



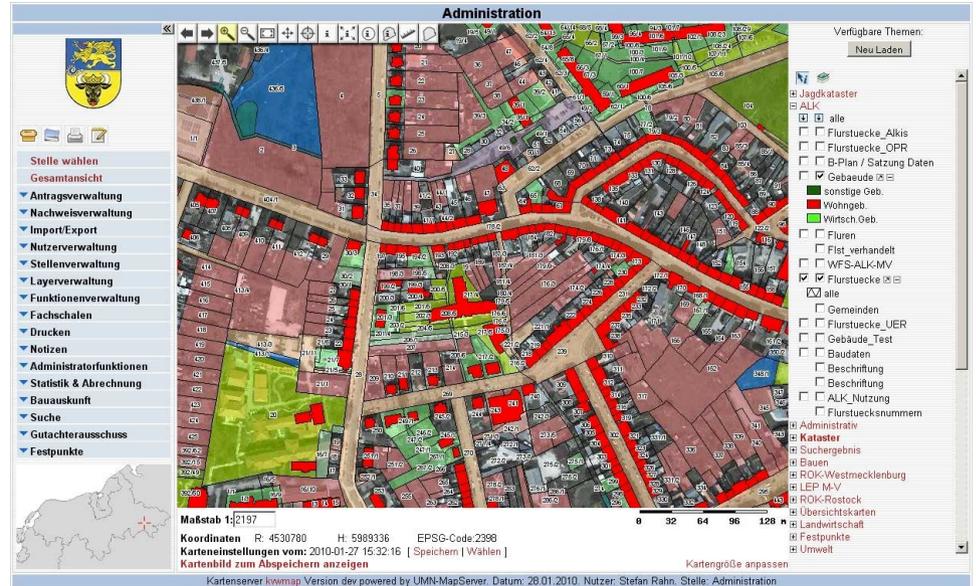
Vervollständigung des PostNAS-Schemas und die Anpassung des NAS-Imports - an Hand des AAA-Implementierungsmodells

Dr.-Ing. Peter Korduan, GDI-Service Rostock
Salzburg, 04.07.2016



GDI Service Rostock

Softwareentwicklung
Schulung
Support
Installation
Hosting



WebGIS kwmap



Fahrzeug Online Tracking



Problemstellung

Aktuelles PostNAS Datenbankschema für ogr2ogr umfasst nicht das vollständige AAA-Modell

- 1) Folge: NAS-GML-Dateien werden nicht vollständig in die Datenbank übernommen
- 2) Es fehlen Tabellen und Attribute
- 3) ogr2ogr macht komplexe XML-Elemente falsch => doppelte Namen
- 4) PostgreSQL hat Begrenzung in Länge der Tabellen und Attributnamen => doppelte Namen



Projekt beauftragt von:

- Landkreis Vorpommern-Rügen, Stralsund
 - Markus Hentschel
- Land Rheinland-Pfalz
 - Landesamt für Vermessung und Geobasisinformation Rheinland-Pfalz, Koblenz
 - Stefan Schliebner
- Saarland
 - Landeamt für Vermessung, Geoinformation und Landentwicklung, Saarbrücken



Lösungsansatz

- 1) AAA-UML-Anwendungsschema in Implementierungsschema konvertieren (shapechange)
- 2) Modellelemente des Implementierungsschemas in Datenbank einlesen (xmi2db)
- 3) Datenbankschema der GML-Klassen erzeugen (db2classes)
- 4) Datenmodell für ogr2ogr ableiten (classes2ogr)
- 5) Filter zur Anpassung des vollständigen ogr Modells
- 6) Umbenennungsscript für einzulesenden NAS-Dateien unter Anwendung des Filters (nas2ogrgml)



Aufgabenstellung

pkorduan / xmi2db Unwatch 3

[Code](#) [Issues 9](#) [Pull requests 0](#) [Wiki](#) [Pulse](#) [Graphs](#) [Settings](#)

Filters [Labels](#) [Milestones](#)

Clear current search query, filters, and sorts

6 Open 2 Closed Author Labels Milestones

- NAS-6 Dokumentation und Koordinierung mit dem PostNAS Projekt** enhancement
#15 opened on 31 Mar by pkorduan [PostNAS](#)
- NAS-5 Erstellung eines Umbenennungsskriptes (nas2ogrgml)** enhancement
#14 opened on 31 Mar by pkorduan [PostNAS](#)
- NAS-4 Erstellung des Datenmodells für den ogr Import (classes2ogr)** enhancement
#13 opened on 31 Mar by pkorduan [PostNAS](#)
- NAS-3 UML-Schema in Klassenschema überführen (db2classes)** enhancement
#12 opened on 31 Mar by pkorduan [PostNAS](#)
- NAS-2 UML-Modell in Datenbank einlesen (xmi2db)** enhancement
#11 opened on 31 Mar by pkorduan [PostNAS](#)
- NAS-1 UML-Modell in Implementierungsschema konvertieren (ShapeChange)** enhancement
#10 opened on 31 Mar by pkorduan [PostNAS](#)



NAS-1 UML-Modell in Implementierungsschema konvertieren (ShapeChange)

- Laut GeoInfoDok 6.0.1, Abschnitt 4.2 Normen und Standards, muss das konzeptionelle UML-Modell des AAA-Anwendungsschemas in ein Implementierungsschema konvertiert werden um ein Modell zu erhalten, welches zur NAS Schnittstelle passt.
- Die Erzeugung des Implementierungsschemas erfolgt mit dem Plug-In ShapeChange und der dazugehörigen Konfigurationsdatei mit den Konvertierungsregeln für das AAA-Modell. Der Export erfolgt aus Enterprise Architect heraus in eine Metadata Interchange Datei (XMI) in der Version 1.0.
- Ergebnis:
<http://gdi-service.de/xmi2db/xmis/>

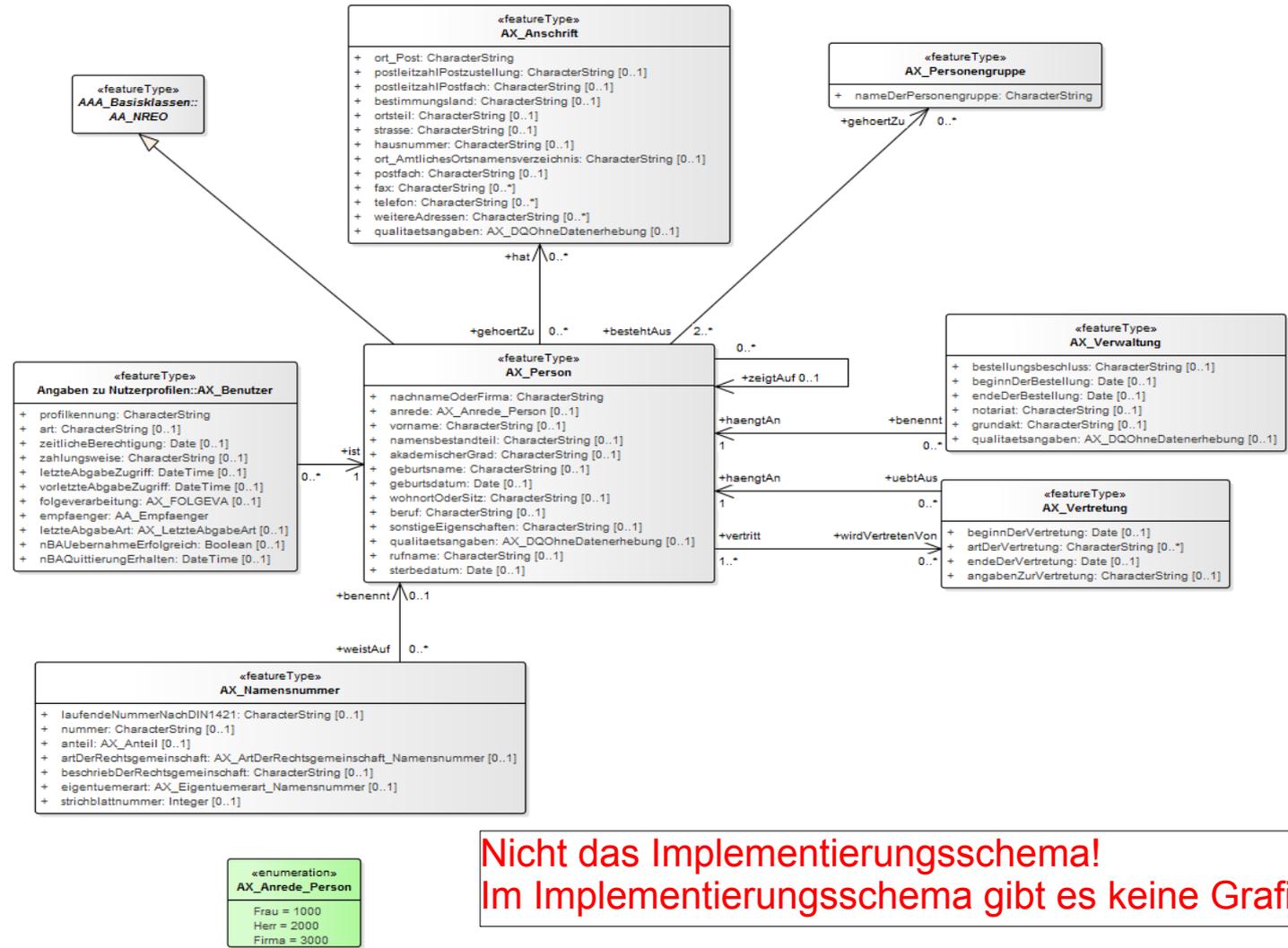
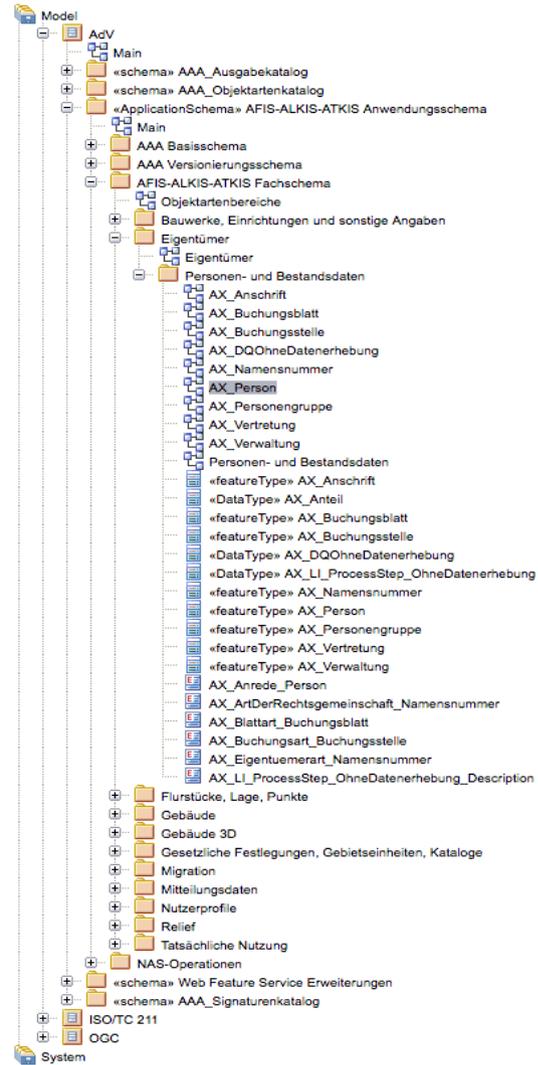
EA-Katalog vom AAA-UML-Modell



http://gdi-service.de/public/schemas/aaa/ea_catalog_uml_modell/index.htm



Model



**Nicht das Implementierungsschema!
Im Implementierungsschema gibt es keine Grafiken!**

EA-Katalog vom Implementierungsmodell

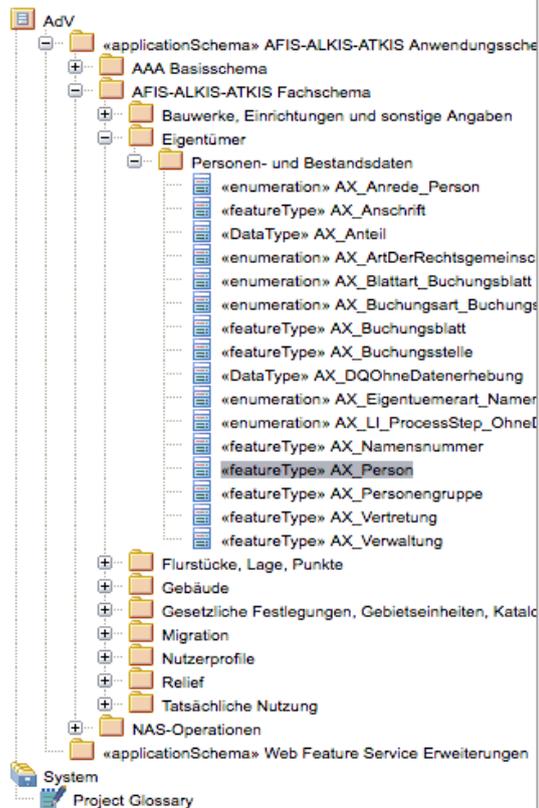


http://gdi-service.de/public/schemas/aaa/ea_catalog_implementierungsmodell/index.htm

gdi-service.de/public/schemas/aaa/ea_catalog_implementierungsmodell/index.htm



AdV



AX_Person : Public <<featureType>> Class

Created: 30.09.2009 12:11:04
Modified: 22.03.2010 20:05:20

Project:
Advanced:

[E] 'Person' ist eine natürliche oder juristische Person und kann z.B. in den Rollen Eigentümer, Erwerber, Verwalter oder Vertreter in Katasterangelegenheiten geführt werden.

=== Bildungsregel ===

Die Relationsarten 'weist_auf' und/oder 'benennt' und/oder 'übt_aus' sowie die inverse Relationsart zum 'Benutzer' sind objektbildend. Eine dieser Relationen muss vorhanden sein. Diese Relationen sind nicht zu verwenden, wenn auf die 'Person' die rekursive Relation 'zeigtAuf' zeigt. In diesem Fall ist 'zeigtAuf' objektbildend.

=== Erfassungskriterium Basis-DLM ===

Es werden Personen nur im Zusammenhang mit Nutzerprofilen erfasst.

=== Erfassungskriterium DLM50 ===

Attributes Associations To Associations From Tagged Values Other Links

Element	Source Role	Target Role
«featureType» AX_Gebaeude Class	Name: gehoert 'Gebäude' gehört 'Person'. Die Relation kommt nur vor, wenn unabhängig von Eintragungen im Grundbuch ('Buchungsstelle' mit der Attributart 'Buchungsart') für das Gebäude ein Eigentum nach BGB begründet ist.	Name: besitzt
Details:		
«featureType» AX_Benutzer Class	Name: ist 'Benutzer' ist 'Person'.	Name: inversZu_ist
Details:		
«featureType» AX_Vertretung Class	Name: haengtAn Die Relation 'Vertretung' hängt an 'Person' sagt	Name: uebtAus Die 'Person' übt die 'Vertretung' in



AAA-Implementierungsmodell XMI-Datei

```
<?xml version="1.0" encoding="windows-1252"?>
<XMI xmi.version="1.2" xmlns:UML="org.omg.xmi.namespace.UML" timestamp="2016-04-01 12:28:10">
  <XMI.header>
    <XMI.documentation>
      <XMI.exporter>Enterprise Architect</XMI.exporter>
      <XMI.exporterVersion>4.1</XMI.exporterVersion>
    </XMI.documentation>
    <XMI.metamodel xmi.name="UML" xmi.version="1.4"/>
  </XMI.header>
  <XMI.content>
    <UML:Model name="AFIS-ALKIS-ATKIS Anwendungsschema" xmi.id="MX_EAID_56A9BE2"
isRoot="true" isLeaf="false" isAbstract="false" isSpecification="false" visibil
    >
      <UML:Namespace.ownedElement>
        <UML:Stereotype xmi.id="EAID_8D679B4A_61EC_4a54_AD65_975DC641970D" name
isSpecification="false" isRoot="false" isLeaf="false" isAbstract="false">
          <UML:Stereotype.baseClass>Package</UML:Stereotype.baseClass>
        </UML:Stereotype>
        <UML:ModelElement.stereotype>
          <UML:Stereotype xmi.idref="EAID_8D679B4A_61EC_4a54_AD65_975DC641970D"
          </UML:ModelElement.stereotype>
        <UML:TagDefinition xmi.id="EAID_CB537825_ACF0_4f2e_89E4_CB383678C045" n
tagType="String"/>
        <UML:TagDefinition xmi.id="EAID_8D045149_9AF3_4225_982A_628F3F1ABAB1" n
tagType="String"/>
        <UML:TagDefinition xmi.id="EAID_9E609666_49F7_40f7_AF0E_C9217497BCEE" name="modified" isSpecification="false"
tagType="String"/>
        <UML:TagDefinition xmi.id="EAID_FB6F3144_63C6_469d_B138_A26B98342253" name="iscontrolled"
isSpecification="false" tagType="String"/>
        <UML:TagDefinition xmi.id="EAID_732DB4EF_3501_408a_BA6C_999914D5B560" name="lastloaddate"
isSpecification="false" tagType="String"/>
        <UML:TagDefinition xmi.id="EAID_B1C500F8_BA05_41f4_ADFE_23252DF5F603" name="lastsavedate"
isSpecification="false" tagType="String"/>
        <UML:TagDefinition xmi.id="EAID_B7984F59_EE4B_4530_B8AB_07A96DF2364E" name="isprotected"
isSpecification="false" tagType="String"/>
        <UML:TagDefinition xmi.id="EAID_D69A8521_7558_46a3_B9F7_2A944CF59723" name="usedtd" isSpecification="false"
tagType="String"/>
        <UML:TagDefinition xmi.id="EAID_B7EF684C_7EC3_4017_9450_BD06F8D015DC" name="logxml" isSpecification="false"
tagType="String"/>
        <UML:TagDefinition xmi.id="EAID_86CA2F3C_9107_4e9e_9BB5_990D58F3224C" name="tpos" isSpecification="false"
tagType="String"/>
        <UML:TagDefinition xmi.id="EAID_DC2BC432_C335_4185_AE8D_E5452FE1A814" name="batchsave" isSpecification="false"
tagType="String"/>
      </UML:Namespace.ownedElement>
    </UML:Model>
  </XMI.content>
</XMI>
```

gdi-service.de/xmi2db/xmis/

Index of /xmi2db/xmis

Name	Last modified	Size	Description
Parent Directory		-	
2016-04-01 ImplModell AAA-xmi12-uml14.xmi	28-Jun-2016 19:53	42M	

Apache/2.2.16 (Debian) Server at gdi-service.de Port 80

NAS-2 UML-Modell in Datenbank einlesen (xmi2db)



- Die im Schritt NAS-1 erzeugte XMI-Datei repräsentiert das Implementierungsschema als UML-Modell in Form von XML.
- Es wird das Skript xmi2db weiterentwickelt, mit dem die UML-Dinge (Klassen, Generalisierungen, Typen, Assoziationen etc.) in eine PostgreSQL Datenbank geschrieben werden. Im Ergebnis gibt es die befüllten Tabellen für UML-Klassen, UML-Attribute, Klassengeneralisierungen, Beziehungen, Stereotypen und Datentypen. Die Inhalte dieser Datenbank bilden das gesamte ALKIS Schema als Meta-Schema ab.
- Das Ergebnis ist
 1. PHP-Skript xmi2db und
 2. ein Datenbankschema alkis_uml gefüllt mit den entsprechenden Modellelementen.



xmi2db

- xmi-Datei auswählen
- Zielschema auswählen
- Befüllung starten



xmi2db

xmi2db überträgt die UML-Modell Elemente der ausgewählten xmi Datei in das ausgewählte Datenbank Schema. Eingelesen werden nur die Elemente ab dem ausgewählten Basispaket.

Dateiauswahl

Zur Auswahl weiterer Dateien diese vorher auf dem Server in das Unterverzeichnis xmis dieser Anwendung ablegen.

Schemaauswahl/-eingabe

Das Schema wird entsprechend der gewählten Konfiguration in der Datenbank "kvwmapsp" angelegt.

BasePackageauswahl/-eingabe

Bei einem EA-Export des XPlan-Modells "XPlanGML 4.1" wählen, bei einem ArgoUML Export leer lassen oder ein Package eintragen, falls man nur das eine laden möchte.

- Tabellen vor dem Einlesen leeren
- Argo Export mit ISO19136 Profil

✓ Fülle DB mit XML Inhalten



```
# Packages of AAA Schema~
> $packages [] := 'AAA_Basisschema';~
> $packages [] := 'AAA_Basisklassen';~
> $packages [] := 'AAA_GemeinsameGeometrie';~
> $packages [] := 'AAA_Nutzerprofile';~
> $packages [] := 'AAA_Operationen';~
> $packages [] := 'AAA_Praesentationsobjekte';~
> $packages [] := 'AAA_Praesentationsobjekte_3D';~
> $packages [] := 'AAA_Projektsteuerung';~
> $packages [] := 'AAA_Punktmengenobjekte';~
> $packages [] := 'AAA_Spatial_Schema';~
> $packages [] := 'AAA_Spatial_Schema_3D';~
> $packages [] := 'AAA_Unabhaengige_Geometrie';~
> $packages [] := 'AAA_Unabhaengige_Geometrie_3D';~
> $packages [] := 'Codelisten';~
> $packages [] := 'AFIS-ALKIS-ATKIS_Fachschemata';~
> $packages [] := 'Bauwerke, Einrichtungen und sonstige Angaben';~
> $packages [] := 'Bauwerke und Einrichtungen in Siedlungsflächen';~
> $packages [] := 'Bauwerke, Anlagen und Einrichtungen für den Verkehr';~
> $packages [] := 'Besondere Angaben zum Gewässer';~
> $packages [] := 'Besondere Angaben zum Verkehr';~
> $packages [] := 'Besondere Anlagen auf Siedlungsflächen';~
> $packages [] := 'Besondere Eigenschaften von Gewässern';~
> $packages [] := 'Besondere Vegetationsmerkmale';~
> $packages [] := 'Eigentümer';~
> $packages [] := 'Personen- und Bestandsdaten';~
> $packages [] := 'Flurstücke, Lage, Punkte';~
> $packages [] := 'Angaben zu Festpunkten der Landesvermessung';~
```



← → ↻ 🏠 gdi-service.de/xmi2db/use-xmi2db.php?truncate=1&file=x

Übergeordnetes:

```
8 Stereotypes in die DB geschrieben.  
100 TagDefinitions in die DB geschrieben.  
71 DataTypes in die DB geschrieben.
```

Package: AAA Basisschema

```
AAA Basisschema  
AAA_Basisklassen  
  Objektbereich: AAA_Basisklassen  
    Klasse: AA_Fachdatenobjekt  
      16 taggedValues in die DB geschrieben.  
      2 Attribute in die DB geschrieben.  
    Klasse: AA_NREO  
      13 taggedValues in die DB geschrieben.  
      0 Attribute in die DB geschrieben.  
    Klasse: AA_ZUSO  
      13 taggedValues in die DB geschrieben.  
      0 Attribute in die DB geschrieben.  
    Klasse: AA_Objekt  
      13 taggedValues in die DB geschrieben.  
      4 Attribute in die DB geschrieben.  
    Klasse: AA_REO  
      14 taggedValues in die DB geschrieben.  
      0 Attribute in die DB geschrieben.  
    Klasse: AA_AdVStandardModell  
      8 taggedValues in die DB geschrieben.  
      20 Attribute in die DB geschrieben.  
    Klasse: AA_Lebenszeitintervall  
      13 taggedValues in die DB geschrieben.  
      2 Attribute in die DB geschrieben.  
    Klasse: AA_Fachdatenverbindung  
      13 taggedValues in die DB geschrieben.  
      2 Attribute in die DB geschrieben.  
    Klasse: AA_Modellart  
      16 taggedValues in die DB geschrieben.  
      2 Attribute in die DB geschrieben.  
9 Klassen in die DB geschrieben.  
59 Generalisierungen in die DB geschrieben.  
10 Assoziationen in die DB geschrieben.
```



UML Tabellen in Schema aaa_uml

- aaa_uml
 - Aggregate (0)
 - Sortierfolgen (0)
 - Konversionen (0)
 - Fremdtabellen (0)
 - Funktionen (0)
 - Operatoren (0)
 - Sequenzen (11)
 - Tabellen (11)
 - association_classes
 - association_ends
 - class_generalizations
 - comments
 - datatypes
 - packages
 - stereotypes
 - tagdefinitions
 - taggedvalues
 - uml_attributes
 - uml_classes**
 - Spalten (15)
 - Constraints (1)
 - Indizes (0)
 - Regeln (0)
 - Trigger (0)
 - Triggerfunktionen (0)

FULLfaktor	
Zeilen (gezählt)	698
Erbt Tabellen	Nein
Anzahl geerbter Tabellen	0
nicht geloggt?	Nein
Hat OIDs?	Nein
System-Tabelle?	Nein
Kommentar	


```
SQL-Feld
-- Table: aaa_uml.uml_classes
-- DROP TABLE aaa_uml.uml_classes;

CREATE TABLE aaa_uml.uml_classes
(
  xmi_id CHARACTER VARYING(255),
  NAME CHARACTER VARYING(255),
  visibility CHARACTER VARYING(255),
  "isSpecification" BOOLEAN,
  "isRoot" BOOLEAN,
  "isLeaf" BOOLEAN,
  "isActive" BOOLEAN,
  package_id INTEGER,
  model_id INTEGER,
  created_at TIMESTAMP WITHOUT TIME ZONE,
  updated_at TIMESTAMP WITHOUT TIME ZONE,
  "isAbstract" BOOLEAN,
  id INTEGER NOT NULL DEFAULT nextval('aaa_uml.uml_classes_id2_seq'::regclass),
  stereotype_id CHARACTER VARYING,
  general_id CHARACTER VARYING,
  CONSTRAINT uml_classes_pkey PRIMARY KEY (id)
)
```



Tabelle uml_classes

Daten editieren - GDI-Service kvwmap (81.169.181.149:5432) - kvwmapsp - aaa_uml.uml_classes

Keine Begrenz

	xmi_id character varying(255)	name character varying(255)	vis ch	is be	is be	is be	package_id integer	m in	cr ti	isAbstract boolean	id [PK] integer	stereotype_id character varying
1	EAID_2DFFAE30_E22F_4500_A64F_8F1179DC4E26	AA_Fachdatenobjekt	pl	F	F	F	2	20		FALSE	1	EAID_96EAAAC73_153A_4c14_8E6D_52F3F21A35D7
2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	AA_NREO	pl	F	F	F	2	20		TRUE	2	EAID_78D417A1_EAC6_41ac_8907_E0DBE6DC03B3
3	EAID_3849AA05_D7D6_401c_A44E_A1C6BF548495	AA_ZUSO	pl	F	F	F	2	20		TRUE	3	EAID_78D417A1_EAC6_41ac_8907_E0DBE6DC03B3
4	EAID_4A9703AA_8236_4cb8_B5AB_41496B8C35B1	AA_Objekt	pl	F	F	F	2	20		TRUE	4	EAID_78D417A1_EAC6_41ac_8907_E0DBE6DC03B3
5	EAID_5F5503C3_936A_4ad3_8439_01B156E06521	AA_REO	pl	F	F	F	2	20		TRUE	5	EAID_78D417A1_EAC6_41ac_8907_E0DBE6DC03B3
6	EAID_7EF9E9CB_3C70_4db3_9A8D_933DA5C3F30D	AA_AdVStandardModell	pl	F	F	F	2	20		FALSE	6	EAID_DBC6065F_5E90_4ff0_92D0_8C475F46C387
7	EAID_A0DB4035_6EEE_451f_A2D0_9D9E0BA2EAC1	AA_Lebenszeitintervall	pl	F	F	F	2	20		FALSE	7	EAID_BED119C1_311A_4a74_996D_121184388A0F
8	EAID_A2E9D8B9_47C7_4d69_8A07_CD4057B72488	AA_Fachdatenverbindung	pl	F	F	F	2	20		FALSE	8	EAID_BED119C1_311A_4a74_996D_121184388A0F
9	EAID_C245707C_4485_4a73_92D3_8150770A145B	AA_Modellart	pl	F	F	F	2	20		FALSE	9	EAID_96EAAAC73_153A_4c14_8E6D_52F3F21A35D7
10	EAID_1B8A24B3_EEC8_4afa_8834_B68C236098C0	AG_Linienobjekt	pl	F	F	F	3	20		TRUE	10	EAID_78D417A1_EAC6_41ac_8907_E0DBE6DC03B3
11	EAID_91FA2763_B701_44cf_BBB4_91BEE74CFFDD	AG_Objekt	pl	F	F	F	3	20		TRUE	11	EAID_78D417A1_EAC6_41ac_8907_E0DBE6DC03B3
12	EAID_CA6A6F3D_7855_49b7_8D7F_7A895F1B0293	AG_Punktobjekt	pl	F	F	F	3	20		TRUE	12	EAID_78D417A1_EAC6_41ac_8907_E0DBE6DC03B3
13	EAID_F74BD922_E127_43b5_8790_990E44FD2A8E	AG_Flaechenobjekt	pl	F	F	F	3	20		TRUE	13	EAID_78D417A1_EAC6_41ac_8907_E0DBE6DC03B3
14	EAID_231E7867_2489_41ac_B872_1FEA1863F377	AA_Benutzer	pl	F	F	F	4	20		TRUE	14	EAID_78D417A1_EAC6_41ac_8907_E0DBE6DC03B3
15	EAID_790E22CA_4625_4d59_AF8D_29E058482C95	AA_Benutzergruppe	pl	F	F	F	4	20		TRUE	15	EAID_78D417A1_EAC6_41ac_8907_E0DBE6DC03B3
16	EAID_0A327D95_C5F5_4a7b_8613_C58B00AC9B5F	AA_Koordinatenreferenzsystemar	pl	F	F	F	5	20		FALSE	16	EAID_BED119C1_311A_4a74_996D_121184388A0F
17	EAID_108EB205_31AC_4370_8DE4_BC672E15A702	AA_NAS_Ausgabeform	pl	F	F	F	5	20		FALSE	17	EAID_DBC6065F_5E90_4ff0_92D0_8C475F46C387
18	EAID_172601BD_EE39_4bab_82BE_1FE2296E13BD	DCP	pl	F	F	F	5	20		FALSE	18	EAID_B5282FE1_1CAD_41b5_A82B_9EAA8F0299B3
19	EAID_22146C13_98BF_4a75_8731_08FF84D79147	AA_Themendefinition	pl	F	F	F	5	20		TRUE	19	EAID_BED119C1_311A_4a74_996D_121184388A0F
20	EAID_398F050D_901B_445a_89E6_EBCF33AF8B0A	NAS_Filter_Capabilities	pl	F	F	F	5	20		FALSE	20	EAID_DBC6065F_5E90_4ff0_92D0_8C475F46C387
21	EAID_418B3300_852F_4aa2_9316_FF95A28BE50E	ExceptionFortfuehrung	pl	F	F	F	5	20		FALSE	21	EAID_BED119C1_311A_4a74_996D_121184388A0F
22	EAID_48DB7A0D_46BB_4d3a_853E_51AAE0350915	AA_Benutzungsauftrag	pl	F	F	F	5	20		FALSE	22	EAID_B5282FE1_1CAD_41b5_A82B_9EAA8F0299B3
23	EAID_5C416F62_72E3_43b2_90E5_9FDEA8393AC4	AA_Themendimension	pl	F	F	F	5	20		FALSE	23	EAID_DBC6065F_5E90_4ff0_92D0_8C475F46C387

698 Zeilen.



Tabelle uml_attributes

xmi_id character varying(255)	name character varying(255)	m in	uml_class_id integer	vis ct	ov ct	cl ct	ta ct	or ct	cr ct	uj ct	id serial	datatype character va	classifier character varying	m ct	m ct	multip charac	multiplicity_range_upper character varying	in ct	initialvalue_body character varying
EAID_E439E753_E16B_	name	1		pl							1	'	EAID_52D63F12_39C	E/	E/	1	1	E/	'
EAID_D4F8938C_8DC9_	uri	1		pl							2	eaxmiid0	'	E/	E/	1	1	E/	'
EAID_13545985_463B_	anlass	4		pl							3	eaxmiid1	'	E/	E/	0	2	E/	'
EAID_84027B0A_F9AD_	lebenszeitintervall	4		pl							4	'	EAID_A0DB4035_6EE	E/	E/	1	1	E/	'
EAID_0F9681DD_5FB9_	modellart	4		pl							5	eaxmiid2	'	E/	E/	1	*	E/	'
EAID_93AA2671_36FF_	zeigtAufExternes	4		pl							6	eaxmiid3	'	E/	E/	0	*	E/	'
EAID_EEFEE252_7CE7_	LiegenschaftskatasterMode	6		pl							7	eaxmiid4	'	E/	E/	1	1	E/	DLKM
EAID_F23C0E96_4457_	KatasterkartenModell500	6		pl							8	eaxmiid4	'	E/	E/	1	1	E/	DKKM500
EAID_6814465F_EBE9_	KatasterkartenModell1000	6		pl							9	eaxmiid4	'	E/	E/	1	1	E/	DKKM1000
EAID_1D65EF17_1429_	KatasterkartenModell2000	6		pl							10	eaxmiid4	'	E/	E/	1	1	E/	DKKM2000
EAID_4525F49C_F925_	KatasterkartenModell5000	6		pl							11	eaxmiid4	'	E/	E/	1	1	E/	DKKM5000
EAID_F85A1621_F1E2_	BasisLandschaftsModell	6		pl							12	eaxmiid4	'	E/	E/	1	1	E/	Basis-DLM
EAID_FC72F9BF_AAF9_	LandschaftsModell50	6		pl							13	eaxmiid4	'	E/	E/	1	1	E/	DLM50
EAID_F2E85CDD_1119_	LandschaftsModell250	6		pl							14	eaxmiid4	'	E/	E/	1	1	E/	DLM250
EAID_ABC557C9_C2F5_	LandschaftsModell1000	6		pl							15	eaxmiid4	'	E/	E/	1	1	E/	DLM1000
EAID_B87252E0_01C6_	TopographischeKarte10	6		pl							16	eaxmiid4	'	E/	E/	1	1	E/	DTK10
EAID_994B4A5A_707E_	TopographischeKarte25	6		pl							17	eaxmiid4	'	E/	E/	1	1	E/	DTK25
EAID_CCC1B71D_0748_	TopographischeKarte50	6		pl							18	eaxmiid4	'	E/	E/	1	1	E/	DTK50
EAID_394FC618_FD5F_	TopographischeKarte100	6		pl							19	eaxmiid4	'	E/	E/	1	1	E/	DTK100
EAID_EC18A9B2_8C20_	TopographischeKarte250	6		pl							20	eaxmiid4	'	E/	E/	1	1	E/	DTK250
EAID_BA1FBD9B_7FD0_	TopographischeKarte1000	6		pl							21	eaxmiid4	'	E/	E/	1	1	E/	DTK1000
EAID_5FBCAC74_218D_	Festpunktmodell	6		pl							22	eaxmiid4	'	E/	E/	1	1	E/	DFGM
EAID_1A4B7C67_3559_	DigitalesGelaendemodell2	6		pl							23	eaxmiid4	'	E/	E/	1	1	E/	DGM2
EAID_21D736B9_C225_	DigitalesGelaendemodell5	6		pl							24	eaxmiid4	'	E/	E/	1	1	E/	DGM5



Tabelle class_generalizations

xmi_id character varying(255)	name character	is_builtin boolean	package_id integer	parent_id character varying	child_id character varying	creation_time timestamp	id serial
EAID_8283A0AC_447E_4225_902C_24938AE852D2	<l	F	2	EAID_4A9703AA_8236_4cb8_B5AB_41496B8C35B1	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	20	1
EAID_013F8256_8648_4be5_87A4_552586968CAC	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_2E4CASC3_2345_4bb7_8FF0_DA2F3E256166	20	2
EAID_02170D31_839B_41ab_97D5_3B2B89AE69F8	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_D33E84DD_BD18_45ef_9B49_F2AD6AC4432B	20	3
EAID_0CA7FEE8_8B3C_48f2_B94B_FDF8DE5F7C08	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_231E7B67_2489_41ac_B872_1FEA1863F377	20	4
EAID_0F1D760C_0B37_442f_9D00_342A4FAA165C	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_088CC668_9F2C_488b_8095_000364CE59F5	20	5
EAID_18C4D1D1_2000_4bee_9CCF_2D0E9A812EC3	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_EFD1279E_2F8B_4c10_B5E0_40BC0688AC27	20	6
EAID_1D45896C_27C7_4a62_808B_47F0FBE262C4	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_55F6F289_0488_4f94_B818_B13A54AF1BC8	20	7
EAID_2650AA16_6CDF_40b6_8B18_3C044DA6E3CD	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_FD2C86FD_4B19_459b_840B_CC9CFD4425A3	20	8
EAID_28016FE3_C61E_49ac_9A9D_4B83D01BF0D0	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_B57F3E70_A73A_4021_8703_45A1456CADE3	20	9
EAID_2CDB812B_C82E_4af8_878D_3340D8747169	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_A511A02E_668C_4dfc_BC99_BF39B0898DFB	20	10
EAID_320755FA_7BD3_4878_A3A8_C6E29A90EB8C	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_EDA6B92F_769A_43c4_9E9E_584132A80BB7	20	11
EAID_35721CFC_C684_426d_BA22_CBC642394805	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_D116BE95_8F06_4685_91AF_3E450A85C00D	20	12
EAID_4A164246_8338_4410_BECD_5C7DA7466E5A	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_42434985_0ABB_4719_BCC3_D4E2D916A5CB	20	13
EAID_51C6EBB3_C829_4154_9115_74798762D338	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_2D5AF80B_E2CD_4b4d_A434_A79CC0AAF921	20	14
EAID_5B408FFF_4019_4efc_9ED4_B613FC578D9B	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_365CE55B_9775_4103_A260_F5B0F75492BF	20	15
EAID_5F6D8F00_F9EA_4ea3_90F5_4997A853F460	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_1669654B_F947_4fda_9BE8_6DACA802E650	20	16
EAID_6C43893A_9E7E_4e7b_820B_29E692C1F6E8	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_C38C0EAA_35D9_49f7_A94F_DC191697C456	20	17
EAID_77DC8898_DEB2_47fd_B272_BC294116D89B	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_258D09E5_845E_4c9c_B7CB_74D08F985D47	20	18
EAID_894BC00C_31E0_4a0a_B1CE_30C441AD6A9F	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_96DFC89D_4C9E_481b_B1A7_41E3F25CFF5B	20	19
EAID_8C2B8137_0319_4374_9B6C_2B6428039A4E	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_714981E9_463D_40be_A01B_02B850FDE5EA	20	20
EAID_93D78036_3CB4_45cb_86E3_74865016C4B2	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_DA61315B_18A9_40eb_A2DA_5AD60B774F79	20	21
EAID_94584D8C_3B66_43d2_8D95_01ECFD2DF344	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_D386B3AF_B3C8_4664_B7EF_AB65C07A7622	20	22
EAID_94B152B7_2E36_4a1b_97A7_551BF6D80FCE	<l	F	2	EAID_32ACFF3C_8857_4c4a_8A9D_37893C565305	EAID_C28678CA_E853_48f4_BEDB_060CC4C6DF99	20	23



NAS-3 UML-Schema in Klassenschema überführen (db2classes)

- Aus aaa_uml Datenbank =>
- Pro Datatype ein Datentyp in Postgres
- Pro CodeListe, Enumeration und FeatureType je eine Tabelle
- Generalisierte Tabellen erben alle Attribute von Generalisierung
- AAA-Datentypen als Typen der FeatureType Attribute
- Attribute mit Multiplizität > 1 => Arrays
- Assoziationen als Attribute der FeatureTypen
- Das Ergebnis ist
 1. PHP-Skript db2classes und
 2. ein Datenbankschema alkis_gml mit allen Klassen und Beziehungen die es in ALKIS gibt.



db2classes

← → ↻ 🏠 📄 gdi-service.de/xmi2db/db2classes/

Get Enumerations

```
SELECT
    c.id,
    c.xmi_id,
    c.name
FROM
    aaa_uml.packages p LEFT JOIN
    aaa_uml.uml_classes c ON p.id = c.package_id LEFT JOIN
    aaa_uml.stereotypes s ON c.stereotype_id = s.xmi_id
WHERE
    lower(s.name) = 'enumeration' AND
    p.name IN ('AAA Basisschema', 'AAA_Basisklassen', 'AAA_GemeinsameGeometrie', 'AAA_Nutzerprofile', 'AAA_Operationen', 'AA
```



Abfrage der Attribute einer Enumeration

Create Enumeration: AA_AdVStandardModell (EAID_7EF9E9CB_3C70_4db3_9A8D_933DA5C3F30D)
Get Attributes:

```
SELECT
    a.name AS name,
    CASE
        WHEN d.name IS NULL THEN cc.name
        ELSE d.name
    END AS datatype,
    CASE
        WHEN d.name IS NULL THEN cs.name
        ELSE ds.name
    END AS stereotype,
    CASE
        WHEN d.name IS NULL THEN CASE
            WHEN cs.name IS NULL THEN NULL
            ELSE 'UML-Classifier'
        END
        ELSE 'UML-DataType'
    END AS attribute_type,
    a.multiplicity_range_lower::integer,
    a.multiplicity_range_upper,
    a.initialvalue_body
FROM
    aaa_uml.uml_classes c JOIN
    aaa_uml.uml_attributes a ON c.id = a.uml_class_id LEFT JOIN
    aaa_uml.datatypes d ON a.datatype = d.xmi_id LEFT JOIN
    aaa_uml.uml_classes dc ON d.name = dc.name LEFT JOIN
    aaa_uml.stereotypes ds ON dc.stereotype_id = ds.xmi_id Left JOIN
    aaa_uml.uml_classes cc ON a.classifier = cc.xmi_id LEFT JOIN
    aaa_uml.stereotypes cs ON cc.stereotype_id = cs.xmi_id
WHERE
    uml_class_id = 6
```



Table Create Statement und Inserts

```
CREATE TABLE IF NOT EXISTS aa_advstandardmodell (  
    wert character varying,  
    beschreibung character varying,  
    CONSTRAINT aa_advstandardmodell_pkey PRIMARY KEY (wert)  
) WITH OIDS;
```

```
INSERT INTO aa_advstandardmodell (wert,beschreibung) VALUES  
( 'DLKM', 'LiegenschaftskatasterModell'),  
( 'DKKM500', 'KatasterkartenModell500'),  
( 'DKKM1000', 'KatasterkartenModell1000'),  
( 'DKKM2000', 'KatasterkartenModell2000'),  
( 'DKKM5000', 'KatasterkartenModell5000'),  
( 'Basis-DLM', 'BasisLandschaftsModell'),  
( 'DLM50', 'LandschaftsModell50'),  
( 'DLM250', 'LandschaftsModell250'),  
( 'DLM1000', 'LandschaftsModell1000'),  
( 'DTK10', 'TopographischeKarte10'),  
( 'DTK25', 'TopographischeKarte25'),  
( 'DTK50', 'TopographischeKarte50'),  
( 'DTK100', 'TopographischeKarte100'),  
( 'DTK250', 'TopographischeKarte250'),  
( 'DTK1000', 'TopographischeKarte1000'),  
( 'DFGM', 'Festpunktmodell'),  
( 'DGM2', 'DigitalesGelaendemodell2'),  
( 'DGM5', 'DigitalesGelaendemodell5'),  
( 'DGM25', 'DigitalesGelaendemodell25'),  
( 'DGM50', 'Digitales Gelaendemodell50');
```



Abfragen und Erzeugen der CodeListen

Get CodeList

```
SELECT
    c.id,
    c.name,
    c.xmi_id
FROM
    aaa_uml.packages p LEFT JOIN
    aaa_uml.uml_classes c ON p.id = c.package_id LEFT JOIN
    aaa_uml.stereotypes s ON c.stereotype_id = s.xmi_id
WHERE
    s.name LIKE '%odeList' AND
    p.name IN ('AAA Basisschema', 'AAA_Basisklassen', 'AAA_GemeinsameGeometrie', 'AAA_Nutzerprofile', 'AAA_Opera
```

CodeList: AA_BesondereMeilensteinkategorie (EAID_5E3C3EA9_23D6_4310_9FFD_9C9253D33E7A)

```
CREATE TABLE IF NOT EXISTS aa_besonderemeilensteinkategorie (
    id integer,
    name character varying,
    status character varying,
    definition text,
    additional_information text,
    CONSTRAINT aa_besonderemeilensteinkategorie_pkey PRIMARY KEY (id)
) WITH OIDS;
```



GML Klassen in Schema aaa_classes

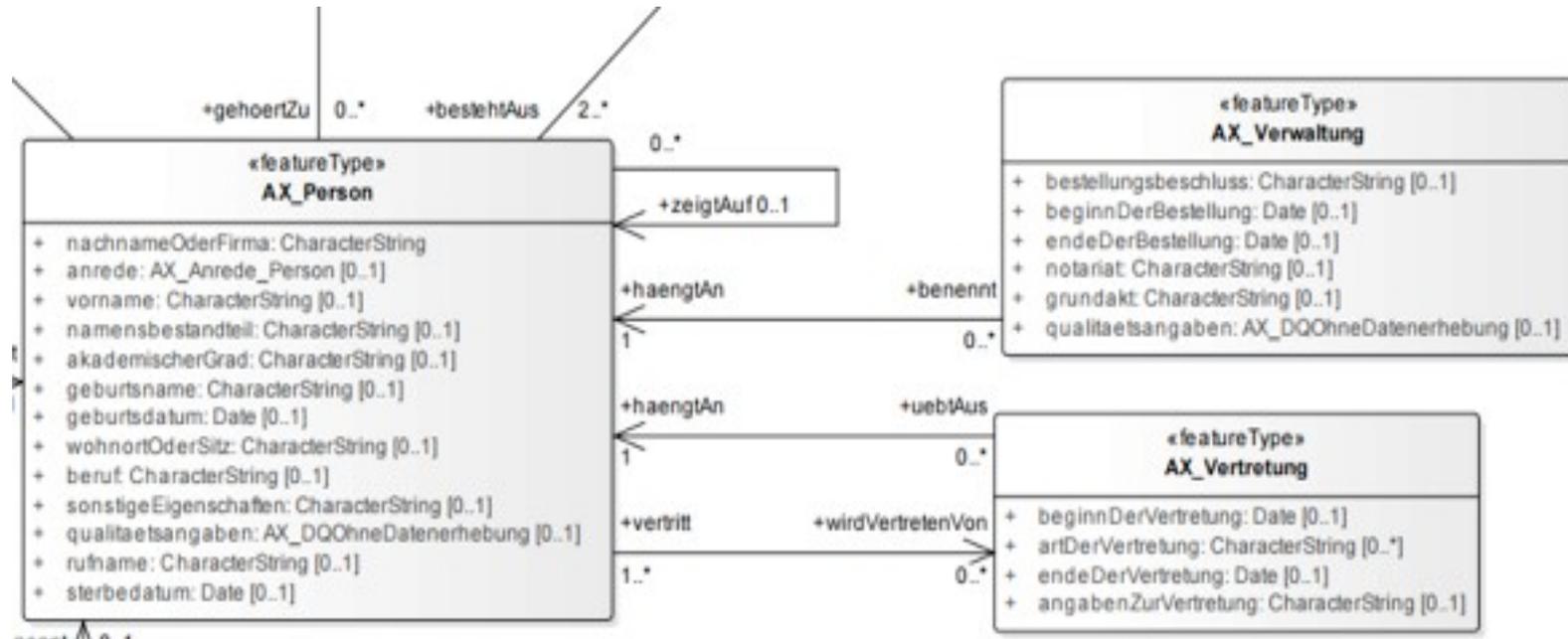
- ▼ aaa_classes
 - Aggregate (0)
 - Sortierfolgen (0)
 - Konversionen (0)
 - Fremdtabellen (0)
 - Funktionen (0)
 - Operatoren (0)
 - Sequenzen (0)
 - ▼ Tabellen (310)
 - aa_advstandardmodell
 - aa_aktivitaet
 - aa_anlassart
 - aa_anlassart_benutzungsauftrag
 - aa_antrag
 - aa_antragsgebiet
 - aa_art_themendefinition
 - ▼ aa_benutzer
 - ▼ Spalten (6)
 - gml_id
 - lebenszeitintervall
 - zeigtaufexternes
 - modellart
 - anlass
 - inverszu_wer
 - ▶ Constraints (1)
 - Indizes (0)
 - Regeln (0)
 - Trigger (0)
 - aa_benutzergruppe
 - aa_besonderemeilensteinkategorie
 - aa_instanzenthemem

Zeilen (geschätzt)	0
Füllfaktor	
Zeilen (gezählt)	0
Erbt Tabellen	Ja
Anzahl geerbter Tabellen	1
Geerbte Tabellen	aa_nreo
nicht geloggt?	Nein
Hat OIDs?	Ja
System-Tabelle?	Nein

```
SQL-Feld
-- Table: aaa_classes.aa_benutzer
-- DROP TABLE aaa_classes.aa_benutzer;
CREATE TABLE aaa_classes.aa_benutzer
(
-- Geerbt from table aaa_classes.aa_nreo: gml_id text NOT NULL,
-- Geerbt from table aaa_classes.aa_nreo: lebenszeitintervall uuid NOT NULL,
-- Geerbt from table aaa_classes.aa_nreo: zeigtaufexternes text,
-- Geerbt from table aaa_classes.aa_nreo: modellart text NOT NULL,
-- Geerbt from table aaa_classes.aa_nreo: anlass text,
  inverszu_wer CHARACTER VARYING[], -- featureType
  CONSTRAINT aa_benutzer_pkey PRIMARY KEY (gml_id)
)
INHERITS (aaa_classes.aa_nreo)
WITH (
  OIDS=TRUE
);
ALTER TABLE aaa_classes.aa_benutzer
  OWNER TO kvwmap;
COMMENT ON TABLE aaa_classes.aa_benutzer
  IS 'Tabelle AA_Benutzer abgeleitet von AA_NREO';
COMMENT ON COLUMN aaa_classes.aa_benutzer.inverszu_wer IS 'featureType';
```



Umsetzung von 1:n Associations



Class	Assoc	Multiplicity	Class name	Stereotyp
AX_Person	inversZu_zeigtAuf	*	AX_Person	featureType
AX_Person	zeigtAuf	1	AX_Person	featureType
AX_Person	weistAuf	*	AX_Namensnummer	featureType
AX_Person	benennt	*	AX_Verwaltung	featureType
AX_Person	uebtAus	*	AX_Vertretung	featureType
AX_Person	inversZu_ist	*	AX_Benutzer	featureType

inversZu_zeigtAuf..text[]
 zeigtAuf.....text
 benennt.....text[]
 uebtAus.....text[]
 wirdVertretenVon...text[]
 ...



Abfragen der 1:n Relationen

```
SELECT
    ca.name a_class_name,
    b.id b_id,
    b.name b_name,
    b.multiplicity_range_lower b_multiplicity_range_lower,
    b.multiplicity_range_upper b_multiplicity_range_upper,
    a.id a_id,
    a.name a_name,
    a.multiplicity_range_lower a_multiplicity_range_lower,
    a.multiplicity_range_upper a_multiplicity_range_upper,
    cb.name b_class_name,
    sb.name b_class_stereotype
FROM
    aaa_uml.uml_classes ca JOIN
    aaa_uml.association_ends a ON (ca.xmi_id = a.participant) JOIN
    aaa_uml.association_ends b ON (a.assoc_id = b.assoc_id) JOIN
    aaa_uml.uml_classes cb ON (cb.xmi_id = b.participant) JOIN
    aaa_uml.stereotypes sb ON (cb.stereotype_id = sb.xmi_id)
WHERE
    a.id != b.id
    AND NOT (b.multiplicity_range_upper = '-1' AND a.multiplicity_range_upper = '-1')
    AND ca.name = 'AX_Person'
```

a_class_name character varying(255)	b_id integer	b_name character varying	b_multiplicity_range_lower character varying	b_multiplicity_range_upper character varying	a_id integer	a_name character varying	a_multiplicity_range_lower character varying	a_multiplicity_range_upper character varying	b_class_name character varying(255)	b_class_stereotype character varying(255)
AX_Person	128	inversZu_zeigtAuf	0	-1	127	zeigtAuf	0	1	AX_Person	featureType
AX_Person	127	zeigtAuf	0	1	128	inversZu_zeigtAuf	0	-1	AX_Person	featureType
AX_Person	126	weistAuf	0	-1	125	benennt	0	1	AX_Namensnummer	featureType
AX_Person	124	benennt	0	-1	123	haengtAn	1	1	AX_Verwaltung	featureType
AX_Person	130	uebtAus	0	-1	129	haengtAn	1	1	AX_Vertretung	featureType
AX_Person	132	inversZu_ist	0	-1	131	ist	1	1	AX_Benutzer	featureType



NAS-4 Erstellung des Datenmodells für den ogr Import (classes2ogr)

- OGR-Bibliothek unterstützt nur flaches Datenschema --> verschachtelte Typen „flach machen“:
 - verschachtelte Attribute als zusätzliche Attribute in die Tabelle des FeatureTyps übernehmen

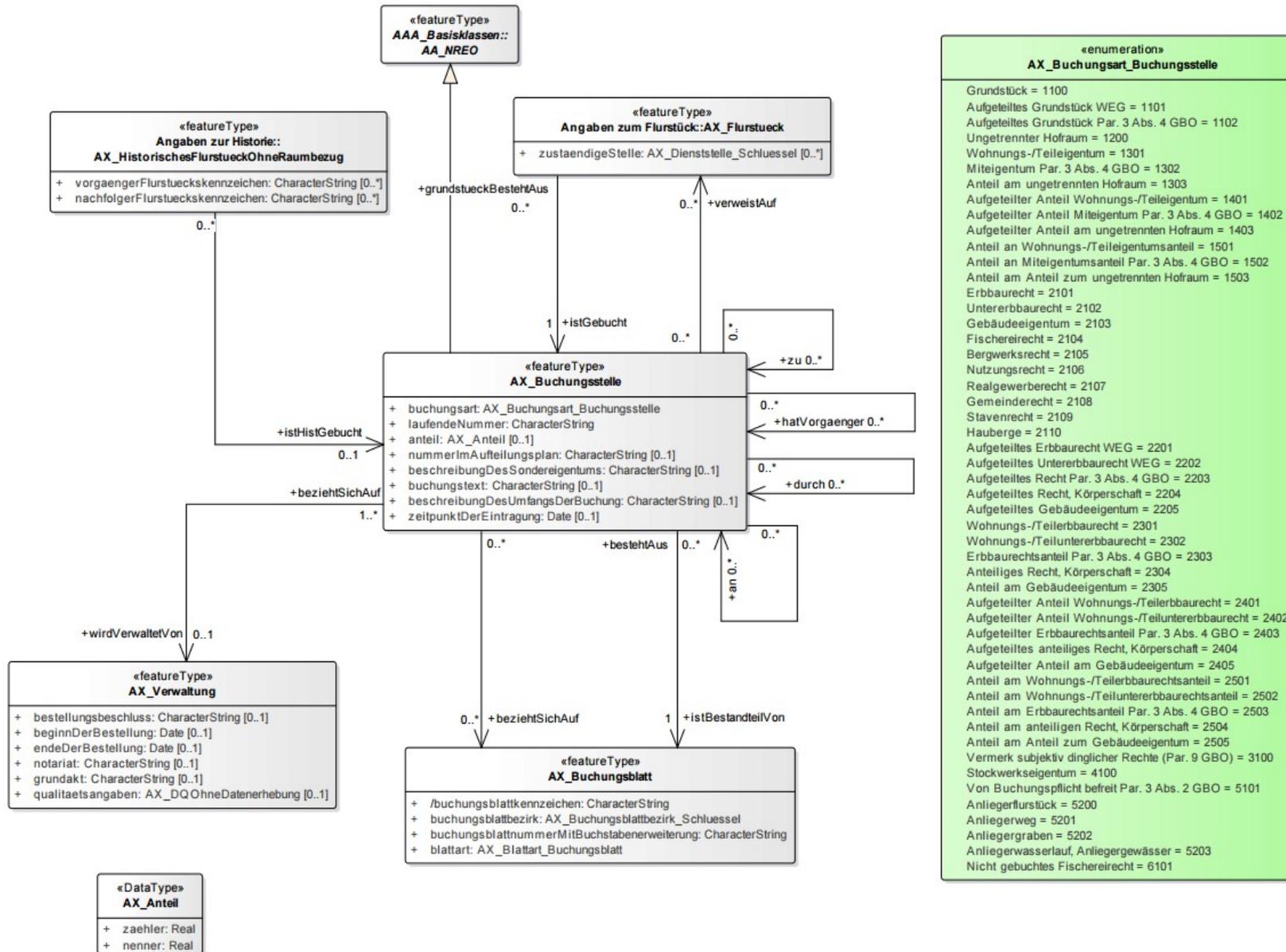
```
<AX_Flurstueck gml:id="DEMVAL76Z0001XfS">
  <anlass>000000</anlass>
  <lebenszeitintervall>
    <AA_Lebenszeitintervall>
      <beginnt>2014-08-01T07:57:17Z</beginnt>
    </AA_Lebenszeitintervall>
  </lebenszeitintervall>
</AX_Flurstueck>
```



```
CREATE TABLE ax_flurstueck(
  anlass VARCHAR[],
  beginnt CHARACTER(20),
  ...
)
```

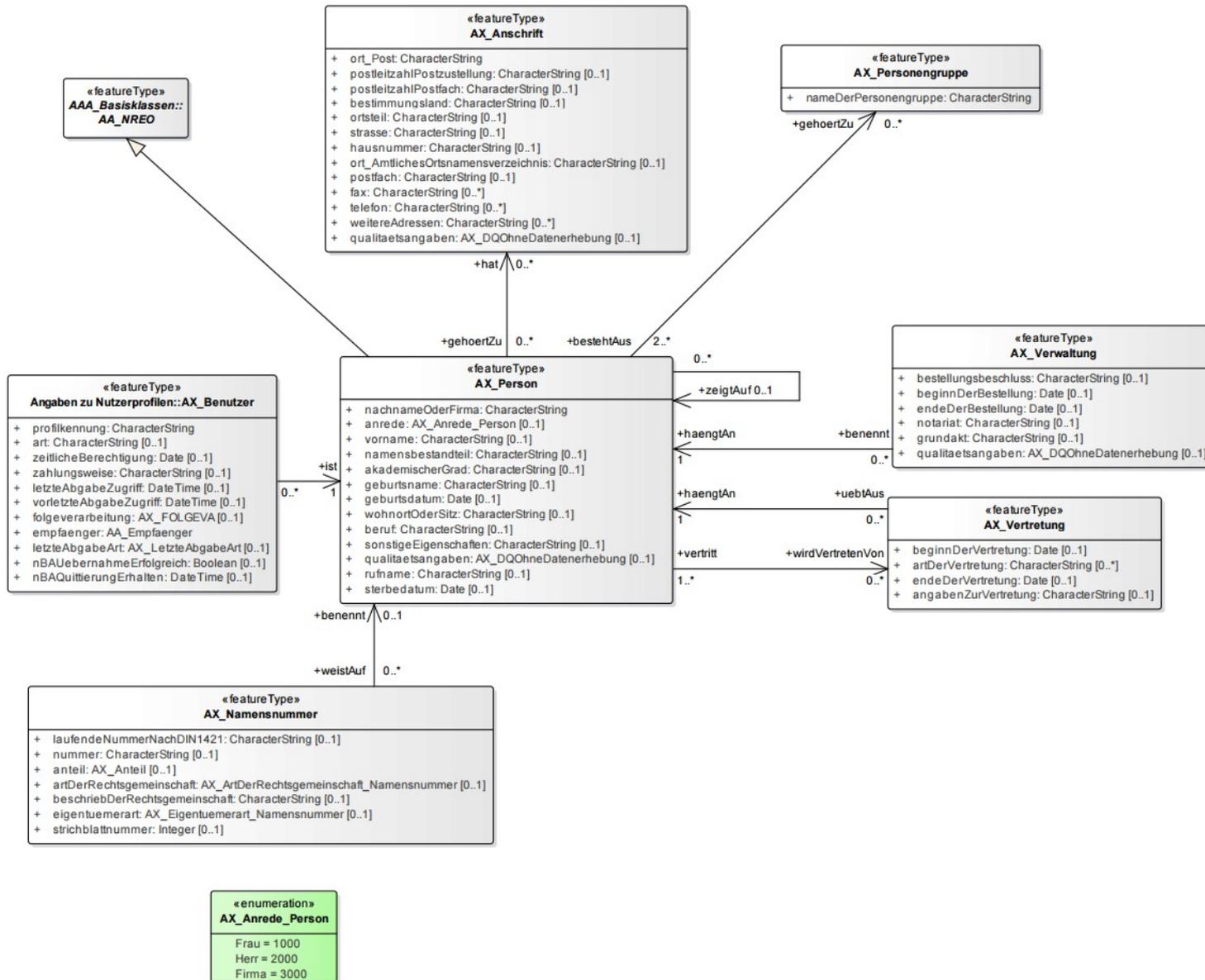


Class diagram AX_Buchungsstelle





Class diagram AX_Person





NAS-4 Erstellung des Datenmodells für den ogr Import (classes2ogr)

- OGR-Bibliothek unterstützt nur flaches Datenschema --> verschachtelte Typen „flach machen“:
 - verschachtelte Attribute als zusätzliche Attribute in die Tabelle des FeatureTyps übernehmen
 - prüfen ob es Doppelungen bei den Attributnamen gibt --> Umbenennungsliste

```
<AX_Flurstueck gml:id="DEMVAL76Z0001Xfs">
  <gemarkung>
    <AX_Gemarkung_Schluessel>
      <land>13</land>
      <gemarkungsnummer>1054</gemarkungsnummer>
    </AX_Gemarkung_Schluessel>
  </gemarkung>
  <zustaendigeStelle>
    <AX_Dienststelle_Schluessel>
      <land>13</land>
      <stelle>4087</stelle>
    </AX_Dienststelle_Schluessel>
  </zustaendigeStelle>
</AX_Flurstueck>
```



```
CREATE TABLE ax_flurstueck(
  land INTEGER,
  gemarkungsnummer INTEGER,
  land INTEGER,
  stelle INTEGER,
  ...
)
```



Vorschlag für Umbenennung

- gemarkung_land
- gemarkungsnummer
- zustaendige_stelle_land
- stelle

```
<AX_Flurstueck gml:id="DEMVAL76Z0001XFS">  
  <gemarkung>  
    <AX_Gemarkung_Schluessel>  
      <land>13</land>  
      <gemarkungsnummer>1054</gemarkungsnummer>  
    </AX_Gemarkung_Schluessel>  
  </gemarkung>  
  <zustaendigeStelle>  
    <AX_Dienststelle_Schluessel>  
      <land>13</land>  
      <stelle>4087</stelle>  
    </AX_Dienststelle_Schluessel>  
  </zustaendigeStelle>  
</AX_Flurstueck>
```

Comment on ax_f lurstueck.gemarkung_land IS

'Modellpfad:

gemarkung/AX_Gemarkung_Schluessel/land'

Comment on ax_f lurstueck.zustaendige_stelle_land

IS 'Modellpfad:

zustaendige_stelle/AX_Dienststelle_Schluessel/land



Umbenennungsbeispiel in NAS GML

```
<AX_Flurstueck id="DEMVAL76Z0001Xfs">
```

```
- <gemarkung>
```

```
- <AX_Gemarkung_Schlüssel>
```

```
<land>13</land>
```



```
<gemarkungsnummer>1054</gemarkungsnummer>
```

```
</AX_Gemarkung_Schlüssel>
```

```
</gemarkung>
```

```
- <zustaendigeStelle>
```

```
- <AX_Dienststelle_Schlüssel>
```

```
<land>13</land>
```



```
<stelle>4087</stelle>
```

```
</AX_Dienststelle_Schlüssel>
```

```
</zustaendigeStelle>
```

```
<AX_Flurstueck id="DEMVAL76Z0001Xfs">
```

```
- <gemarkung>
```

```
- <AX_Gemarkung_Schlüssel>
```

```
<gemarkung_land>13</gemarkung_land>
```

```
<gemarkungsnummer>1054</gemarkungsnummer>
```

```
</AX_Gemarkung_Schlüssel>
```

```
</gemarkung>
```

```
- <zustaendigeStelle>
```

```
- <AX_Dienststelle_Schlüssel>
```

```
<zustaendigeStelle_land>13</zustaendigeStelle_land>
```

```
<stelle>4087</stelle>
```

```
</AX_Dienststelle_Schlüssel>
```

```
</zustaendigeStelle>
```

- Umbenennung nur an konkreter Stelle unter AX_Flurstueck



NAS-4 Erstellung des Datenmodells für den ogr Import (classes2ogr)

- zu lange Attributnamen oder Tabellennamen auch zur Umbenennungsliste hinzufügen

```
-- ax_entstehungsartoderklimastufewasserverhaeltnisse_bodenschaetzung  
  
CREATE TABLE ax_entstehungsartoderklimastufewasserverhaeltnisse_bodensc  
(  
  wert CHARACTER VARYING NOT NULL,  
  beschreibung CHARACTER VARYING,  
  ax_entstehungsartoderklimastufewasserverhaeltnisse_bodensc CHARACTER VARYING(255),  
)
```

<ax_entstehungsartoderklimastufewasserverhaeltnisse_bodenschaetzung>

=>

<ax_entstehungsartoderklimastufewasserverhaeltnisse_bodensc>

NAS-4 Erstellung des Datenmodells für den ogr Import (classes2ogr)



- `<zeigtAufExternes>` wird von ogr2ogr ignoriert
- Wird als `text[]` ins Modell eingebaut.



NAS-4 Erstellung des Datenmodells für den ogr Import (classes2ogr)

- OGR-Schema-Erstellung konfigurierbar:
 - => Tabellen und Attribute können ausgeschlossen werden
 - => ein Standardprofil
 - => bundeslandspezifische Profile



NAS-4 Erstellung des Datenmodells für den ogr Import (classes2ogr)

Das Ergebnis ist

1. das Skript classes2ogr,
2. eine Umbenennungsliste für Attribute und Tabellen,
3. eine Konfigurationsdatei zum Ausschließen von Klassen und Attributen
4. Das Datenbankschema „alkis“ zum Einlesen von NAS-Dateien mit ogr2ogr



NAS-5 Erstellung eines Umbenennungsskriptes (nas2ogrgml)

- die Umbenennungen auf Datenbankebene aus Schritt NAS-4 nun auch auf die NAS-Dateien anwenden
- danach sind Datenmodell und NAS-Dateien kompatibel
- Ergebnis
=> Das Umbenennungsskript nas2ogrgml



Ergebnis

- Nach der Erweiterung erfolgt das Einlesen von NAS mit PostNAS wie folgt:
- Erweitertes Schema anlegen (einmalig)
- Ggf. Prof Idatei anpassen (i.d.R. einmalig)
- NAS-Datei mit Umbenennungsskript bearbeiten
- ogr2ogr ausführen

- Die letzten beiden Schritte möglichst zusammen in einem Shell-Skript, bei jedem Einlesevorgang erforderlich

NAS-6 Dokumentation und Koordinierung mit dem PostNAS Projekt



- Dokumentation
 - Beschreibung der ausgeführten Schritte zur Erweiterung des OGR Modells
- Darstellung der Unterschiede zwischen dem derzeit vorhandenen Datenmodell der PostNAS Suite und dem erstellten neuen vollständigen Modell.
- Bereitstellung der Dokumente im PostNAS Projekt.
- Source-Code (PHP) in öffentlichem Git-Repository
- Datenbankmodelle werden für PostgreSQL ≥ 9.1 und PostGIS > 2.1 erstellt und als SQL-Dumps ebenfalls im Git-Repository veröffentlicht.
- Sämtliche Skripte und Schemata werden unter die GNU General Public License zur freien Nachnutzung zur Verfügung gestellt.



Hilfsmittel und Dokumentationen

- Hilfsmittel und Dokus

<http://gdi-service.de/public/schemas/aaa/>

Software

- <https://github.com/pkorduan/xmi2db/>



Index of /public/schemas/aaa

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
Parent Directory		-	
2015-11-5 AAA Modell EA.xmi	21-Jan-2016 18:10	18M	
AFIS-ALKIS-ATKIS Anwendungsschema.pdf	22-Jun-2016 16:57	6.8M	
Hauptdokument GeoInfoDok V6_0_1.pdf	22-Jun-2016 16:54	1.3M	
Images/	21-Jan-2016 17:16	-	
diagrams.php	22-Jun-2016 17:03	1.5K	
ea_catalog_implementierungsmodell/	29-Jun-2016 11:38	-	
ea_catalog_uml_model/	29-Jun-2016 11:37	-	

Apache/2.2.16 (Debian) Server at gdi-service.de Port 80



pkorduan / xmi2db

<> Code Issues 9 Pull requests 0 Pulse Graphs

Convert UML class models from XMI files into a database.

17 commits 10 branches

Branch: master New pull request

pkorduan Merge branch 'dev-3' into develop

- XPlan2iQvoc Create init files and improve README.md
- conf Add association tables. Filter packages.
- lib Create init files and improve README.md
- sql Create init files and improve README.md
- .gitignore Add .gitignore.
- README.md Make minor format changing. Add uuid generation for gml_id.



Fragen

- Dürfen Attribute in PostNAS Modell umbenannt werden?
- Wo stellen wir das Schema zur Verfügung?
- Wo stellen wir die Profile zur Verfügung?
- Wo stellen wir das Umbenennungsskript zur Verfügung?



Kontakt

peter.korduan@gdi-service.de