

Documentation

of the MyCoRe Maven module

dptbase-viaf

by

Jens Kupferschmidt, University of Leipzig, Germany

Version 2017-06

Leipzig, 13/04/2018

Abstract

This paper should document the VIAF module of the dptbase-module collection. This module collection is implemented to support all MyCoRe-applications used in scope of projects in Leipzig, Germany.

Changes

Version	Changes
2014-06-01	first implementation
2014-12-15	corrections
2015-10-22	corrections
2018-04-03	add sex and bibReferences

Table of content

1Purpose.....	4
2Data model in MyCoRe.....	5
2.1Data model.....	5
2.2Mapping with Marc21 for GND.....	5
2.3Mapping with Marc21 for LCNAF.....	7
3Overview.....	8
4XSLT API.....	9

1 Purpose

For a lot of applications we need a storage structure for authority file entries of big national libraries like the Library of Congress or the Deutsche Nationalbibliothek. On the other side, some applications store their own data on persons without adding them to an inventory in a global authority file systems. In order to enable both local storage as well as the implementation of international authority files, the most common fields for both are implemented in the data model. To convert data from the module to the library data model MARC21, we have an import / export interface. This way, you can migrate your own (local) data to an authority file system of the libraries.

The implementation is a MAVEN-module which can be used in a lot of different MyCoRe¹ applications. You can also use the editor subselect of this datamodel in your own editor forms.

1 see <http://www.mycore.de>

2 Data model in MyCoRe

2.1 Data model

fieldname	required	repeater	MCR-Type	remark
preferredName	yes	for lang	MCRMetaPersonName	
variantName	no	yes	MCRMetaPersonName	
recordType	yes	no	MCRMetaLangText	
recordSource	yes	no	MCRMetaClassification viafRecordSource	
identifier	no	for types	MCRMetaLangText	
firstDate	no	no	MCRMetaHistoryDate	
firstPlace	no	for lang	MCRMetaLangText	
firstPlacelIdentifier	no	no	MCRMetaLangText	
secondDate	no	no	MCRMetaHistoryDate	
secondPlace	no	for lang	MCRMetaLangText	
secondPlacelIdentifier	no	no	MCRMetaLangText	
gender	no	no	MCRMetaBoolean	
sex	yes	no	MCRMetaClassification viafSexISO5218	see Wikipedia ISO/IEC 5218
confession	no	no	MCRMetaClassification viafConfessionType	defined by Boris Liebreuz
profession	no	yes	MCRMetaLangText	
professionIdentifier	no	yes	MCRMetaLangText	
areaOfActivity	no	yes	MCRMetaLangText	
areaOfActivityIdentifier	no	yes	MCRMetaLangText	
floruit	no	no	MCRMetaHistoryDate	
literarySource	no	yes	MCRMetaLangText	
note	no	yes	MCRMetaLangText	
bibReference	no	yes	MCRMetaLangText	
internalRemark	no	yes	MCRMetaLangText	
link	no	yes	MCRMetaLink	
gndxml	no	no	MCRMetaXML	autofilled
lcnafxml	no	no	MCRMetaXML	autofilled
editor	no	yes	MCRMetaLangText	required in the future

Tabelle 1: Data model dptbase-viaf

2.2 Mapping with Marc21 for GND

fieldname	marc21 import	marc21 update
preferredName → fullname	datafield[@tag='100']/ subfield[@code = 'a'] or datafield[@tag='110']/ @ind1 = '1' and subfield[@code = 'b'] or @ind1 = '2' and ubfield[@code = 'a']	datafield[@tag='100']/ subfield[@code = 'a'] or datafield[@tag='110']/ @ind1 = '1' and subfield[@code = 'b'] or @ind1 = '2' and ubfield[@code = 'a']

fieldname	marc21 import	marc21 update
	or datafield[@tag='111']/ @ind1 = '2' and subfield[@code = 'a'] or datafield[@tag='151']/ subfield[@code = 'a']	or datafield[@tag='111']/ @ind1 = '2' and subfield[@code = 'a'] or datafield[@tag='151']/ subfield[@code = 'a']
preferredName → callname	datafield[@tag='100' and @ind1 ="0"]/ subfield[@code = 'a']	datafield[@tag='100' and @ind1 ="0"]/ subfield[@code = 'a']
preferredName → numeration	datafield[@tag='100' and @ind1 ="0"]/ subfield[@code = 'b']	datafield[@tag='100' and @ind1 ="0"]/ subfield[@code = 'b']
preferredName → title	datafield[@tag='100' and @ind1 ="0"]/ subfield[@code = 'c']	datafield[@tag='100' and @ind1 ="0"]/ subfield[@code = 'c']
variantName → fullname	datafield[@tag='400' and @ind1 ="1"]/ subfield[@code = 'a'] or datafield[@tag='410' and @ind1 ="1 or 2"]/ subfield[@code = 'a'] or datafield[@tag='411"]/ subfield[@code = 'a'] or datafield[@tag='451']/ subfield[@code = 'a']	datafield[@tag='400' and @ind1 ="1"]/ subfield[@code = 'a'] or datafield[@tag='410' and @ind1 ="1 or 2"]/ subfield[@code = 'a'] or datafield[@tag='411"]/ subfield[@code = 'a'] or datafield[@tag='451']/ subfield[@code = 'a']
variantName → callname	datafield[@tag='400' and @ind1 ="0"]/ subfield[@code = 'a']	datafield[@tag='100' and @ind1 ="0"]/ subfield[@code = 'a']
variantName → numeration	datafield[@tag='400' and @ind1 ="0"]/ subfield[@code = 'b']	datafield[@tag='400' and @ind1 ="0"]/ subfield[@code = 'b']
variantName → title	datafield[@tag='400' and @ind1 ="0"]/ subfield[@code = 'c'] or datafield[@tag='410' and @ind1 ="0"]/ subfield[@code = 'b']	datafield[@tag='400' and @ind1 ="0"]/ subfield[@code = 'c'] or datafield[@tag='410' and @ind1 ="0"]/ subfield[@code = 'b']
recordType	fix entry tag 100 → 'person' fix entry tag 110 → 'person' fix entry tag 111 → 'person' fix entry tag 151 → 'person'	copy from old dataset
recordSource	fix categid 'lcnaf' if data from LCNAF fix categid 'gnd' if data from GND	copy from old dataset
identifier	controlfield[@tag='001'] with (588...)ID or blank ID without (copy from old dataset
firstDate		
firstPlace	datafield[@tag='551']/subfield[@code='i' and text() = 'Geburtsort']][@code='a']	copy from old dataset
firstPlaceIdentifier	datafield[@tag='551']/subfield[@code='i' and text() = 'Geburtsort']][@code='0']	datafield[@tag='551']/subfield[@code='i' and text() = 'Geburtsort']][@code='0']
secondDate		
secondPlace	datafield[@tag='551']/subfield[@code='i' and text() = 'Sterbeort']][@code='a']	copy from old dataset
secondPlaceIdentifier	datafield[@tag='551']/subfield[@code='i' and text() = 'Sterbeort']][@code='0']	datafield[@tag='551']/subfield[@code='i' and text() = 'Sterbeort']][@code='0']
gender	datafield[@tag='375']/subfield[@code='a'] 1 – man / 2 - woman	datafield[@tag='375']/subfield[@code='a'] 1 – man / 2 - woman
profession		copy from old dataset
professionIdentifier		

fieldname	marc21 import	marc21 update
areaOfActivity	not set	copy from old dataset
areaOfActivityIdentifier	not set	copy from old dataset
literarySource	not set	copy from old dataset
note	not set	copy from old dataset
link	not set	copy from old dataset
gndxml	complete XML tree if GND	complete XML tree if GND
lcnafxml	not set	not set

Current problems:

- language detection in Latin script (no ISO lang codes for names)
- very simple date notation
- descriptors in German like 'Exakte Lebensdaten'

2.3 Mapping with Marc21 for LCNAF

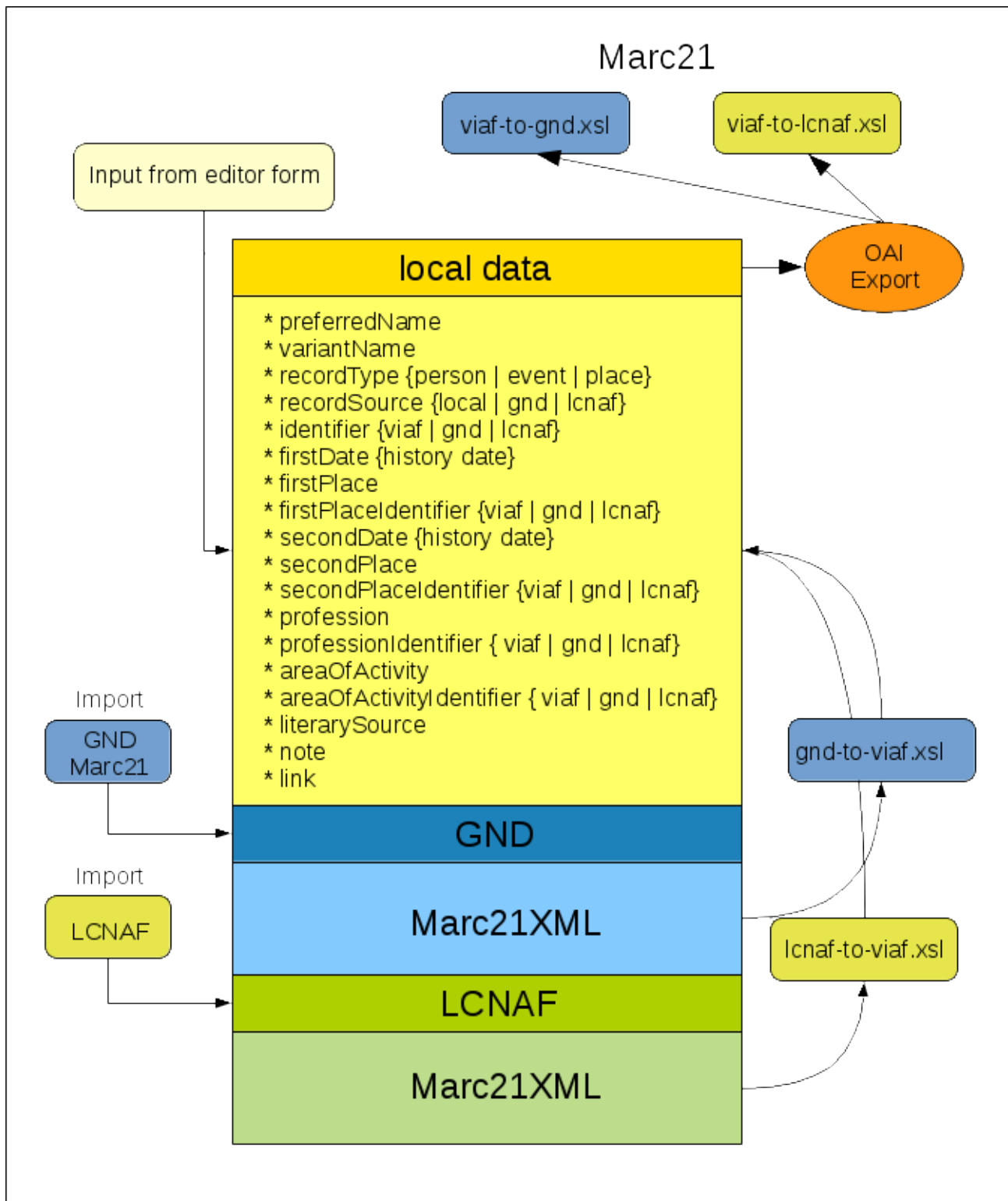
fieldname	marc21 import	marc21 update
preferredName → fullname	datafield[@tag='100']/subfield[@code = 'a']	datafield[@tag='100']/subfield[@code = 'a']
preferredName → callname	datafield[@tag='100' and @ind1 = "0"]/subfield[@code = 'a']	datafield[@tag='100' and @ind1 = "0"]/subfield[@code = 'a']
preferredName → numeration	datafield[@tag='100' and @ind1 = "0"]/subfield[@code = 'b']	datafield[@tag='100' and @ind1 = "0"]/subfield[@code = 'b']
preferredName → title	datafield[@tag='100' and @ind1 = "0"]/subfield[@code = 'c']	datafield[@tag='100' and @ind1 = "0"]/subfield[@code = 'c']
variantName → fullname	datafield[@tag='400' and @ind1 = "1"]/subfield[@code = 'a']	datafield[@tag='400' and @ind1 = "1"]/subfield[@code = 'a']
variantName → callname	datafield[@tag='100' and @ind1 = "0"]/subfield[@code = 'a']	datafield[@tag='100' and @ind1 = "0"]/subfield[@code = 'a']
variantName → numeration	datafield[@tag='100' and @ind1 = "0"]/subfield[@code = 'b']	datafield[@tag='100' and @ind1 = "0"]/subfield[@code = 'b']
variantName → title	datafield[@tag='100' and @ind1 = "0"]/subfield[@code = 'c']	datafield[@tag='100' and @ind1 = "0"]/subfield[@code = 'c']
recordType	fix entry 'person'	copy from old dataset
recordSource	fix catetid 'lcnaf' if data from LCNAF	copy from old dataset
identifier	controlfield[@tag='001']	copy from old dataset
firstDate		
firstPlace	datafield[@tag='551']/subfield[@code='i' and text() = 'Geburtsort']][@code='a']	copy from old dataset
firstPlaceIdentifier	datafield[@tag='551']/subfield[@code='i' and text() = 'Geburtsort']][@code='0']	datafield[@tag='551']/subfield[@code='i' and text() = 'Geburtsort']][@code='0']
secondDate		
secondPlace	datafield[@tag='551']/subfield[@code='i' and text() = 'Sterbeort']][@code='a']	copy from old dataset
secondPlaceIdentifier	datafield[@tag='551']/subfield[@code='i' and text() = 'Sterbeort']][@code='0']	datafield[@tag='551']/subfield[@code='i' and text() = 'Sterbeort']][@code='0']
gender	datafield[@tag='375']/subfield[@code='a'] 1 – man / 2 - woman	datafield[@tag='375']/subfield[@code='a'] 1 – man / 2 - woman
profession		copy from old dataset
professionIdentifier		
areaOfActivity	not set	copy from old dataset
areaOfActivityIdentifier	not set	copy from old dataset
literarySource	not set	copy from old dataset
note	not set	copy from old dataset

fieldname	marc21 import	marc21 update
link	not set	copy from old dataset
gndxml	not set	not set
lcnafxml	complete XML tree if LCNAF	complete XML tree if LCNAF

Current problems:

- special characters in LCNAF names (see Saladin)

3 Overview



4 XSLT API

For the integration of the **dptbase-viaf** module in your MyCoRe application XSLT templates are provided that you can use/apply in your own XSLT templates. All templates use the parameter *mycoreobject_node* and *next*.

template	decription
printVIAFFirstDate	print the first date text in the current language
printVIAFFirstPlace	print the first place as text in the current language or as output of the Identifier data link
printVIAFSecondDate	print the second date text in the current language
printVIAFSecondPlace	print the second place as text in the current language or as output of the Identifier data link
printVIAFAreaOfActivity	print the area of activity as text in the current language or as output of the Identifier data link
printVIAFProfession	print the profession as text in the current language or as output of the Identifier data link