

DIMAG Schnittstellenbeschreibung

David Radlinger

2019/12/18 14:25

Inhaltsverzeichnis

DIMAG Schnittstellenbeschreibung	3
DIMAG SOAP	3
WSDL	3
DIMAG SOAP-Operationen im Überblick	3
getStatus	3
importDoc	4
exportDoc	5
changeStatusCode	7
getAIDforKeyValue	8
getObjectsChangedSince	9
moveObject	9
getPrefixList	10
importDocAfis	11
getForms	12
getFields	13
LoadXML	15

DIMAG Schnittstellenbeschreibung

- [DIMAG Schnittstellenbeschreibung](#)
- [DIMAG SOAP](#)
 - [WSDL](#)
 - [DIMAG SOAP-Operationen im Überblick](#)
 - [getStatus](#)
 - [importDoc](#)
 - [exportDoc](#)
 - [changeStatusCode](#)
 - [getAIDforKeyValue](#)
 - [getObjectsChangedSince](#)
 - [moveObject](#)
 - [getPrefixList](#)
 - [importDocAfis](#)
 - [getForms](#)
 - [getFields](#)
- [LoadXML](#)

Diese Dokumentation beschreibt die Programmierschnittstelle des DIMAG-Kernmoduls. (Getestet mit KM-Version 3.2.0, Afis-Connector 0.7). Hierzu zählen die eigentliche SOAP-Schnittstelle, die Beschreibung des LoadXML-Formats und der Laderampe für Ingest und Export.

DIMAG SOAP

DIMAG-SOAP basiert auf SOAP über http. Neben grundlegenden Operationen des Kernmoduls kann auch ein mit DIMAG gekoppeltes Archivinformationssystem (AFIS) über SOAP angesprochen werden.

Die Schnittstelle ermöglicht dem SOAP-Client folgende Operationen durchzuführen:

- Abfrage nach dem Status des DIMAG Storage ([getStatus](#))
- Importieren von Dokumenten in den DIMAG Storage ([importDoc](#))
- Exportieren von Dokumenten aus dem DIMAG Storage ([exportDoc](#))
- Ändern des Status eines Objekts ([changeStatusCode](#))
- Suchen von AID im DIMAG Storage über Metadatenuche ([getAIDforKeyValue](#))
- Suchen von AID nach Änderungsdatum ([getObjectsChangedSince](#))
- Verschieben von Strukturelement im DIMAG Storage ([moveObject](#))
- Abfrage aller angeschlossenen AFIS-Systeme ([getPrefixList](#))
- Importieren von Strukturdaten nach AFIS ([importDocAfis](#))
- Abfragen aller Formulare (Erschließungsmasken) des AFIS ([getForms](#))
- Abfragen der Feldnamen eines AFIS-Formulars ([getFields](#))

WSDL

DIMAG veröffentlicht die angebotenen SOAP-Operationen in einem WSDL-Dokument. Dieses befindet sich im Quellcode unter `"/soap/webservice_3_2_0.wsdl"`

Der Endpoint ist in `"/soap/webservice_3_2_0.php"` definiert. Dieser ist nicht Bestandteil der WSDL und muss separat gesetzt werden.

(Für die vorige DIMAG-Kernmodul-Version 3.1.0 gilt `"soap/webservice_3_1_0.wsdl"` und der entsprechende Endpoint.)

DIMAG SOAP-Operationen im Überblick

getStatus

Diese Operation liefert den Status von DIMAG zurück. Der Request ist None.

Request

Response

status	Integer
--------	---------

	2xx: ok
	5xx: server problems
msg	String

Bsp. Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://dimag.la-bw.de/WebService.wsdl">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getStatus/>
  </soapenv:Body>
</soapenv:Envelope>
```

Bsp. Response

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://dimag.la-bw.de/WebService.wsdl">
  <SOAP-ENV:Body>
    <ns1:getStatusResponse>
      <status>200</status>
      <msg>DIMAG-3.0.4-afis-006-labw [10:35:30]
URL:      https://dimag2.la-bw.de/
ArchivName:  DimagDay
sftp Path:   /data/sftpusers/ftpdimag-afis-kopplung/Import
Import XSD Version: LaBW_IngExt_2019-05-22.xsd</msg>
    </ns1:getStatusResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

importDoc

Zum Transfer größerer Datenmengen verfügt DIMAG über eine Laderampe, also ein Verzeichnis in das von außen Daten eingestellt werden können, mit dem Ziel, dass DIMAG sie anschließend aufnimmt (Ingest). Ingestlist und Ingesttool legen die zu transferierenden Daten in der Laderampe ab. Im Ingestlist-Fall müssen abgelegte Dateien vom Nutzer über die Weboberfläche (im passenden R-Objekt) in den DIMAG-Storage importiert werden. Ingesttool lässt die abgelegten Daten per SOAP-Call von DIMAG importieren.

Die Dateiübertragung von Primärdaten geschieht über sftp. Die SOAP-Calls stoßen lediglich den Upload-Prozess an. Via SOAP werden nur Metadaten übertragen.

importDoc veranlasst DIMAG das Importieren von Dokumenten aus dem DIMAG sftp-Transferbereich in DIMAG Storage zu starten. Diese Dokumente müssen schon vorher über SFTP in den DIMAG SFTP Transferbereich transferiert werden. Die Dokumentendateien im DIMAG sftp-Transferbereich werden von DIMAG gelöscht, auch im Fehlerfall. Die Durchführung findet in einer Transaktion statt. Gibt es bei einem <verz-obj> einen Fehler, so gilt die gesamte Transaktion als fehlerhaft. Beim Ingest über importDoc findet implizit ein Integritäts-Check mittels Validierung der md5-Prüfsummen statt.

Das Controlfile muss im [LoadXML](#)-Format vorliegen. Die Objekte werden in der darin vorgegebenen Struktur in DIMAG angelegt.

Request

username	String	DIMAG User to make the Operation with
password	String	Password of the user
ControlFile	String	Location of the control file on the sftp server relative to sftp root

Im Gegensatz zur früheren SOAP-Schnittstelle werden Passwörter nun im Klartext übermittelt (nicht mehr als md5-Hash).

Response

status	Integer
--------	---------

	2xx	successfully imported to DIMAG
	3xx	temporary fail: try again identical
	4xx	permanent fail: change request
	5xx	server problem: client cannot fix it
msg	String -- human readable return message	

Bsp. Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://dimag.la-bw.de/WebService.wsdl">
  <soapenv:Header/>
  <soapenv:Body>
    <web:importDoc>
      <username>testuser</username>
      <password>mypass</password>
      <ControlFile>path/mycontrolfile.xml</ControlFile>
    </web:importDoc>
  </soapenv:Body>
</soapenv:Envelope>
```

mycontrolfile.xml

```
<i><?xml version="1.0" encoding="UTF-8" standalone="no"?>
<verzeichnungseinheit xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="LaBW_IngExt_2019-05-22.xsd">
  <rootid>test-2190</rootid>
  <verz-obj>
    <pid></pid>
    <titel>MeinStrukturObjekt</titel>
    <merkmal>keine</merkmal>
    <typ>S</typ>
  </verz-obj>
  <verz-obj>
    <pid></pid>
    <titel>MeinStrukturObjekt2</titel>
    <merkmal>auch keine</merkmal>
    <typ>S</typ>
  </verz-obj>
</verzeichnungseinheit></i>
```

Bsp. Response

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://dimag.la-bw.de/WebService.wsdl">
  <SOAP-ENV:Body>
    <ns1:importDocResponse>
      <status>200</status>
      <msg>Success: createAIP(): ok: Strukturobjekt test-2204 [] : MeinStrukturObjekt inserted<br/>
ok: Protokollobjekt test-2193 [] : Protokoll zu test-2190
Ebene1 updated<br/>
ok: Strukturobjekt test-2205 [] : MeinStrukturObjekt2 inserted<br/>
ok: Protokollobjekt test-2193 [] : Protokoll zu test-2190
Ebene1 updated<br/></msg>
    </ns1:importDocResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

exportDoc

Die Laderampe wird auch zum Export beliebig großer Datenmengen genutzt. In diesem Verzeichnis kann DIMAG Dateien bereitstellen, die extern abgeholt werden können (Access). Ein DIMAG Client fordert DIMAG durch einen SOAP-Call auf, einen bestimmten Teilbaum zu exportieren. Daraufhin legt DIMAG ein eindeutiges Unterverzeichnis im DIMAG sftp-Transferbereich an und kopiert die Primärdateien aller betreffenden

Spezifikation - DIMAG Schnittstellenbeschreibung

Objekte in dieses Unterverzeichnis. Die Metadatenbeschreibung wird als generierte Datei ("verz_obj.xml") ebenfalls in dieses Verzeichnis gespeichert. Der DIMAG Client wertet den Return Wert des DIMAG SOAP-Call aus und er kann im Gutfall das Exportpaket von der DIMAG-Laderampe über sftp abholen.

Die Export-Laderampe wird nur zum Export mittels SOAP:exportDoc() verwendet.

Diese Operation veranlasst DIMAG das Exportieren von Dokumenten aus DIMAG Storage in den DIMAG SFTP Transferbereich zu starten.

Je nach Angaben werden unterschiedliche Daten exportiert:

defaultRepOnly

*first: Es wird nur die erste Repräsentation (Anlegedatum) exportiert

*all: Es werden alle Repräsentationen exportiert

*last: Es wird nur die neueste Repräsentation (Anlegedatum) exportiert

withStruct

*true: Es werden zusätzlich die darüber liegenden Objekte (Pfad zum Wurzelobjekt, d.h. Tektonikeinordnung) mit exportiert

*false: Es wird keine überliegende Struktur exportiert, sondern nur der Teilbaum des ausgewählten Objekts

ControlFileOnly

*true: Es wird ausschließlich das Controlfile exportiert. Dieses enthält alle Metadaten

*false: Es werden das Controlfile und die separaten Meta- und ggf. Primärdatendateien der selektierten Objekte exportiert

LogFile

*true: Es werden die Protokollobjekte mit exportiert

*false: Es werden keine Protokollobjekte exportiert

Request

username	String	
password	String	
aID	String	aID of the Object to start export with
defaultRepOnly	RepOptionsType	first (emulation), all, last (access) -- which representation to export
withStruct	Boolean	export the tree above (true) or only the tree below the aID
ControlFileOnly	Boolean	
LogFile	Boolean	

Response

status	Integer	
msg	String	human readable response text
ControlFile	String	the relative filename of the control file

Bsp. Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://dimag.la-bw.de/WebService.wsdl">
  <soapenv:Header/>
  <soapenv:Body>
    <web:exportDoc>
      <username>testuser</username>
      <password>mypass</password>
      <aID>test-2194</aID>
    </web:exportDoc>
  </soapenv:Body>
</soapenv:Envelope>
```

```

    <defaultRepOnly>all</defaultRepOnly>
    <withStruct>true</withStruct>
    <ControlFileOnly>>false</ControlFileOnly>
    <LogFile>true</LogFile>
  </web:exportDoc>
</soapenv:Body>
</soapenv:Envelope>

```

Bsp. Response

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://dimag.la-
bw.de/WebService.wsdl">
  <SOAP-ENV:Body>
    <ns1:exportDocResponse>
      <status>200</status>
      <msg>arguments: aid=test-2194; RepOptions=all; withStruct=true; ControlFileOnly=false; LogFile=true;
Number Objects exported: 10</msg>
      <ControlFile>/Export/2019/10/25/11.52.38-945/verz_obj.xml</ControlFile>
    </ns1:exportDocResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Die Daten werden in diesem Beispiel von DIMAG in Export/2019/10/25/11.52.38-945 des sftp-Bereiches des DIMAG-Servers bereitgestellt.

Es wurden die Metadaten aller Repräsentationen, die dazugehörenden Primärdaten sowie das Controlfile (verz_obj.xml) exportiert.

```

test-2189.xml
test-2190.xml
test-2194.xml
test-2195.xml
test-2197
test-2197.xml
test-2199.xml
test-2200
test-2200.xml
test-9.xml
verz_obj.xml

```

changeStatusCode

Diese Operation veranlasst DIMAG, den Statuscode eines Objekts zu verändern und auf den eingegebenen Status zu setzen.

Request

username	String	
password	String	
aID	String	object to change
newStatus	String	the new status code

Response

status	Integer
msg	String

Bsp. Request

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://dimag.la-bw.de/
WebService.wsdl">
  <soapenv:Header/>
  <soapenv:Body>
    <web:changeStatusCode>
      <username>testuser</username>
      <password>mypass</password>
      <aID>test-2192</aID>
      <newStatus>30</newStatus>
    </web:changeStatusCode>
  </soapenv:Body>
</soapenv:Envelope>

```

Bsp. Response

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://dimag.la-bw.de/WebService.wsdl">
  <SOAP-ENV:Body>
    <ns1:changeStatusCodeResponse>
      <status>200</status>
      <msg>1</msg>
    </ns1:changeStatusCodeResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

getAIDforKey Value

Diese Operation liefert die AID für ein Key-Value Paar eines spezifischen Objekttyps zurück.

Request

username	String	
password	String	
key	String	the XML tag name, e.g.: titel
value	String	the content including % as wild char
typ	DimagObjectType	"Any", "Struct", "Representation", "DigitalObject", "File"

Response

status	Integer	
msg	String	
hits	String	number of hits
aIDList	String	Semicolon-separated list of AIDs

Bsp. Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://dimag.la-bw.de/WebService.wsdl">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getAIDforKey Value>
      <username>testuser</username>
      <password>mypass</password>
      <key>titel</key>
      <value>test%</value>
      <typ>Struct</typ>
    </web:getAIDforKey Value>
  </soapenv:Body>
</soapenv:Envelope>
```

Bsp. Response

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://dimag.la-bw.de/WebService.wsdl">
  <SOAP-ENV:Body>
    <ns1:getAIDforKey ValueResponse>
      <status>200</status>
      <msg>found some objects</msg>
      <hits>2</hits>
      <aIDList>test-7;test-9</aIDList>
    </ns1:getAIDforKey ValueResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```


getObjectsChangedSince

Diese Operation liefert die AIDs aller Objekte zurück, die seit einem bestimmten Datum in DIMAG geändert wurden.

Request

username	String	
password	String	
sinceDate	String	Date string in the form "YYYY-MM-DD HH:MM:SS"

Response

status	Integer	
msg	String	
hits	String	number of hits
aIDList	String	Semicolon-separated list of AIDs

Bsp. Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://dimag.la-bw.de/WebService.wsdl">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getObjectsChangedSince>
      <username>testuser</username>
      <password>mypass</password>
      <sinceDate>2019-01-01 23:00:00</sinceDate>
    </web:getObjectsChangedSince>
  </soapenv:Body>
</soapenv:Envelope>
```

Bsp. Response

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://dimag.la-bw.de/WebService.wsdl">
  <SOAP-ENV:Body>
    <ns1:getObjectsChangedSinceResponse>
      <status>200</status>
      <msg>found some objects</msg>
      <hits>4</hits>
      <aIDList>D-1;test-1;test-3;test-7</aIDList>
    </ns1:getObjectsChangedSinceResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

moveObject

Mit dieser Operation kann ein Objekt verschoben werden. Hierbei wird das Objekt inklusive all seiner Kindobjekte an ein neues Elternobjekt umgehängt.

Die Operation stellt sicher, dass ein Objekt entsprechend des strukturellen Aufbaus nur unter erlaubte Objektarten platziert werden darf.

Request

username	String	
password	String	
aID	String	object to move
newpID	String	the new parent object

Response

status	Integer	
msg	String	

Bsp. Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://dimag.la-bw.de/
WebService.wsdl">
  <soapenv:Header/>
  <soapenv:Body>
    <web:moveObject>
      <username>testuser</username>
      <password>mypass</password>
      <aID>test-2212</aID>
      <newPID>test-2205</newPID>
    </web:moveObject>
  </soapenv:Body>
</soapenv:Envelope>
```

Bsp. Response

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://dimag.la-
bw.de/WebService.wsdl">
  <SOAP-ENV:Body>
    <ns1:moveObjectResponse>
      <status>200</status>
      <msg>ok: Strukturobjekt test-2212 [] : WeiteresStrukturObjekt updated<br/></msg>
    </ns1:moveObjectResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Die folgenden Operationen dienen dazu, ein an DIMAG gekoppeltes Archivinformationssystem (AFIS) anzusprechen. Hierbei stoßen die Soap-Aufrufe die Operationen eines Afis-Connectors an.

getPrefixList

Mit dieser Operation werden alle angeschlossenen AFIS-Systeme aufgelistet. Der Request ist None.

Request

Response

status	Integer
msg	String
prefixlist	

Bsp. Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://dimag.la-bw.de/
WebService.wsdl">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getPrefixList/>
  </soapenv:Body>
</soapenv:Envelope>
```

Bsp. Response (Ausschnitt)

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://dimag.la-
bw.de/WebService.wsdl">
  <SOAP-ENV:Body>
    <ns1:getPrefixListResponse>
      <status>200</status>
      <msg>Success</msg>
      <prefixList><![CDATA[<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<afisResp>
  <!--created by AFIS Connector 2019-11-05 JOB started Tue Nov 12 13:52:14 CET 2019-->
  <connector>
    <prefix>1</prefix>
    <title>Haupt StaatsArchiv</title>
    <server>ZDV-OS02</server>
    <db>HSTASDB</db>
    <port>1521</port>
```

```

<usr>aplkn_archv</usr>
<maxLoginTime>60</maxLoginTime>
<maxVE>300</maxVE>
<maxMinuteDelete>2880</maxMinuteDelete>
</connector>
<connector>
  <prefix>8</prefix>
  <title>TEST Server scope used by RLANG</title>
  <server>ZDV-OS02</server>
  <db>GBZADB</db>
  <port>1521</port>
  <usr>aplkn_archv_4</usr>
  <maxLoginTime>60</maxLoginTime>
  <maxVE>300</maxVE>
  <maxMinuteDelete>2880</maxMinuteDelete>
</connector>
<verbose/>
<!--JOB DONE Tue Nov 12 13:52:14 CET 2019 elapsed time = 6 ms-->
</afisResp>]]></prefixList>
  </ns1:getPrefixListResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

importDocAfis

Mit dieser Operation werden die Daten eines Controlfile in ein Archivinformationssystem (AFIS) übertragen. Strukturobjekte und Informationsobjekte werden im AFIS angelegt, Repräsentationen und Primärdaten werden nur in DIMAG gespeichert. Das Controlfile enthält die Metadaten, welche im AFIS angelegt werden und die Metadaten, welche in DIMAG angelegt werden. Das Controlfile und die Primärdateien müssen im Import-Verzeichnis des sftp-Servers liegen.

Request

username	String
password	String
usernameafis	String
passwordafis	String
ControlFile	String

Response

status	Integer
msg	String

Bsp. Request

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://dimag.la-bw.de/
WebService.wsdl">
  <soapenv:Header/>
  <soapenv:Body>
    <web:importDocAfis>
      <username>testuser</username>
      <password>mypass</password>
      <usernameafis>thesame</usernameafis>
      <passwordafis>anotherpass</passwordafis>
      <ControlFile>filepath/mycontrolfile_afis.xml</ControlFile>
    </web:importDocAfis>
  </soapenv:Body>
</soapenv:Envelope>

```

Bsp. Controlfile.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<verzeichnungseinheit>
  <rootid>8-15457</rootid>
  <verz-obj>
    <typ>O</typ>

```

```

<formid>10039</formid>
<fid-1>Soap Test-Akte</fid-1>
<verz-obj>
  <signatur-anteil>R1</signatur-anteil>
  <titel>Dimag-Titel</titel>
  <typ>R</typ>
  <verz-obj>
    <dateiname>Uploadfile.txt</dateiname>
    <sftp-dateiname>/filepath/Uploadfile.txt</sftp-dateiname>
    <typ>F</typ>
    <titel>Primärdatei-Titel</titel>
  </verz-obj>
</verz-obj>
</verz-obj>
</verzeichnungseinheit>

```

Bsp. Response

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://dimag.la-bw.de/
WebService.wsdl">
  <SOAP-ENV:Body>
    <ns1:importDocAfisResponse>
      <status>200</status>
      <msg><![CDATA[Success: importAfisDimag(): ok: Strukturobjekt 8-1 [] : [A0] DEMO Archiv updated<br/>
ok: Strukturobjekt 8-4229 [] : B1 Gruppen inserted<br/>
ok: Protokollobjekt test-6 [] : Protokoll zu 8-1
Testarchiv (gekoppelt) updated<br/>
ok: Strukturobjekt 8-4230 [] : B2 Beständegruppe inserted<br/>
ok: Protokollobjekt test-2218 [] : Protokoll zu 8-4229
B1 Gruppen updated<br/>
ok: Strukturobjekt 8-15456 [] : B7 inserted<br/>
ok: Protokollobjekt test-2219 [] : Protokoll zu 8-4230
B2 Beständegruppe updated<br/>
ok: Strukturobjekt 8-15457 [] : D0 inserted<br/>
ok: Protokollobjekt test-2220 [] : Protokoll zu 8-15456
B7 updated<br/>
ok: Informationsobjekt 8-16590 [] : Soap Test-Akte inserted<br/>
ok: Protokollobjekt test-2221 [] : Protokoll zu 8-15457
D0 updated<br/>
ok: Repräsentation test-2222 [A R1] : Dimag-Titel inserted<br/>
ok: Protokollobjekt test-2223 [] : Protokoll zu 8-16590
Soap Test-Akte updated<br/>
ok: Primärdatenobjekt test-2224 [] : Primärdatei-Titel inserted<br/>
ok: Protokollobjekt test-2225 [] : Protokoll zu test-2222
Dimag-Titel updated<br/>]]></msg>
    </ns1:importDocAfisResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

getForms

Diese Operation ermöglicht es, alle in einem AFIS hinterlegten Formulare (Erschließungsmasken) abzufragen.

Request

username	String
password	String
usernameafis	String
passwordafis	String
prefix	String

Response

status	Integer
msg	String

forms	String	forms xml
-------	--------	-----------

Bsp. Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://dimag.la-bw.de/WebService.wsdl">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getForms>
      <username>testuser</username>
      <password>mypass</password>
      <usernameafis>thesame</usernameafis>
      <passwordafis>anotherpass</passwordafis>
      <prefix>8</prefix>
    </web:getForms>
  </soapenv:Body>
</soapenv:Envelope>
```

Bsp. Response (Ausschnitt)

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://dimag.la-bw.de/WebService.wsdl">
  <SOAP-ENV:Body>
    <ns1:getFormsResponse>
      <status>200</status>
      <msg>Success</msg>
      <forms><![CDATA[<?xml version="1.0" encoding="UTF-8"?>
<afisResp>
  <!--created by AFIS Connector 2019-05-09
JOB started Tue Oct 29 08:09:29 CET 2019-->
  <form>
    <formId>10004</formId>
    <type>O</type>
    <label>Gliederungstitel</label>
  </form>
  <form>
    <formId>10005</formId>
    <type>O</type>
    <label>Standardformular</label>
  </form>
  <form>
    <formId>10008</formId>
    <type>O</type>
    <label>Bilddokument</label>
  </form>
  <form>
    <formId>10009</formId>
    <type>O</type>
    <label>Plandokument</label>
  </form>
  <form>
    <formId>10010</formId>
    <type>O</type>
    <label>Akte</label>
  </form>
</afisResp>]]></forms>
    </ns1:getFormsResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

getFields

Mit dieser Operation werden die Feldnamen eines Erschließungsformulars abgefragt.

Request

username	String
password	String
usernameafis	String
passwordafis	String
prefix	String
formid	String

Response

status	Integer
msg	String
fields	String fields xml

Bsp. Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://dimag.la-bw.de/WebService.wsdl">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getFields>
      <username>testuser</username>
      <password>mypass</password>
      <usernameafis>thesame</usernameafis>
      <passwordafis>anotherpass</passwordafis>
      <prefix>8</prefix>
      <formid>10004</formid>
    </web:getFields>
  </soapenv:Body>
</soapenv:Envelope>
```

Bsp. Response

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://dimag.la-bw.de/WebService.wsdl">
  <SOAP-ENV:Body>
    <ns1:getFieldsResponse>
      <status>200</status>
      <msg>Success</msg>
      <fields><![CDATA[<?xml version="1.0"?>
<fieldDescription>
  <field>
    <xmhtag>fid-2</xmhtag>
    <name>Klassifikation</name>
    <maxoccurs>0</maxoccurs>
    <required>>false</required>
    <length>4000</length>
    <readOnly>>false</readOnly>
    <description>Eindeutiger Referenzcode f&#xFC;r die Verzeichnungseinheit respektive Bestellsignatur (allenfalls mit L&#xE4;ndercode).
Reference code (E); R&#xE9;f&#xE9;rence (F); Segnatura archivistica (I)</description>
    <hint>Text (max. 4000 Z.)</hint>
    <layoutsequence>1</layoutsequence>
    <defaultValue/>
    <regexpr/>
  </field>
  <field>
    <xmhtag>fid-3</xmhtag>
    <name>Bestellsignatur</name>
    <maxoccurs>0</maxoccurs>
    <required>>false</required>
    <length>4000</length>
    <readOnly>>false</readOnly>

```

```

<hint>Text (max. 4000 Z.)</hint>
<layoutsequence>2</layoutsequence>
<defaultvalue/>
<regex/>
</field>
<field>
<xmltag>fid-1</xmltag>
<name>Titel</name>
<maxoccurs>0</maxoccurs>
<required>true</required>
<length>4000</length>
<readOnly>false</readOnly>
<description>Bezeichnung/Betreff einer Verzeichnungseinheit.
Title (E); Institut&#xE9; (F); titolo (I)</description>
<hint>= Bestandsbezeichnung = Klassifikations&#xFC;berschrift und -gliederungszahl = Regest = Haupttitel = Nachname,
Vornamen Text (max. 4000 Z.)</hint>
<layoutsequence>3</layoutsequence>
<defaultvalue/>
<regex/>
</field>
<field>
<xmltag>fid-6</xmltag>
<name>Stufe</name>
<maxoccurs>0</maxoccurs>
<required>false</required>
<length>4000</length>
<readOnly>true</readOnly>
<description>Zuordnung einer VE zu einer Verzeichnungsstufe.
Level of description (E); Niveau de description (F); Livello di descrizione (I)</description>
<hint>Text (max. 4000 Z.)</hint>
<layoutsequence>4</layoutsequence>
<defaultvalue/>
<regex/>
</field>
</fieldDescription>]]</fields>
</ns1:getFieldsResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

LoadXML

Als *LoadXML* wird in DIMAG ein XML-Format bezeichnet, das zum Austausch von Metadaten beim Import und Export zum Einsatz kommt. Sogenannte Controlfiles bei SOAP:importDoc() und SOAP:exportDoc() sind in diesem Format.

Die grundsätzliche Struktur von LoadXML ist folgende:

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<verzeichnungseinheit>
<rootid></rootid>
<verz-obj>
...Metadaten des Objekts...
<verz-obj>
...Metadaten des Objekts...
</verz-obj>
...
</verz-obj>
...
</verzeichnungseinheit>

```

Ein Minimalbeispiel (mit Titel und Typ als Pflichtfeld) könnte so aussehen:

```

<?xml version="1.0" encoding="UTF-8"?>
<verzeichnungseinheit>
<!--

```

The location for the following <verz-obj> is specified below this rootid:

```
-->
<rootid>D-1</rootid>
<verz-obj>
  <!-- no aID given, so DIMAG will create it -->
  <aid/>
  <!-- Title -->
  <titel>One new Entry</titel>
  <!-- Structure Element -->
  <typ>S</typ>
  <merkmal>-</merkmal>
</verz-obj>
</verzeichnungseinheit>
```

Eine von Ingesttool generierte LoadXML-Datei (Ausschnitt):

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<verzeichnungseinheit>
  <rootid/></rootid>
  <verz-obj>
    <aid/></aid>
    <alternate-id>2cc4556a-097d-44c7-a8de-44d9a283b1df</alternate-id>
    <archivalien-typ>Zeug</archivalien-typ>
    <entstehungs-zeitraum>2019</entstehungs-zeitraum>
    <titel>Bilder</titel>
    <typ>O</typ>
  </verz-obj>
  <aid>AID 111 DO1</aid>
  <alternate-id>ac0386eb-2e99-438f-b73e-c0639d0b8065</alternate-id>
  <titel>R1</titel>
  <typ>R</typ>
  <verz-obj>
    <aid/></aid>
    <aenderungs-datum-datei>2019-02-11 11:57:06</aenderungs-datum-datei>
    <alternate-id>7b6e265d-5015-4ea3-9acf-bcc3f4dc7df7</alternate-id>
    <datei-groesse-in-byte>10</datei-groesse-in-byte>
    <dateiname>Koala.jpg</dateiname>
    <filepath>Bilder/Ausstellungen</filepath>
    <md5>67a5720d85e7add20354d19d6013e263</md5>
    <sftp-dateiname>Bilder/Ausstellungen/Koala.jpg</sftp-dateiname>
    <titel>Koala.jpg</titel>
    <typ>F</typ>
  </verz-obj>
  <verz-obj>
    <aid/></aid>
    <aenderungs-datum-datei>2019-02-11 11:57:06</aenderungs-datum-datei>
    <alternate-id>9634a3f7-7b99-4a07-8322-b67b8c3a654f</alternate-id>
    <datei-groesse-in-byte>10</datei-groesse-in-byte>
    <dateiname>Lighthouse.jpg</dateiname>
    <filepath>Bilder/Ausstellungen</filepath>
    <md5>67a5720d85e7add20354d19d6013e263</md5>
    <sftp-dateiname>Bilder/Ausstellungen/Lighthouse.jpg</sftp-dateiname>
    <titel>Lighthouse.jpg</titel>
    <typ>F</typ>
  </verz-obj>
</verz-obj>
</verz-obj>
  <aid/></aid>
  <alternate-id>02dd69f7-0c79-4355-b5aa-46b47e00a74a</alternate-id>
  ...
```

Einhaengen von Objekten beim Import

Objekte werden folgendermassen eingehängt:

1. Wenn das Objekt eine <pid> angegeben hat, wird es dort eingehängt
2. Wenn durch die Verschachtelung eine implizite Hierarchie vorgegeben ist, wird diese umgesetzt
3. Wenn nichts davon der Fall ist, wird das Objekt unter <rootid> eingehängt

(<rootid> muss nur gesetzt sein, wenn der dritte Fall auftaucht. Sonst kann das Feld leer- oder weggelassen werden.)

Seit DIMAG-3 ist die Reihenfolge der Tags eines Objekts egal.

Die Verschachtelung (<verz-obj>) muss aber immer am Ende sein (Tail-Recursion).

Die Verschachtelung löst das Problem, eine Hierarchie aufzubauen für Objekte deren Elternobjekte im gleichen Zuge angelegt werden und darum noch keine AID haben. Wenn alle Objekte bereits eine AID haben, z.B. bei Updates, oder zumindest ihre Elternobjekte bekannt sind, können sie, unabhängig von ihrer Hierarchiebeziehung, flach nebeneinander gelegt werden, z.B. so:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<verzeichnisseinheit>
  <rootid></rootid>
  <verz-obj>
    <pid>test-55</pid>
    <aid>test-123</aid>
    ...Metadaten des Objekts...
  </verz-obj>
  <verz-obj>
    <pid>test-123</pid>
    <aid>test-9999</aid>
    ...Metadaten des Objekts...
  </verz-obj>
  <verz-obj>
    <pid>test-3</pid>
    <aid></aid>
    ...Metadaten des Objekts...
  </verz-obj>
  ...
</verzeichnisseinheit>
```

Insert vs. Update

In DIMAG-3:

1. Wenn für ein Objekt keine AID angegeben ist -> Insert (mit neu generierter AID)
2. Wenn eine AID angegeben ist:
 1. Wenn diese bereits in DIMAG existiert -> Update
 2. Wenn sie nicht existiert -> Insert (mit der angegebenen AID)

Metadaten beim Update

Feld angegeben mit Inhalt -> Wert wird gesetzt

Feld angegeben ohne Inhalt -> Wert wird entfernt (d.h. der leere angegebene Wert wird übernommen und damit genaugenommen auch von Fall 1 abgedeckt)

Feld fehlt -> Wert bleibt unverändert

Validierung

Das Schema ist in den XSDs *LaBW_IngExt_*.xsd* bzw. *LaBW_AccExt_*.xsd* im DIMAG-Code im Verzeichnis XML/ hinterlegt. Es kann auch über die DIMAG-Weboberfläche im Menü Dokumentation/XML-Schema eingesehen werden.

Die (generierte) XSD (IngExt bzw. AccExt) ist derzeit (2019-09, 3.0.x) sehr rudimentär. Sie prüft nur die grundsätzliche Verschachtelungsstruktur und die insgesamt erlaubten Felder. Sie hat aber kein Verständnis von Typen. Somit schränkt sie auch nicht ein, welcher Typ welche Metadatenfelder haben darf oder muss (z.B. "S" muss <merkmal> haben, darf aber kein <dateiname> haben). Auch Feldinhalte werden nicht validiert.