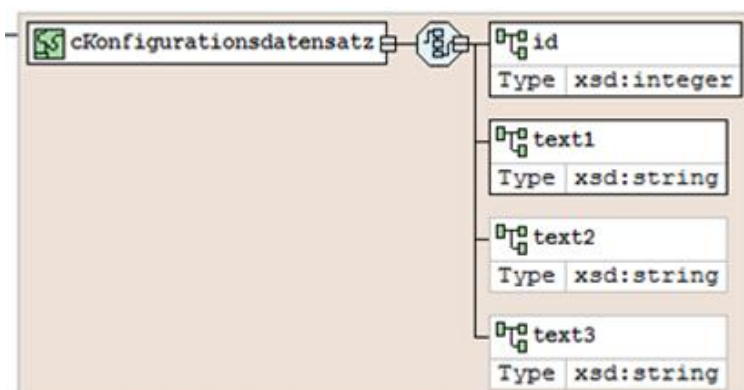
</identifikation>
  <fehlermeldungen>
    <fehler>
      <code>10001</code>
      <beschreibung>ConfigurationLists-Abfrage erfolgreich</beschreibung>
    </fehler>
  </fehlermeldungen>
  <konfigliste>
    ...
    <konfigdatensatz>
      <id>1</id>
      <text1>Deutschland</text1>
      <text2>deutsch</text2>
      <text3>DE</text3>
    </konfigdatensatz>
    <konfigdatensatz>
      <id>40</id>
      <text1>Arabische Länder Asiens</text1>
      <text2>arabisch</text2>
      <text3>AE</text3>
    </konfigdatensatz>
    ...
  </konfigliste>
</ns2:configuration-lists>
</S:Body>
</S:Envelope>

```

3.5.2 Bedeutung der Felder von Objekt „configuration-lists“

In dieser allgemeinen Funktion zum Abholen diverser Konfigurationslisten wird im Verarbeitungskennzeichen des Identifikations-Blocks bestimmt, welche Konfigurationsliste angefragt wird. Weiterführende Parametrisierung der entsprechenden Konfigurationsliste kann über die Objekte ‚params‘ erfolgen.

3.5.2.1 Rückgabe-Element konfigliste



Beispiel:

```
<konfigliste>
  <konfigdatensatz>
    <id>123</id>
    <text1>string1</text1>
    <!--Element text2 is optional-->
    <text2>string2</text2>
  </konfigdatensatz>
  <konfigdatensatz>
    <id>234</id>
    <text1>string3</text1>
    <!--Element text2 is optional-->
    <text2>string4</text2>
  </konfigdatensatz>
  <konfigdatensatz>
    <id>345</id>
    <text1>string5</text1>
    <!--Element text2 is optional-->
    <text2>string6</text2>
  </konfigdatensatz>
</konfigliste>
```

Jeder Konfigurationslisten-Datensatz gibt hier die jeweilige Id (Anrede-Id...) sowie einen bezeichnenden Text wieder. Für einen alternativen Text bzw. weiterführende Informationen des Datensatzes ist das TAG <text2>-Feld vorgesehen (wird evtl. noch erweitert).

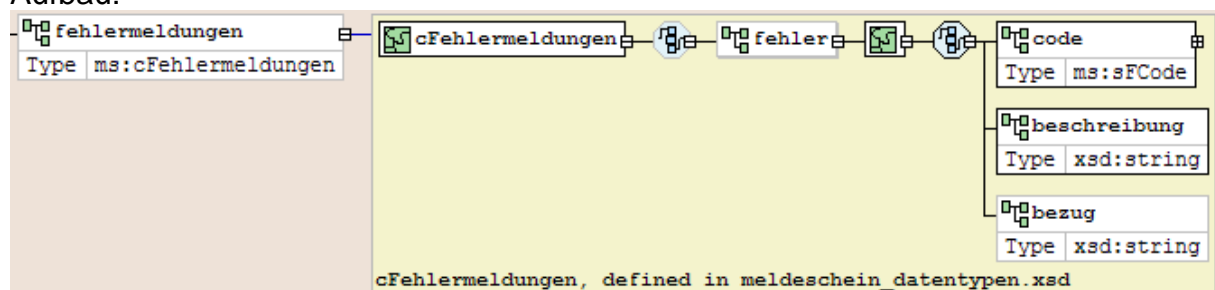
Bei Abfragen der Kategorien-Liste (CONFIG-L-Kat) wird z.B. im TAG <text2> das Kategorie-Kürzel eingetragen.

Bei Abfragen der Land-Liste (CONFIG-L-LAND) wird z.B. im TAG <text2> die Staatsangehörigkeit und im TAG <text3> das ISO-Kennzeichen übergeben.

4 Verhalten im Fehlerfall

Im fachlichen Fehlerfall ist in der Antwort am Webservice-Klienten ein Fehlerobjekt vom Typ „cFehlermeldungen“ enthalten, das eine Menge von Objekten des Typs Fehler beinhalten kann.

Aufbau:



Beispiel:

```
<fehlermeldungen>
  <fehler>
    <code>string</code>
    <beschreibung>string</beschreibung>
    <!--Element bezug is optional-->
    <bezug>string</bezug>
  </fehler>
  <fehler>
    <code>string</code>
    <beschreibung>string</beschreibung>
    <bezug>string</bezug>
  </fehler>
</fehlermeldungen>
```

Die momentan definierten Fehler sind im Dokument „Meldeschein_Webservice_Fehlercodes_.pdf“ aufgeführt. Im technischen Fehlerfall wird ein Standard-SOAP-Fault als Antwort gesendet.

5 Anwendungsfälle

Anwendungsfall	Meldescheine erstellen									
Kurzbeschreibung	Im Buchungssystem wird eine Zimmerbuchung angelegt. Die Meldeschein-Schnittstelle erhält aus dieser Buchung die meldescheinrelevanten Daten und erstellt einen Meldeschein.									
Beteiligte Akteure	<table><tr><td>Kürzel</td><td>Bezeichnung</td></tr><tr><td>Ht</td><td>Hotelier</td></tr><tr><td>WS</td><td>Meldescheinschnittstelle</td></tr><tr><td>BS</td><td>Buchungssystem</td></tr></table>		Kürzel	Bezeichnung	Ht	Hotelier	WS	Meldescheinschnittstelle	BS	Buchungssystem
Kürzel	Bezeichnung									
Ht	Hotelier									
WS	Meldescheinschnittstelle									
BS	Buchungssystem									
Auslöser, Vorbedingungen	Die Buchung ist erfasst und wird gespeichert									
Ergebnis, Nachbedingungen	Im Meldescheinsystem ist ein Meldeschein angelegt und kann am Client ausgedruckt werden oder das Buchungssystem erhält eine Fehlermeldung.									
	Ablauf, Interaktionen									
Nr. 1	Meldeschein erstellen									
Nr.	Akteur / System	Beschreibung								
1.1	Ht	Der Akteur schließt seine Buchung im Buchungssystem ab.								
1.2	BS	Das Buchungssystem füllt ein Schnittstellenobjekt (Objekt meldescheine) mit den notwendigen Daten und übergibt das Schnittstellenobjekt dem WS.								
1.3	WS	<p>Der Webservice validiert das Schnittstellenobjekt</p> <p>Validierungen:</p> <p>Die Buchungsnummer wird auf Eindeutigkeit pro Objekt geprüft: Fehlercode: 10104</p> <p>Bei der Kombination Firma und Objekt muss eine Firma-Objekt-Beziehung im AVSMeldeschein bestehen, ansonsten kommt die Fehlercode: 10101</p> <p>Abreise-Datum ist größer gleich Anreise-Datum, ansonsten Fehlercode: 10100</p> <p>Bei den Begleitpersonen muss der Aufenthaltszeitraum innerhalb des Aufenthaltszeitraums der Hauptperson liegen: Fehlercode: 10102</p> <p>Für den ausgewählten Aufenthaltszeitraum und der Kategorie muss ein Tarif definiert sein, um die Kurtaxe zu berechnen. Fehlercode: 10103</p>								

		<p>Ist für die Kategorie ein maximaler Aufenthaltszeitraum definiert (→ Schnuppergast), wird dies geprüft: Fehlercode: 50110</p> <p>Die übergebene Kategorie muss im System angelegt sein, sonst wird der Meldeschein nicht angelegt. Fehlercode 10121</p> <p>Sind Zusatzfelder (siehe Punkt 3.2.4.24) definiert, wird bei einem ungültigen Schwerbehindertenausweis-Ablauf-Datum ein Fehlercode geliefert: Fehlercode: 10201</p> <p>Der Ausdruck des Meldescheins und der Gästekarten kann erst erfolgen, wenn die erforderliche Mindestteilnehmerzahl der Person, welche für Ihre Kurverwaltung hinterlegt wurde, erreicht ist. Fehlercode 10111</p> <p>Hinweise Sonderprozess manuelle Meldescheine:</p> <p>Bei manuellen MS-Firmen wird die Buchungsnummer (entspricht der manuellen MS-Nummer des AVS-Systems) auf Eindeutigkeit geprüft → Fehlercodes: 10104 und/oder 10114 Prüfung, ob manuelle MS-Nummer zur Firma im AVS-System gepflegt, Fehlercode 10113</p> <p>Wird das Kennzeichen <kz_manuelle_ms> mit <i>true</i> übertragen (siehe 3.2.3.2), so dürfen die Firma-Id und Objekt-Id nicht mit angegeben werden. Diese werden automatisch aus der Systemkonfiguration ermittelt. Mögliche Fehlercodes: 10101, 10113</p> <p>Bei manuellen MS-Firmen wird die Buchungsnummer (entspricht der manuellen Ms-Nummer des AVS-Systems) auf Existenz geprüft → Fehlercodes: 10104 oder 10114</p>
1.4	WS	Der Webservice legt den Meldeschein an; Das Schnittstellenobjekt (Objekt meldescheine) wird mit den ursprünglichen Daten, angereichert um Meldescheinnummer, Betrag und AbrechnungsstatusId, zurückgegeben.
1.5	WS	Ausnahme: Die Validierung des Schnittstellenobjektes ist fehlgeschlagen. Der WS liefert ein Fehlerobjekt zurück
1.6	BS	Bei erfolgreicher Erstellung des Meldescheines kann dieser nun über die Webservice-Funktion druckeMeldeschein gedruckt werden.
		Abweichende Interaktionen
Nr. 2		Der WS ist nicht erreichbar

Nr.	Akteur / System	Beschreibung
2.1	HT	Der Beherberger legt eine neue Buchung an und speichert diese.
2.2	BS	Das Buchungssystem ruft den WS auf. Dieser ist jedoch nicht erreichbar
2.3	BS	Das Buchungssystem liefert eine Fehlermeldung dass die Buchung nicht gespeichert werden kann. Alternativ kann ein Objektpool geschaffen werden, in welchen die Schnittstellenobjekte für eine spätere Verarbeitung zwischen gespeichert werden.

Anwendungsfall		Meldeschein ändern								
Kurzbeschreibung		Im Buchungssystem wird eine bestehende Zimmerbuchung geändert. Die Meldeschein-Schnittstelle erhält aus dieser Änderung die meldescheinrelevanten Daten und aktualisiert einen Meldeschein.								
Beteiligte Akteure		<table><tr><td>Kürzel</td><td>Bezeichnung</td></tr><tr><td>Ht</td><td>Hotelier</td></tr><tr><td>WS</td><td>Meldescheinschnittstelle</td></tr><tr><td>BS</td><td>Buchungssystem</td></tr></table>	Kürzel	Bezeichnung	Ht	Hotelier	WS	Meldescheinschnittstelle	BS	Buchungssystem
		Kürzel	Bezeichnung							
		Ht	Hotelier							
		WS	Meldescheinschnittstelle							
		BS	Buchungssystem							
Auslöser, Vorbedingungen		Eine bestehende Zimmerbuchung wird geändert und gespeichert.								
Ergebnis, Nachbedingungen		Im Meldescheinsystem ist ein Meldeschein aktualisiert oder das Buchungssystem erhält eine Fehlermeldung.								
		Ablauf, Interaktionen								
Nr. 1		Meldeschein erstellen								
Nr.	Akteur / System	Beschreibung								
1.1	Ht	Der Akteur ruft eine bestehende Buchung auf, ändert diese und speichert ab.								
1.2	BS	Das Buchungssystem holt anhand der Buchungsnummer + Objekt-ID über die Schnittstellenfunktion MS-HOLEN die aktuellen Daten des Meldescheines.								
1.3	BS	Das Schnittstellenobjekt wird vom Buchungssystem aktualisiert.								
1.4	BS	Das Buchungssystem übergibt das Schnittstellenobjekt der WS-Funktion bucheMeldeschein .								
1.5	WS	Der Meldschein wird anhand der übergebenen Meldescheinnummer vom Webservice geladen. Falls dieserr nicht in der DB gefunden wird die Fehlermeldung 10106 zurückgegeben.								
1.4	WS	Der Webservice validiert die übergebenen Daten. Es wird der Status des Meldescheines überprüft. Falls der aktuelle Status kein Ändern mehr zulässt wird der Fehler 10107 zurückgegeben. Anschließend werden dieselben Validierungen wie bei der Neuanlage durchgeführt. Eine Ausnahme bildet die Buchungsnummer. Diese darf nicht geändert werden. Begleitpersonen/-gruppen werden über ihre Personld (von AVS) identifiziert. Ohne Personld erfolgt eine Neuanlage der Begleitperson/-gruppe. Die Person-ID wird nach dem Speichern oder beim Holen eines Meldescheins mitgeliefert.								

1.5	WS	Ausnahme: Die Validierung des Schnittstellenobjektes ist fehlgeschlagen. Der WS liefert ein Fehlerobjekt zurück
		Abweichende Interaktionen
Nr. 2	Der WS ist nicht erreichbar (siehe Anwendungsfall „ Meldeschein erstellen “).	
Nr.	Akteur / System	Beschreibung

Anwendungsfall	Meldescheine drucken	
Kurzbeschreibung	Im Buchungssystem wird eine Meldeschein-PDF erzeugt.	
Beteiligte Akteure	Kürzel	Bezeichnung
	Ht	Hotelier
	WS	Meldescheinschnittstelle
	BS	Buchungssystem
Auslöser, Vorbedingungen	Eine bestehende Zimmerbuchung wird gedruckt.	
Ergebnis, Nachbedingungen	Im Meldescheinsystem wird ein Meldeschein-PDF erzeugt und der Meldeschein auf Status „gedruckt“ gesetzt.	
	Ablauf, Interaktionen	
Nr. 1	Meldeschein erstellen	
Nr.	Akteur / System	Beschreibung
1.1	Ht	Der Akteur ruft eine bestehende Buchung auf und druckt diese.
1.2	BS	Das Buchungssystem holt anhand der Buchungsnummer + Objekt-ID über die Schnittstellenfunktion holeMeldeschein die Meldescheinnummer.
1.3	BS	Das Schnittstellenobjekt wird vom Buchungssystem gedruckt.
1.4	BS	Das Buchungssystem übergibt die Meldescheinnummer der WS-Funktion druckeMeldeschein
1.5	WS	Der Meldschein setzt anhand der übergebenen Meldescheinnummer den Druckstatus und erzeugt ein

		Meldeschein-PDF. Falls er nicht in der DB gefunden wird die Fehlermeldung 10106 zurückgegeben. Falls der Meldeschein nicht mehr gedruckt werden darf, weil er z.B. storniert wurde, wird die Fehlermeldung 10110 geliefert. Falls bei einem Gruppenmeldeschein die Mindestanzahl an Begleitpersonen nicht erreicht wird, liefert der Webservice den Fehlercode 10111.
		Abweichende Interaktionen
Nr. 2	Der WS ist nicht erreichbar (siehe Anwendungsfall „Meldeschein“ erstellen“	
Nr.	Akteur / System	Beschreibung
Anwendungsfall	Meldescheine stornieren	
Kurzbeschreibung	Im Buchungssystem wird eine bestehende Zimmerbuchung storniert.	
Beteiligte Akteure	Kürzel	Bezeichnung
	Ht	Hotelier
	WS	Meldescheinschnittstelle
	BS	Buchungssystem
Auslöser, Vorbedingungen	Eine bestehende Zimmerbuchung wird storniert	
Ergebnis, Nachbedingungen	Im Meldescheinsystem ist der Meldeschein aktualisiert oder das Buchungssystem erhält eine Fehlermeldung.	
		Ablauf, Interaktionen
Nr. 1	Meldeschein erstellen	
Nr.	Akteur / System	Beschreibung
1.1	Ht	Der Akteur ruft eine bestehende Buchung auf und storniert diese.
1.2	BS	Das Buchungssystem holt anhand der Buchungsnummer + Objekt-ID über die Schnittstellenfunktion holeMeldeschein die aktuellen Daten des Meldescheines.
1.3	BS	Das Schnittstellenobjekt wird vom Buchungssystem aktualisiert: Die <abrechnungstatusid> wird auf 4 gesetzt → Storno-Status.
1.4	BS	Das Buchungssystem übergibt das Schnittstellenobjekt der WS-Funktion bucheMeldeschein .
1.5	WS	Der Melschein wird anhand der übergebenen Meldescheinnummer vom Webservice geladen. Falls er nicht in der DB gefunden wird die Fehlermeldung 10106 zurückgegeben

1.4	WS	Der Webservice validiert die übergebenen Daten. Es wird der Status des Meldescheines überprüft. Falls der akt. Status kein Ändern / Stornieren mehr zulässt wird der Fehler 10107 / 10112 zurückgegeben.
1.5	WS	Ausnahme: Die Validierung des Schnittstellenobjektes ist fehlgeschlagen. Der WS liefert ein Fehlerobjekt zurück.
		Abweichende Interaktionen
Nr. 2	Der WS ist nicht erreichbar (Siehe Anwendungsfall „ Meldeschein erstellen “	
Nr.	Akteur / System	Beschreibung

6 SOAP-Request für Funktion “bucheMeldeschein”

Diese Funktion regelt die Erzeugung und Änderung von Meldescheinen.

Bsp-Request Neuanlage:

```
<?xml version="1.0"?>
<ms:meldeschein xmlns:ms="http://www.av.s.meldeschein.de/ns/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <identifikation>
    <erzeugung>2016-07-01</erzeugung>
    <schnittstelle>xxx</schnittstelle>
    <kurverwaltung>1</kurverwaltung>
    <benutzerid>1</benutzerid>
    <sessionid>string</sessionid>
    <verarbeitung>BUCHEN</verarbeitung>
    <version>string</version>
  </identifikation>
  <ms:meldeschein>
    <buchungsnummer>30501</buchungsnummer>
    <firmaid>222</firmaid>
    <objektid>222</objektid>
    <anreise>2016-07-01</anreise>
    <abreise>2016-07-02</abreise>
    <kategorieid>2</kategorieid>
    <anredeid>1</anredeid>
    <name>Test-Name</name>
    <vorname>Test-Vorname</vorname>
    <strasse>Teststr.</strasse>
    <hausnummer>1</hausnummer>
    <plz>12345</plz>
    <ort>Teststadt</ort>
    <landid>1</landid>
    <staatsangehoerigkeitid>1</staatsangehoerigkeitid>
    <weitere_angaben>12345678</weitere_angaben>
    <ausweisnr>12345678ABC</ausweisnr>
    <kfzkennzeichen>BT TB-44</kfzkennzeichen>
    <geburtsdatum>1972-01-21</geburtsdatum>
    <begleitperson>
      <anredeid>1</anredeid>
      <name>Test-Name-BP</name>
      <vorname>Test-Vorname-BP</vorname>
      <geburtsdatum>1955-03-11</geburtsdatum>
      <kategorieid>3</kategorieid>
      <anreise>2016-07-01</anreise>
      <abreise>2016-07-02</abreise>
      <ausweisnr>123478ABC</ausweisnr>
    </begleitperson>
  </ms:meldeschein>
</ms:meldeschein>
```

```

    <begleitperson>
      <anredeid>2</anredeid>
      <name>string</name>
      <vorname>string</vorname>
      <geburtsdatum>1999-01-21</geburtsdatum>
      <kategorieid>2</kategorieid>
      <ausweisnr>45678ABC</ausweisnr>
    </begleitperson>
  </ms:meldeschein>
</ms:meldescheine>

```

Das Tag `<verarbeitung>` muss den Wert **BUCHEN** bei dieser Funktion haben. Die Id's für Anrede, Firma etc. sind dann den entsprechenden Konfigurationslisten zu entnehmen. Beim Tag `<staatsangehoerigkeitid>` wird die Id von der Land-KonfigListe verwendet.

Bsp-Response:

Im Response werden dann wichtige Informationen angehängt (**grün** hervorgehoben).

```

<?xml version="1.0"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Body>
    <meldescheine xmlns="http://www.avs.meldeschein.de/ns">
      <identifikation xmlns="">
        <erzeugung>2016-07-01</erzeugung>
        <schnittstelle>xxx</schnittstelle>
        <kurverwaltung>1</kurverwaltung>
        <benutzerid>1</benutzerid>
        <sessionid>string</sessionid>
        <verarbeitung>BUCHEN</verarbeitung>
        <version>1.2.06-H03</version>
      </identifikation>
      <fehlermeldungen xmlns="">
        <fehler>
          <code>10001</code>
          <beschreibung>Meldescheine-Buchen erfolgreich</beschreibung>
        </fehler>
      </fehlermeldungen>
      <meldeschein>
        <buchungsnummer xmlns="">30501</buchungsnummer>
        <meldescheinnummer xmlns="">216454</meldescheinnummer>
        <firmaid xmlns="">222</firmaid>
        <objektid xmlns="">222</objektid>
        <anreise xmlns="">2016-07-01</anreise>
        <abreise xmlns="">2016-07-02</abreise>
        <kategorieid xmlns="">2</kategorieid>
        <anredeid xmlns="">1</anredeid>
        <name xmlns="">Test-Name</name>
      </meldeschein>
    </meldescheine>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```



```

<vorname xmlns="">Test-Vorname</vorname>
<strasse xmlns="">Teststr.</strasse>
<hausnummer xmlns="">1</hausnummer>
<plz xmlns="">12345</plz>
<ort xmlns="">Teststadt</ort>
<landid xmlns="">1</landid>
<staatsangehoerigkeitid xmlns="">1</staatsangehoerigkeitid>
<weitere_angaben xmlns="">12345678</weitere_angaben>
<ausweisnr xmlns="">12345678ABC</ausweisnr>
<kfzkennzeichen xmlns="">BT TB-44</kfzkennzeichen>
<geburtsdatum xmlns="">1972-01-21</geburtsdatum>
<betrag xmlns="">10</betrag>
<abrechnungstatusid xmlns="">1</abrechnungstatusid>
<begleitperson xmlns="">
  <personid>529955</personid>
  <anredeid>1</anredeid>
  <name>Test-Name-BP</name>
  <vorname>Test-Vorname-BP</vorname>
  <geburtsdatum>1955-03-11</geburtsdatum>
  <kategorieid>3</kategorieid>
  <ausweisnr >123478ABC</ausweisnr>
  <anreise>2016-07-01</anreise>
  <abreise>2016-07-02</abreise>
</begleitperson>
<begleitperson xmlns="">
  <personid>529956</personid>
  <anredeid>2</anredeid>
  <name>string</name>
  <vorname>string</vorname>
  <geburtsdatum>1999-01-21</geburtsdatum>
  <kategorieid>2</kategorieid>
  <ausweisnr >45678ABC</ausweisnr>
</begleitperson>
</meldeschein>
</meldescheine>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Der TAG **<meldescheinnummer>** ist für das **Ändern / Update** des Meldescheines unbedingt notwendig, ansonsten wird immer ein neuer Meldeschein angelegt! Wird die gleiche Buchungsnummer 2-mal pro Objekt gebucht, wird beim zweiten Mal der Errorcode **10104** ("Meldeschein mit der Buchungsnummer existiert schon!") zurück geliefert.

Im Detail:

Neuanlage Meldeschein:

...
<buchungsnummer>1A2B3C4D5E </buchungsnummer>
...

Ändern / Update Meldeschein:

...
<meldescheinnummer>1A2B3C4D5E </meldescheinnummer>
...

oder

...
<buchungsnummer>1A2B3C4D5E </buchungsnummer>
<meldescheinnummer>1A2B3C4D5E </meldescheinnummer>
...

Der TAG <personid> (von AVS) ist fürs Update zum eindeutigen Identifizieren der Begleitperson(en) zwingend notwendig!
Wird beim Update kein TAG <personid> übergeben, wird eine neue Begleitperson angelegt/hinzugefügt!

7 SOAP-Request „holeMeldeschein“

Diese Funktion dient dazu, die aktuellen Meldescheindaten anhand der

- Buchungsnummer und Objekt
- Meldscheinnummer

zur Anzeige oder spätere Updates abzuholen.

Die Funktion „holeMeldeschein“ wird benutzt, um die Daten des angelegten Meldescheines aus der AVSMeldeschein Anwendung abzufragen. Sollte nach der Anlage noch eine Änderung an dem Meldeschein vorgenommen werden müssen, dann können die Daten über diese Funktion abgeholt bzw. angezeigt werden und über die Funktion „bucheMeldeschein“ geändert werden.

Bei der Methode „MS-Holen“ sind verschiedene Kombinationen von TAG`s möglich (siehe unten).

Wird der TAG <meldescheinnummer> und der TAG <buchungsnummer> gemeinsam übergeben, dann ist der TAG <meldescheinnummer> führend. Der TAG <buchungsnummer> und der TAG <objektid> werden in diesem Fall nur ausgelesen, wenn der TAG <meldescheinnummer> leer übergeben wird.

Der TAG <buchungsnummer> muss immer in Kombination mit dem TAG <objektid> übergeben werden, da diese beiden Werte zusammen einen Meldeschein eindeutig identifizieren. (Vorausgesetzt der Meldeschein wurde mit einer Buchungsnummer angelegt)

Folgende Kombinationen der TAG sind möglich:

Buchungsnummer	+	Objektid		
Meldescheinnummer	+	Objektid		
Meldescheinnummer	+	Buchungsnummer		
Meldescheinnummer	+	Buchungsnummer	+	Objektid
Meldescheinnummer				

7.1 Request holeMeldeschein mit Buchungsnummer

```
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Body>
    <holeMeldeschein xmlns="http://www.avs.meldeschein.de/ns/">
      <identifikation xmlns="">
        <erzeugung>2016-07-01</erzeugung>
        <schnittstelle>xxx</schnittstelle>
        <kurverwaltung>1</kurverwaltung>
        <benutzerid>1</benutzerid>
        <sessionid>string</sessionid>
        <verarbeitung>MS-HOLEN</verarbeitung>
        <version>1.2.06-H03</version>
      </identifikation>
      <anfragedaten xmlns="">
        <buchungsnummer>30001</buchungsnummer>
        <objektid>222</objektid>
      </anfragedaten>
    </holeMeldeschein>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

7.2 Request holeMeldeschein mit Meldescheinnummer

```
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Body>
    <holeMeldeschein xmlns="http://www.avs.meldeschein.de/ns/">
      <identifikation xmlns="">
        <erzeugung>2016-07-01</erzeugung>
```

```

    <schnittstelle>xxx</schnittstelle>
    <kurverwaltung>1</kurverwaltung>
    <benutzerid>1</benutzerid>
    <sessionid>string</sessionid>
    <verarbeitung>MS-HOLEN</verarbeitung>
    <version>1.2.06-H03</version>
  </identifikation>
  <anfragedaten xmlns="">
    <meldescheinnummer>30001</meldescheinnummer>
  </anfragedaten>
</holeMeldeschein>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

7.3 SOAP-Response „holeMeldeschein“

```

<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Body>
    <holeMeldeschein xmlns="http://www.avs.meldeschein.de/ns/">
      <identifikation xmlns="">
        <erzeugung>2016-07-01</erzeugung>
        <schnittstelle>xxx</schnittstelle>
        <kurverwaltung>1</kurverwaltung>
        <benutzerid>1</benutzerid>
        <sessionid>string</sessionid>
        <verarbeitung>MS-HOLEN</verarbeitung>
        <version>1.2.06-H03</version>
      </identifikation>
      <anfragedaten xmlns="">
        <buchungsnummer>30001</buchungsnummer>
        <objektid>222</objektid>
      </anfragedaten>
      <fehlermeldungen xmlns="">
        <fehler>
          <code>10001</code>
          <beschreibung>Meldescheine-Holen erfolgreich</beschreibung>
        </fehler>
      </fehlermeldungen>
      <meldeschein>
        <buchungsnummer xmlns="">30001</buchungsnummer>
        <meldescheinnummer xmlns="">123456</meldescheinnummer>
        <firmaid xmlns="">111</firmaid>
        <objektid xmlns="">222</objektid>
        <anreise xmlns="">2016-07-01</anreise>
      </meldeschein>
    </holeMeldeschein>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

```

<abreise xmlns="">2016-07-02</abreise>
<kategorieid xmlns="">2</kategorieid>
<anredeid xmlns="">1</anredeid>
<name xmlns="">Test-Name</name>
<vorname xmlns="">Test-Vorname</vorname>
<strasse xmlns="">Teststr.</strasse>
<hausnummer xmlns="">1</hausnummer>
<plz xmlns="">12345</plz>
<ort xmlns="">Teststadt</ort>
<landid xmlns="">1</landid>
<staatsangehoerigkeitid xmlns="">1</staatsangehoerigkeitid>
<weitere_angaben xmlns="">12345678</weitere_angaben>
<ausweisnr xmlns="">12345678ABC</ausweisnr>
<kfzkennzeichen xmlns="">BT TB-44</kfzkennzeichen>
<geburtsdatum xmlns="">1972-01-21</geburtsdatum>
<abrechnungstatusid xmlns="">1</abrechnungstatusid>
<begleitperson xmlns="">
  <personid>1234</personid>
  <anredeid>1</anredeid>
  <name>Test-Name-BP</name>
  <vorname>Test-Vorname-BP</vorname>
  <geburtsdatum>1955-03-11</geburtsdatum>
  <kategorieid>3</kategorieid>
  <anreise>2016-07-01</anreise>
  <abreise>2016-07-02</abreise>
  <ausweisnr >123478ABC</ausweisnr>
</begleitperson>
<begleitperson xmlns="">
  <personid>2345</personid>
  <anredeid>2</anredeid>
  <name>string</name>
  <vorname>string</vorname>
  <geburtsdatum>1999-01-21</geburtsdatum>
  <kategorieid>2</kategorieid>
  <anreise>2016-07-01</anreise>
  <abreise>2016-07-02</abreise>
  <ausweisnr >45678ABC</ausweisnr>
</begleitperson>
</meldeschein>
</holeMeldeschein>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

8 SOAP-Request „getKonfigurationsListe“

Für die Kategorien-Abfrage würde die SOAP-Response dann z.B. folgendermaßen aussehen:

```
<?xml version="1.0"?>
<SOAP-ENV:Envelope xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Body>
    <configuration-lists xmlns="http://www.avs.meldeschein.de/ns/">
      <identifikation xmlns="">
        <erzeugung>2016-07-01</erzeugung>
        <schnittstelle>xxx</schnittstelle>
        <kurverwaltung>1</kurverwaltung>
        <benutzerid>1</benutzerid>
        <sessionid>string</sessionid>
        <verarbeitung>CONFIG-L-KAT</verarbeitung>
        <version>1.2.06-H03</version>
      </identifikation>
      <fehlermeldungen xmlns="">
        <fehler>
          <code>10001</code>
          <beschreibung>ConfigurationLists-Abfrage erfolgreich</beschreibung>
        </fehler>
      </fehlermeldungen>
      <konfigliste xmlns="">
        <konfigdatensatz>
          <id>1</id>
          <text1>Erwachsener</text1>
          <text2>E</text2>
        </konfigdatensatz>
        <konfigdatensatz>
          <id>5</id>
          <text1>Erwachsener Begleitperson</text1>
          <text2>EB</text2>
        </konfigdatensatz>
        <konfigdatensatz>
          <id>2</id>
          <text1>Erwachsener 100%</text1>
          <text2>EE</text2>
        </konfigdatensatz>
        <konfigdatensatz>
          <id>3</id>
          <text1>Kind</text1>
          <text2>K</text2>
        </konfigdatensatz>
        <konfigdatensatz>
          <id>4</id>
```

```
<text1>Kind ermäßigt</text1>
<text2>KE</text2>
</konfigdatensatz>
<konfigdatensatz>
  <id>9</id>
  <text1>keine Kategorie</text1>
  <text2>0</text2>
</konfigdatensatz>
</konfigliste>
</configuration-lists>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

9 Beispiele

9.1 Request bucheMeldeschein Gruppenmeldeschein

Bsp.-Request Neuanlage Gruppenmeldeschein:

```
<?xml version="1.0"?>
<ms:meldescheine xmlns:ms="http://www.av.s.meldeschein.de/ns/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <identifikation>
    <erzeugung>2016-07-01</erzeugung>
    <schnittstelle>xxx</schnittstelle>
    <kurverwaltung>1</kurverwaltung>
    <benutzerid>1</benutzerid>
    <sessionid>string</sessionid>
    <verarbeitung>BUCHEN</verarbeitung>
    <version>string</version>
  </identifikation>
  <ms:meldeschein>
    <buchungsnummer>30501</buchungsnummer>
    <firmaid>222</firmaid>
    <objektid>222</objektid>
    <anreise>2016-07-01</anreise>
    <abreise>2016-07-02</abreise>
    <kategorieid>2</kategorieid>
    <anredeid>1</anredeid>
    <name>Test-Name</name>
    <vorname>Test-Vorname</vorname>
    <strasse>Teststr.</strasse>
    <hausnummer>1</hausnummer>
    <plz>12345</plz>
    <ort>Teststadt</ort>
    <landid>1</landid>
    <staatsangehoerigkeitid>1</staatsangehoerigkeitid>
    <weitere_angaben>12345678</weitere_angaben>
    <ausweisnr>12345678ABC</ausweisnr>
    <kfzkennzeichen>BT TB-44</kfzkennzeichen>
    <geburtsdatum>1972-01-21</geburtsdatum>
    <begleitgruppe>
      <kategorieid>1</kategorieid>
      <anzahl>4</anzahl>
      <ausweisnr>12345678ABCD</ausweisnr>
    </begleitgruppe>
  </ms:meldeschein>
</ms:meldescheine>
```


9.2 Response Fehlerhafter Login

Werden beim Aufruf des Meldeschein -Webservices falsche Benutzerdaten (Benutzername, Passwort) übergeben, erhält der Benutzer den HTTP-Statuscode 401. Bei dieser sogenannten HTTP-Basic Authentication werden die Benutzerdaten an den Server base64 kodiert übermittelt und erst in einem weiteren Schritt erfolgt anhand der im Identifikationsblock enthaltenen Daten die Autorisierung.

Bsp.-Response fehlerhafter Login:

Error 401--Unauthorized

From RFC 2068 Hypertext Transfer Protocol -- HTTP/1.1:

10.4.2 401 Unauthorized

The request requires user authentication. The response MUST include a WWW-Authenticate header field (section 14.46) containing a challenge applicable to the requested resource. The client MAY repeat the request with a suitable Authorization header field (section 14.8). If the request already included Authorization credentials, then the 401 response indicates that authorization has been refused for those credentials. If the 401 response contains the same challenge as the prior response, and the user agent has already attempted authentication at least once, then the user SHOULD be presented the entity that was given in the response, since that entity MAY include relevant diagnostic information. HTTP access authentication is explained in section 11.

10 Support Anfragen

Sollten unvorhergesehene Fehler im Webservice auftreten, dann können Sie sich gerne an die AVS GmbH wenden. Die Anfrage sollte immer das exakte Datum mit Uhrzeit beinhalten und zusätzlich sollte auch der Request und der entsprechende Response mit der Fehlermeldung mitgeschickt werden. Eine Anfrage kann ohne diese Details nur eingeschränkt bearbeitet werden und die Antwort dauert dann, je nach Rechercheaufwand, entsprechend länger.

11 Dazu zugehörige Quellen:

Dateiname	Bemerkung
meldeschein_buchen.xsd	Definitionen der Schnittstellenobjekte für Meldescheinverwaltung
meldeschein_datentypen.xsd	Grundlegende Datentypen

IJMeldescheinWebservices.wsdl	WSDL-Definition
Meldeschein Webservice Fehlercodes 6.3	Verzeichnis aller fachlichen Fehlercodes, die der Webservice liefern kann.