

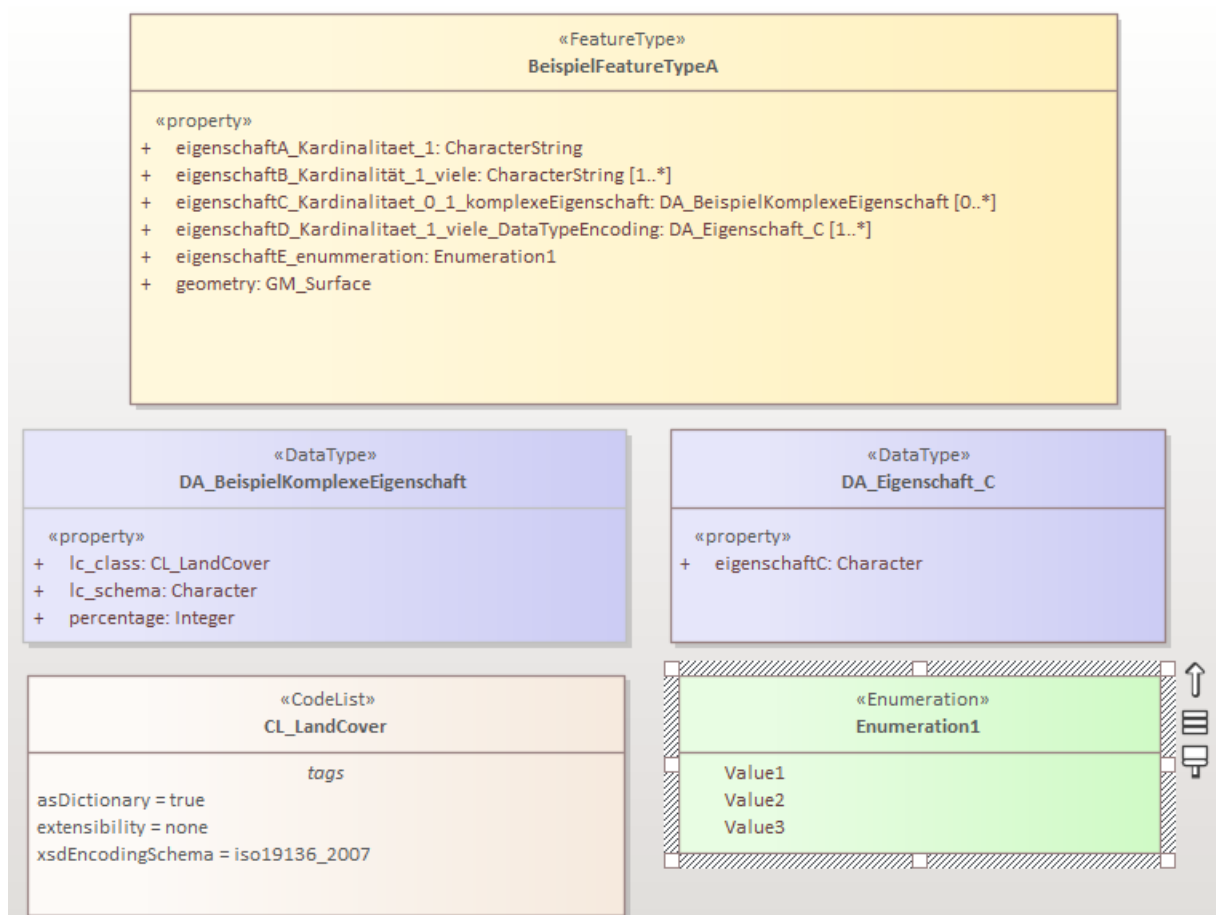
Workflow Model Driven Approach Geodata Modelling with UML/GML

Erstellen des GML-Applikationsschemas

Teil 1

Wesentliche Elemente/Konstrukte von GML

DI ROLAND GRILLMAYER



Erstellen des GML-Applikationsschemas

Um die Datentypen der Standards der ISO 19100 Serie nutzen zu können wir das Enterprise Architect XMI-File mit allen INSPIRE Datenmodellen sowie den ISO TC211 Standards importiert

IMPORT XMI

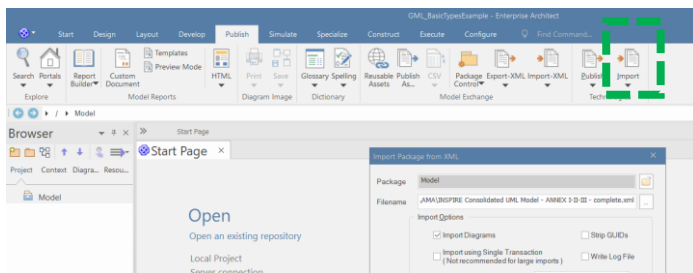
Download XMI (Enterprise Architect) der INSPIRE Datenmodelle

<https://inspire.ec.europa.eu/portfolio/data-models>

XMI Datei verfügbar unter [/Ressourcen/INSPIRE Consolidated UML Model - ANNEX I-II-III - complete](#)

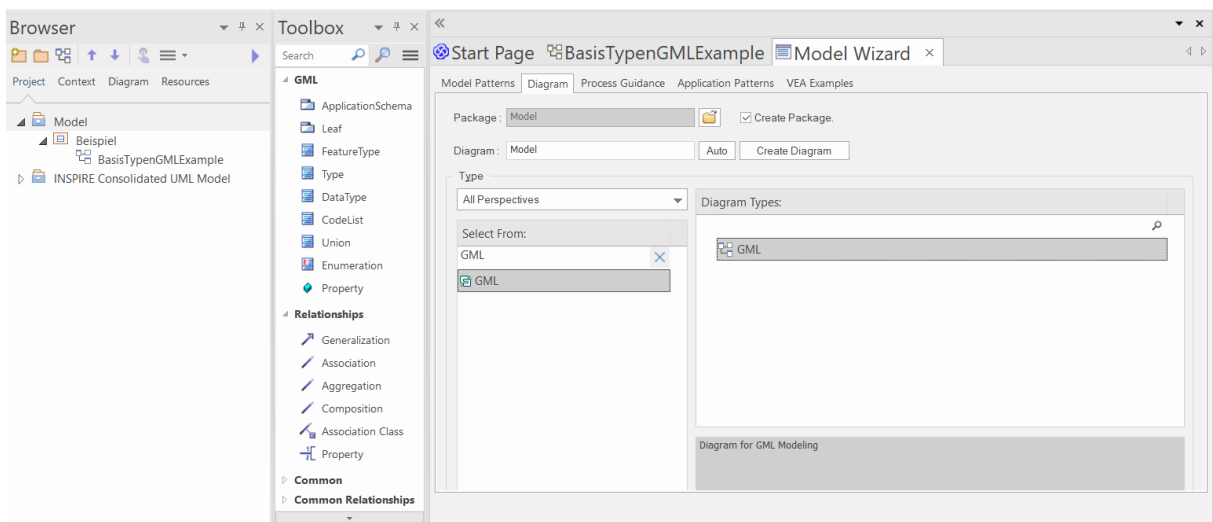
Import der XMI Datei im EA

Publish --> Import XML



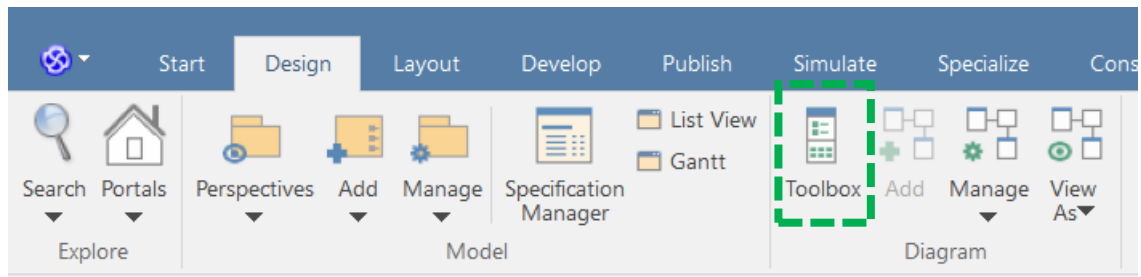
ERSTELLEN EINES GML PROJEKTES MIT DEM MODEL WIZARD

Model Root → Add a new Model using Wizard → Diagram → Search “GML” → Create Diagram (Enable Create Package)



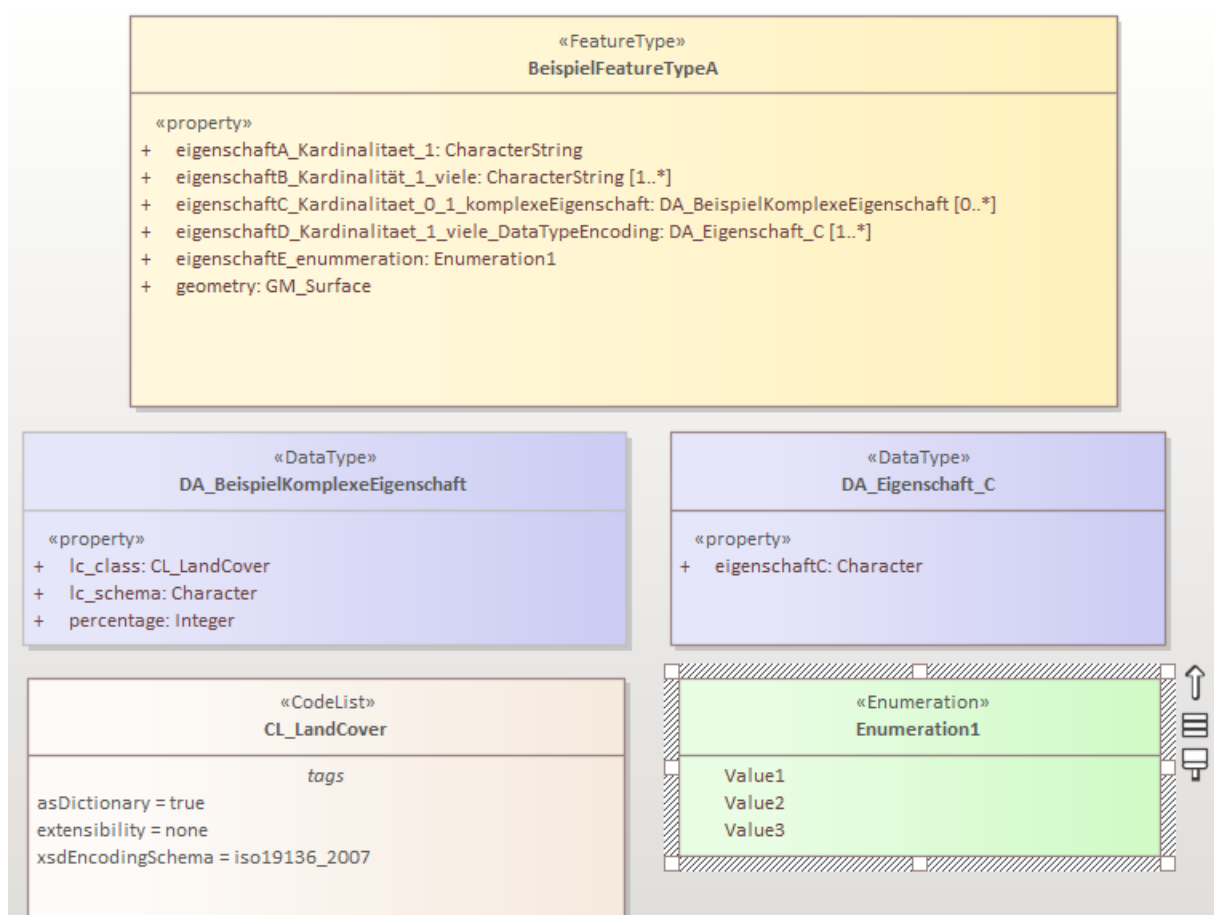
ERSTELLEN DES DATENMODELLS

Toolbox einblenden



BEGINN MODELLIERUNG

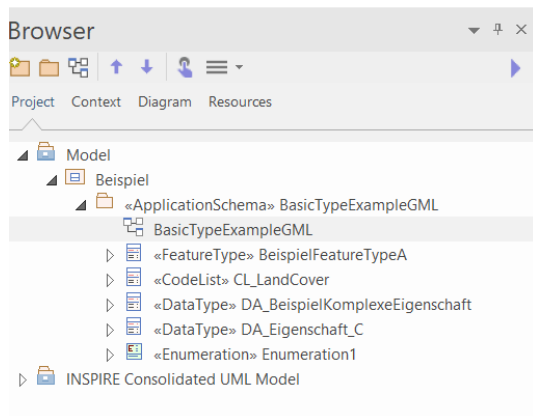
- Erstellen eines Feature Types
- Hinzufügen von Feature Properties
- Erstellen eines komplexen Feature Types
- Erstellen einer Codeliste
- Erstellen einer Enumeration



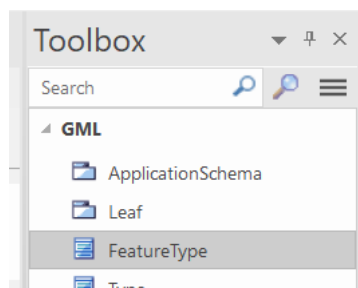
Fertiges UML Diagramm mit den wichtigsten Elementen die GML zur Verfügung stellt. Assoziationen / Kompositionen und Aggregationen werden in einem gesonderten Kapitel abgehandelt

ERSTELLEN EINES FEATURE TYPES

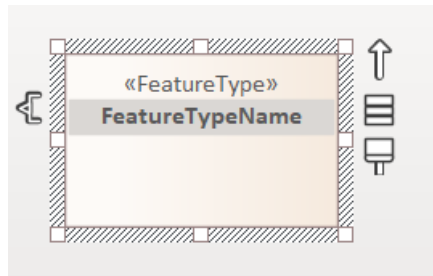
(FT.1) Öffnen des Diagramms „BasicTypeExampleGML“



(FT.2) Toolbox → FeatureType selektieren und auf Diagramm „BasicTypeExampleGML“ ziehen

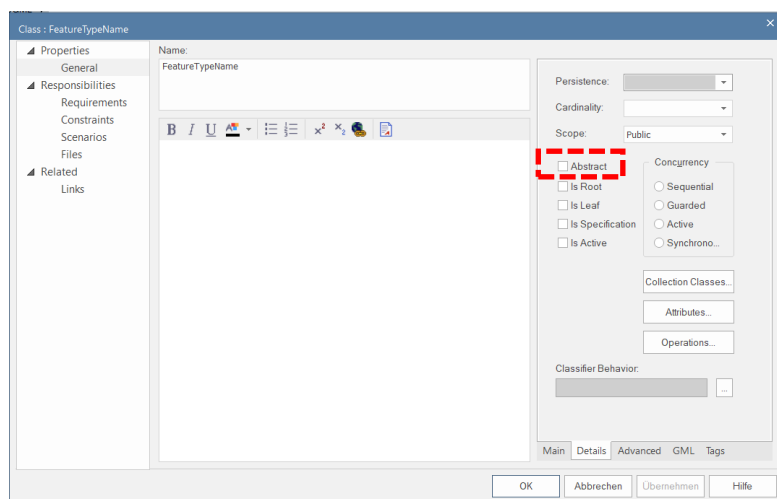


(FT.3) Benennung des FT



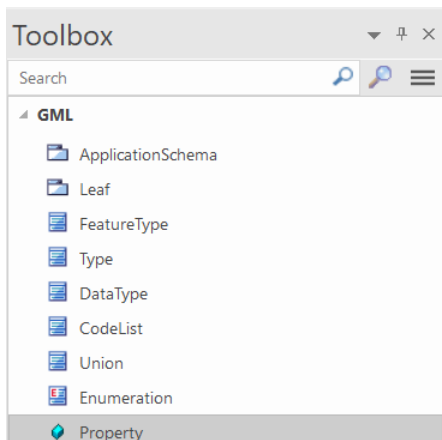
(FT.4) Konfiguration der Eigenschaften des FT → Doppelklick auf den FT

Falle es sich um einen Abstrakten FT handelt, muss dieser Type im Detail-Reiter konfiguriert werden

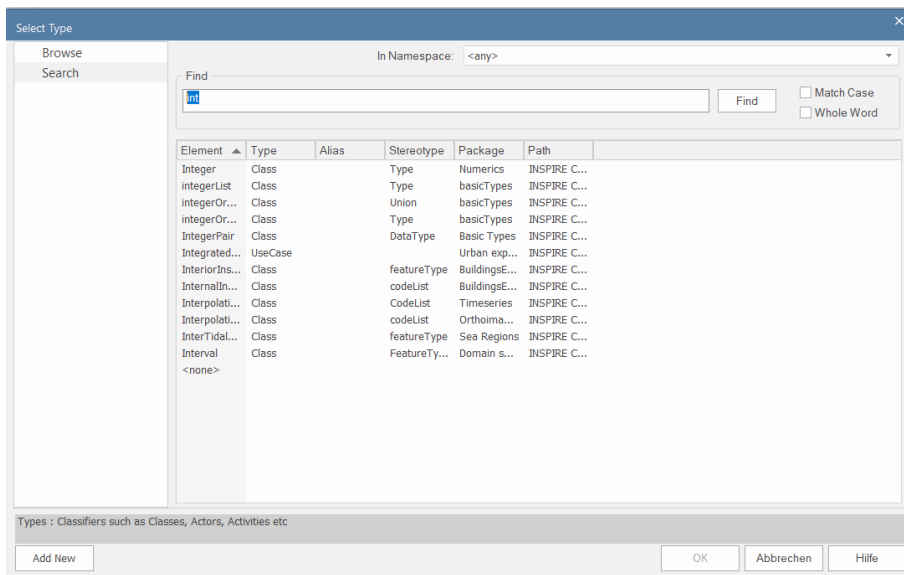
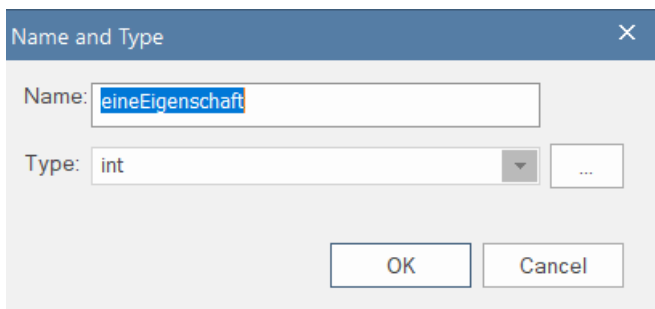


HINZUFÜGEN VON FEATURE PROPERTIES

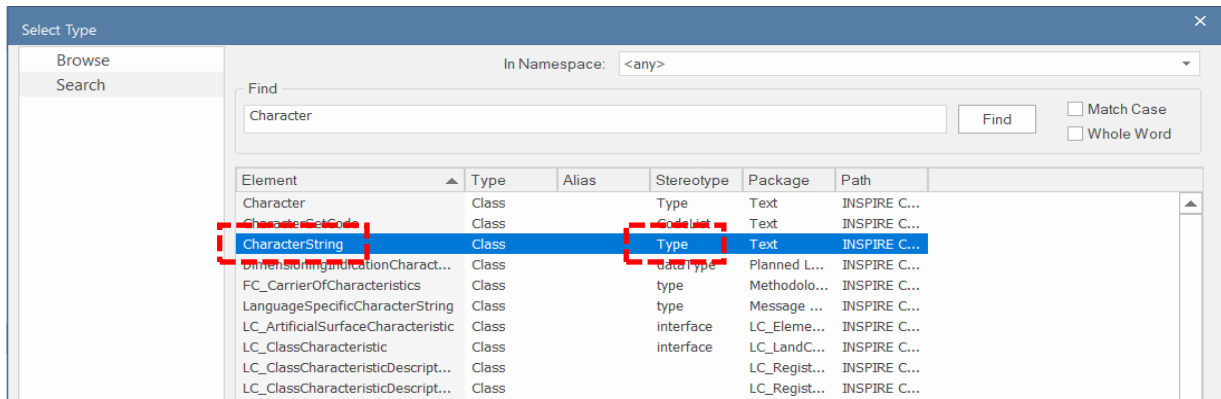
(FP.1) Toolbox → Property selektieren und auf den FT ziehen dem die Eigenschaft zugeordnet werden soll.



(FP.2) Name und (FP.3) Datentyp der Feature Property konfigurieren



Die Zuordnung eines Datentyps erfolgt mit Hilfe der Suche (Search) oder der direkteren Zuordnung eines Datentyps (Browse). Wenn zum Beispiel der Datentyp „CharacterString“ deklariert werden soll, empfiehlt es sich, diesen über die Suchfunktion zuzuordnen. Wichtig ist das man jenes Element wählt, das mit dem Stereotypen „Type“ ausgezeichnet ist.

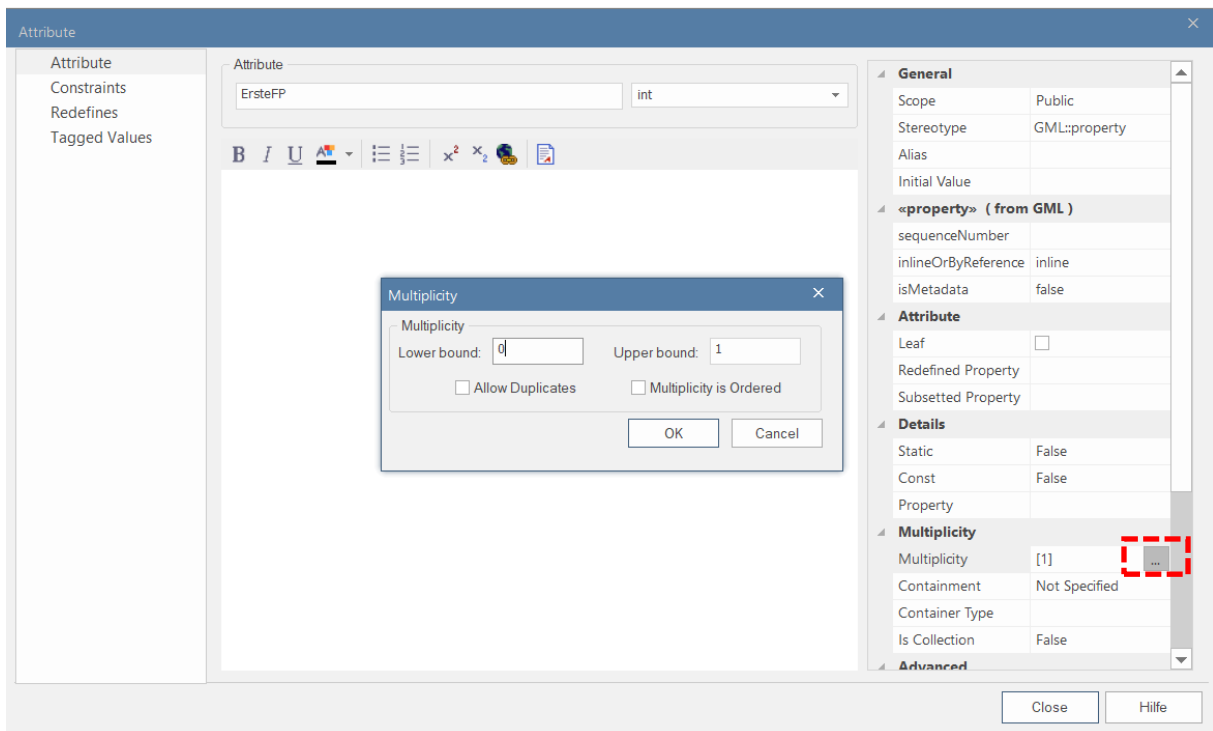


(FP.4) Kardinalität der Feature Property konfigurieren

Doppelklick auf die erstellte Feature Property. Im Fenster Multiplicity kann anschließend die Kardinalität der Eigenschaft definiert werden.

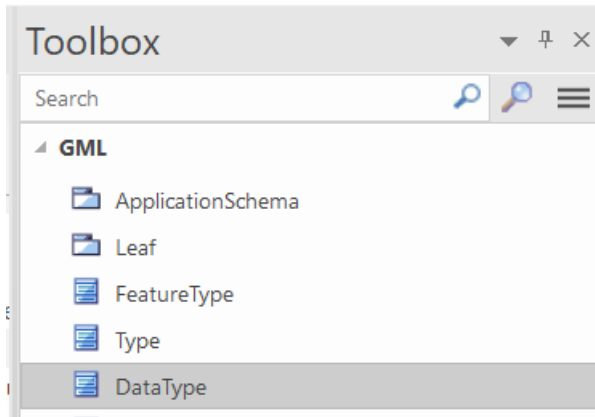
- 0...1 (kann maximal einmal Vorkommen – muss aber nicht angegeben werden)
- 1...1 (FP muss angegeben werden)
- 0...* (FP kann mehrmals angegeben werden)
- 1...* (FP muss einmal, kann mehrmals angegeben werden)

Die Kardinalität werden links neben der FP angezeigt. Keine Anzeige erfolgt bei der Kardinalität 1



ERSTELLEN EINES KOMPLEXEN FEATURE TYPES

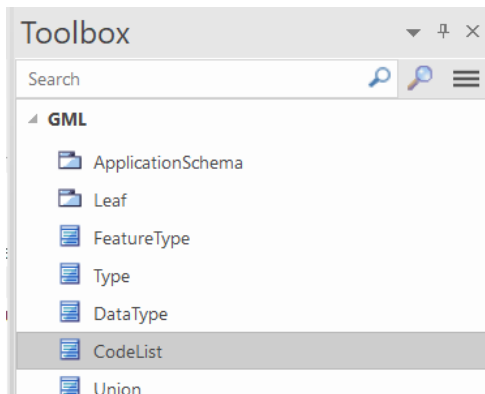
(DT.1) Toolbox → DataType selektieren und auf Diagramm „BasicTypeExampleGML“ ziehen



Komplexe Datentype setzen sich aus mehreren Feature Properties zusammen. Diese werden auf gleiche Art und Weise wie eine einzelnen Feature Property erstellt (siehe FP.1-FP.4)

ERSTELLEN EINER CODELISTE

(CL.1) Toolbox → Codeliste selektieren und auf Diagramm „BasicTypeExampleGML“ ziehen und Codeliste benennen.



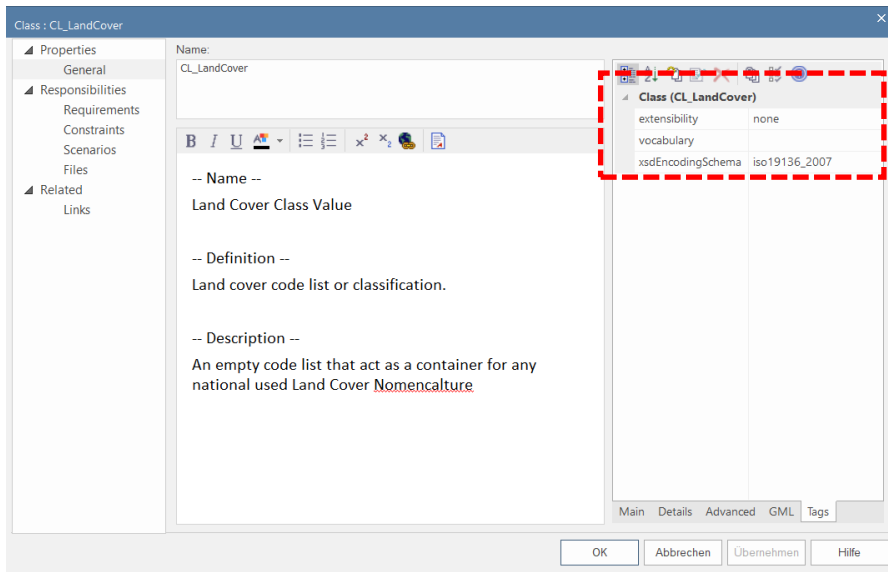
(CL.2) Konfigurieren der Eigenschaften der Codeliste. Codelisten sollten normalerweise in einem Codelistenregister geführt und gewartet werden. Falls die Codeliste in einem Codelistenregister geführt wird sind die folgenden Tags zu konfigurieren.

extensibility none

(nicht erweiterbare Codeliste) – Definition ob Codeliste erweiterbar ist oder nicht

vocabulary

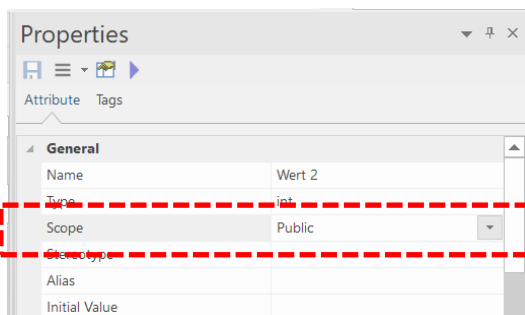
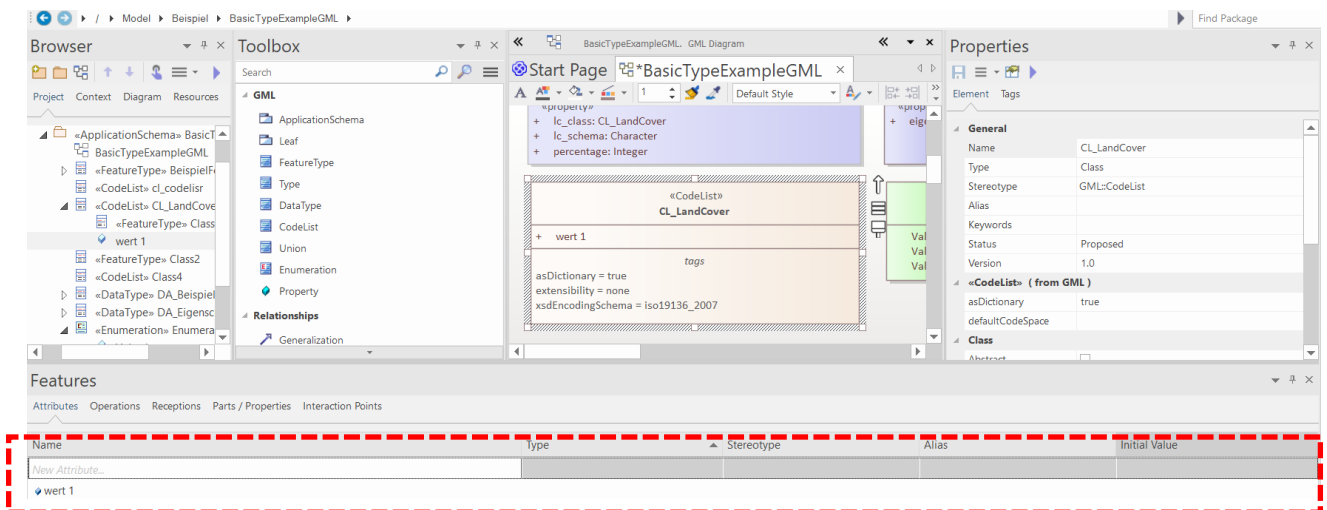
xsdEncodingSchema iso19136_2007



(CL.3 | CL.4) Falls die Codelistenwerte direkt im Schema und nicht in einem Codelisten Register verfügbar gemacht werden – können wie folgt die Codelistenwerte erstellt werden.

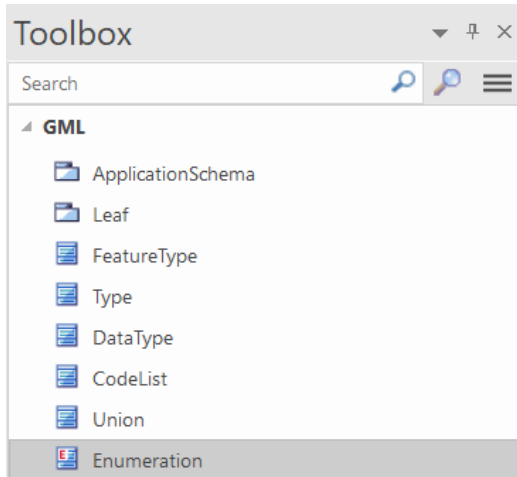
ACHTUNG: Wartbarkeit der Codeliste stark eingeschränkt da diese direkt im Schema eingebettet ist = für einen neuer Codelisten Wert muss eine neue Version des GML Applikationsschemas erstellt werden

Selektieren der Codeliste und eintragen der Codelistenwert unter „Features“. Für den Codelistenwert unter Properties den Scope auf „public“ setzen



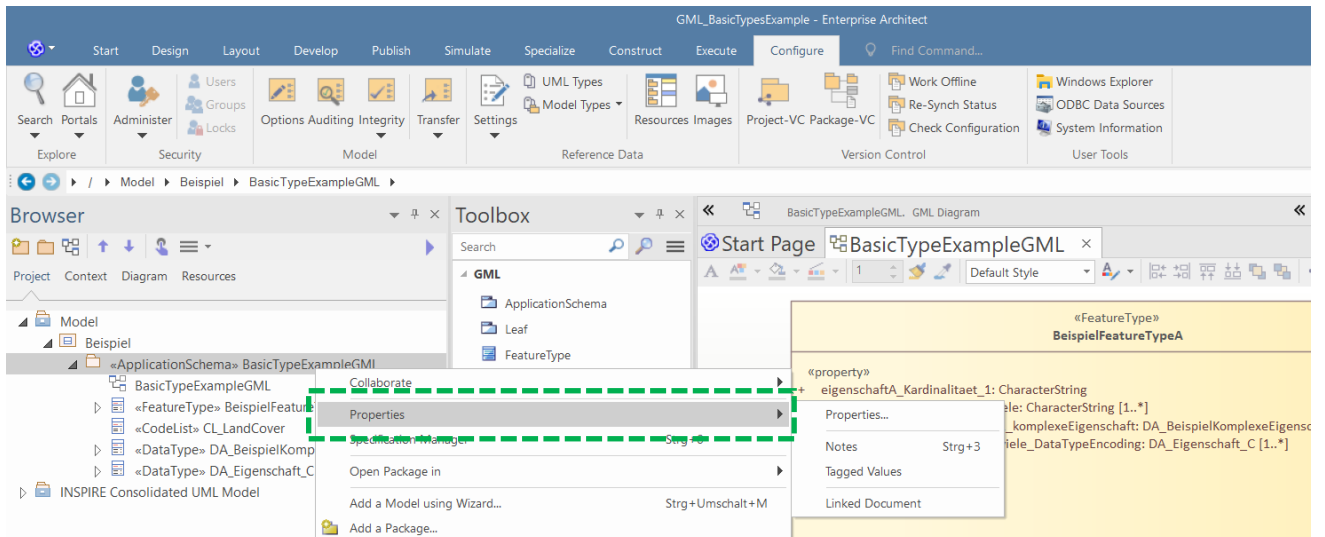
ERSTELLEN EINER ENUMERATION

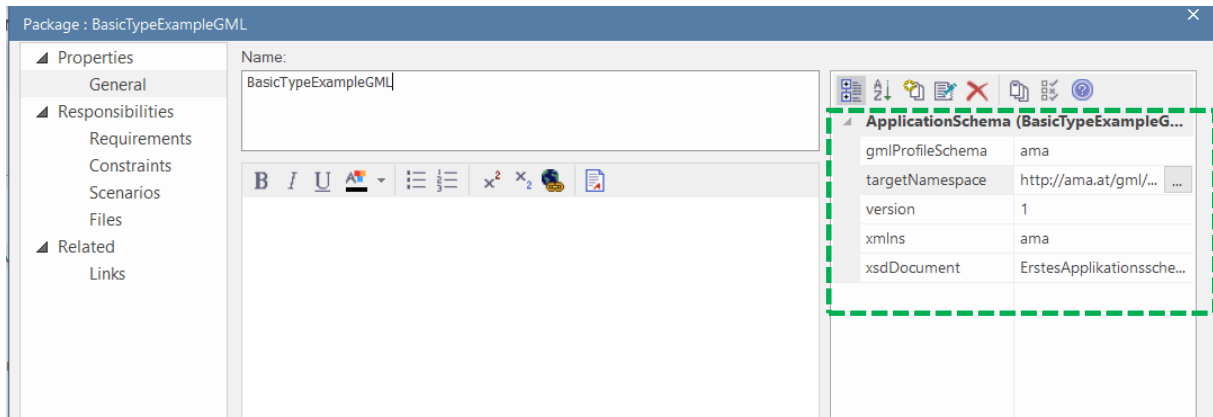
(ENUM.1) Toolbox →Enumeration selektieren und auf Diagramm „BasicTypeExampleGML“ ziehen. Enumerations sind nicht erweiterbare Codelisten. Die Werte der Enumeration werden auf die gleiche Art und Weise wie für Codelisten die direkt in das Schema eingebettet werden definiert (CL.3 | CL.4)



ABLEITEN DES GML-APPLIKATIONSSCHEMAS

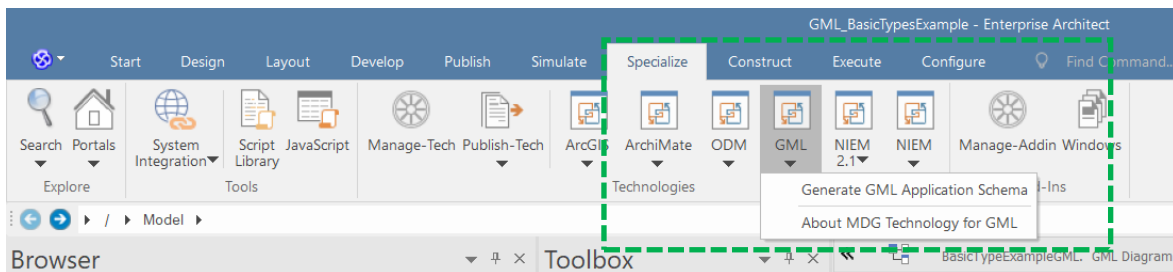
Rechter Mouse Klick Application Schema --> Properties → Konfigurieren der Properties für die Erstellung des Applikationsschemas





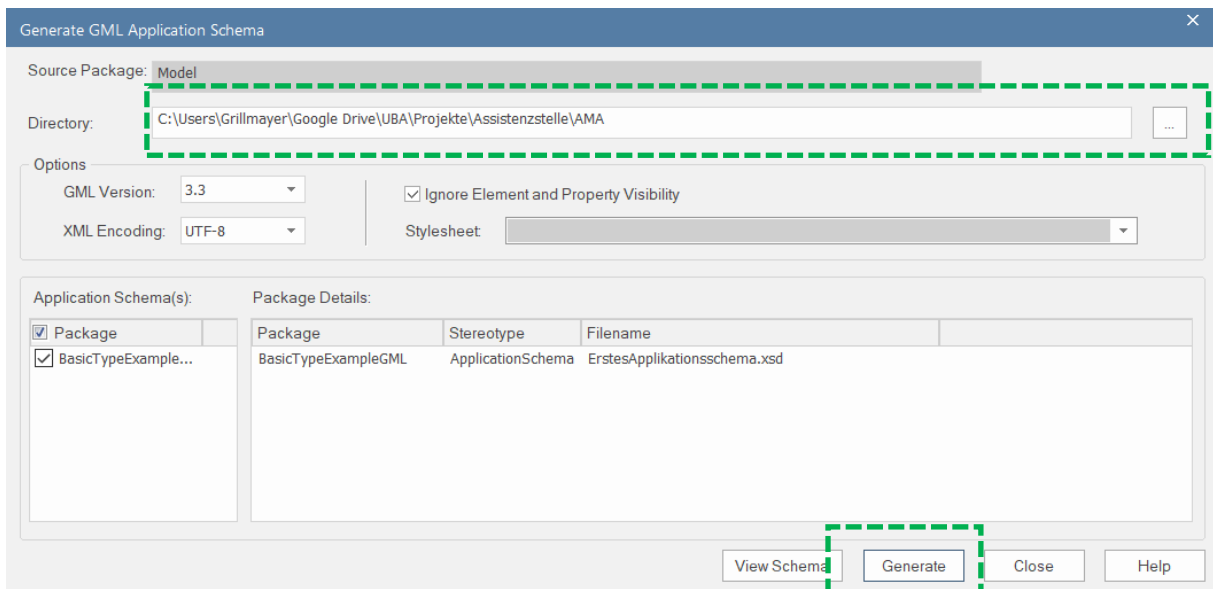
Eintragen des gmlProfileSchmema, targetNamespace, version des Applikationsscemas, Namespace Prefix (xmlns) und des Dateinamens des Applikationsschemas

Ableiten des Applikationsschema



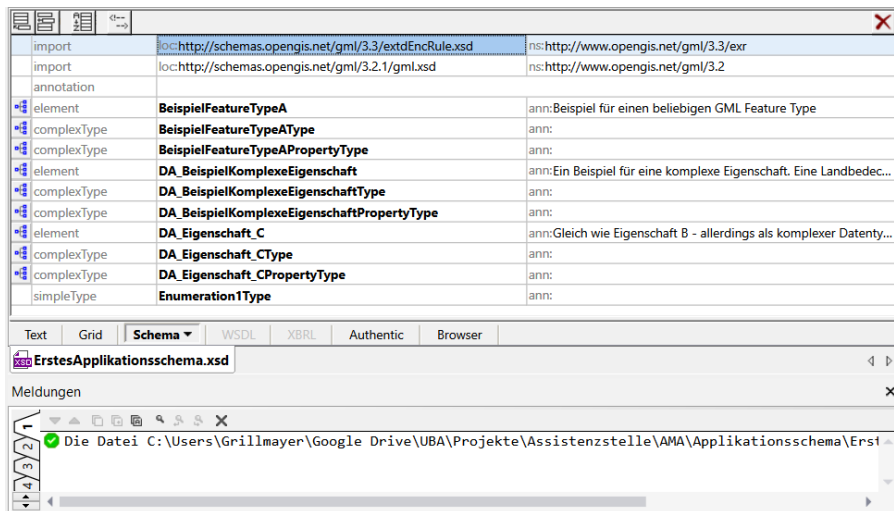
Angabe des Verzeichnisses in dem das Applikationsschema gespeichert wird

Mit „Generate“ wird das Applikationsschema erstellt



VALIDIERUNG DES GML-APPLIKATIONSSCHEMAS

Validierung des GML Applikationsschemas in einem xml-Editor. Allfällige Behebung von Fehlern. Das vorliegende Projekt weist keine Fehler auf.



GML Applikationsschema

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Generated by Enterprise Architect 15.2.1555 ( Build: 1555 )-->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:ama="http://ama.at/gml/erstesApplikationsschema"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:gmlx="http://www.opengis.net/gml/3.3/exr"
targetNamespace="http://ama.at/gml/erstesApplikationsschema" elementFormDefault="qualified" version="1">
  <xs:import namespace="http://www.opengis.net/gml/3.3/exr"
  schemaLocation="http://schemas.opengis.net/gml/3.3/extdEncRule.xsd"/>
  <xs:import namespace="http://www.opengis.net/gml/3.2"
  schemaLocation="http://schemas.opengis.net/gml/3.2.1/gml.xsd"/>
  <xs:annotation>
    <xs:appinfo>
      <gml:gmlProfileSchema>ama</gml:gmlProfileSchema>
    </xs:appinfo>
  </xs:annotation>
  <xs:element name="BeispielFeatureTypeA" type="ama:BeispielFeatureTypeAPropertyType"
  substitutionGroup="gml:AbstractFeature">
    <xs:annotation>
      <xs:documentation>Beispiel für einen beliebigen GML Feature Type</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:complexType name="BeispielFeatureTypeAPropertyType">
    <xs:complexContent>
      <xs:extension base="gml:AbstractFeatureType">
        <xs:sequence>
          <xs:element name="eigenschaftA_Kardinalitaet_1" type="xs:string"/>
          <xs:element name="eigenschaftB_Kardinalitaet_1_viele" type="xs:string"
          minOccurs="1" maxOccurs="unbounded"/>
          <xs:element name="eigenschaftC_Kardinalitaet_0_1_komplexeEigenschaft"
          minOccurs="0" maxOccurs="unbounded">
            <xs:complexType>
              <xs:complexContent>
                <xs:extension base="gml:AbstractMemberType">
                  <xs:sequence>
                    <xs:element
                    ref="ama:DA_BeispielKomplexeEigenschaft"/>
                  </xs:sequence>
                </xs:extension>
              </xs:complexContent>
            </xs:complexType>
          </xs:element>
          <xs:element name="eigenschaftD_Kardinalitaet_1_viele_DataTypeEncoding"
          minOccurs="1" maxOccurs="unbounded">
            <xs:complexType>
              <xs:complexContent>
                <xs:extension base="gml:AbstractMemberType">
                  <xs:sequence>
                    <xs:element
                    ref="ama:DA_Eigenschaft_C"/>
                  </xs:sequence>
                </xs:extension>
              </xs:complexContent>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>
```

```

        </xs:element>
        <xs:element name="eigenschaftE_enumeration"
type="ama:Enumeration1Type"/>
        <xs:element name="geometry" type="gml:SurfacePropertyType"/>
    </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="BeispielFeatureTypeAPropertyType">
    <xs:sequence minOccurs="0">
        <xs:element ref="ama:BeispielFeatureTypeA"/>
    </xs:sequence>
    <xs:attributeGroup ref="gml:AssociationAttributeGroup"/>
    <xs:attributeGroup ref="gml:OwnershipAttributeGroup"/>
</xs:complexType>
<xs:element name="DA_BeispielKomplexeEigenschaft" type="ama:DA_BeispielKomplexeEigenschaftType"
substitutionGroup="gml:AbstractObject">
    <xs:annotation>
        <xs:documentation>Ein Beispiel für eine komplexe Eigenschaft. Eine Landbedeckungsklasse
und deren prozentuelle Abdeckung werden mit dieser komplexen Eigenschaft abgebildet.</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:complexType name="DA_BeispielKomplexeEigenschaftType">
    <xs:sequence>
        <xs:element name="lc_class" type="gml:ReferenceType">
            <xs:annotation>
                <xs:appinfo>
                    <gml:targetCodeList>CL_LandCover</gml:targetCodeList>
                </xs:appinfo>
            </xs:annotation>
        </xs:element>
        <xs:element name="lc_schema" type="xs:string"/>
        <xs:element name="percentage" type="xs:integer"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="DA_BeispielKomplexeEigenschaftPropertyType">
    <xs:sequence>
        <xs:element ref="ama:DA_BeispielKomplexeEigenschaft"/>
    </xs:sequence>
    <xs:attributeGroup ref="gml:OwnershipAttributeGroup"/>
</xs:complexType>
<xs:element name="DA_Eigenschaft_C" type="ama:DA_Eigenschaft_CType" substitutionGroup="gml:AbstractObject">
    <xs:annotation>
        <xs:documentation>Gleich wie Eigenschaft B - allerdings als komplexer Datentyp codiert da
dann die Auflösung und Anzeige im QGIS gewährleistet ist</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:complexType name="DA_Eigenschaft_CType">
    <xs:sequence>
        <xs:element name="eigenschaftC" type="xs:string"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="DA_Eigenschaft_CPropertyType">
    <xs:sequence>
        <xs:element ref="ama:DA_Eigenschaft_C"/>
    </xs:sequence>
    <xs:attributeGroup ref="gml:OwnershipAttributeGroup"/>
</xs:complexType>
<xs:simpleType name="Enumeration1Type">
    <xs:restriction base="xs:string">
        <xs:enumeration value="Value1"/>
        <xs:enumeration value="Value2"/>
        <xs:enumeration value="Value3"/>
    </xs:restriction>
</xs:simpleType>
</xs:schema>

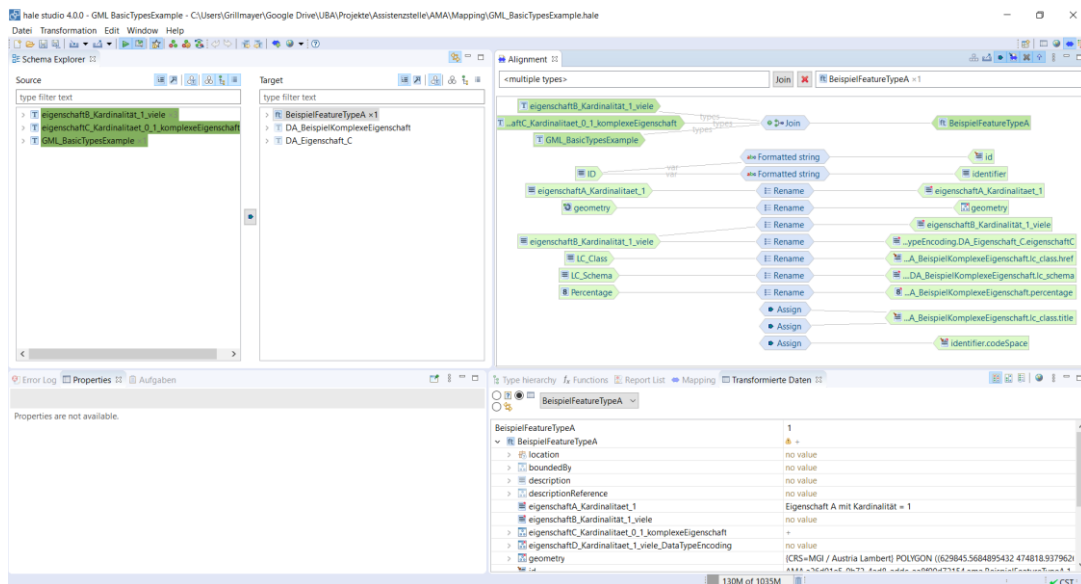
```

TESTEN DES APPLIKATIONSSCHEMAS – ERZEUGEN EINER TESTINSTANZ IN HALE STUDIO

Zum Testen des Applikationsschemas ist es sinnvoll dieses zu instanziiieren. Im Beispiel wurde für diese Zwecke ein [GeoPackage \(GML BasicTypesExample.gpkg\)](#) erzeugt, welches alle Informationen des im UML Diagramm dargestellten Feature Type beinhaltet. Dieses wird in weiterer Folge mit der Software HALE STUDIO auf dem erzeugten [Applikationsschema „Applikationsschema/ErstesApplikationsschema.xsd“](#) abgebildet („mapping“).

Das fertige HALE Studio Projekt befindet sich im Ordner [/Mapping](#).

[Open HALE Studio Projekt](#)



DATENHARMONISIERUNG – ERSTELLEN DES MAPPINGS ZWISCHEN AUSGANGSDATENSATZ UND DEM ZIELAPPLIKATIONSSCHEMA

Laden des Source schema... → [GeoPackage \(GML BasicTypesExample.gpkg\)](#)

Laden des Target schema... → [Applikationsschema „Applikationsschema/ErstesApplikationsschema.xsd“](#)

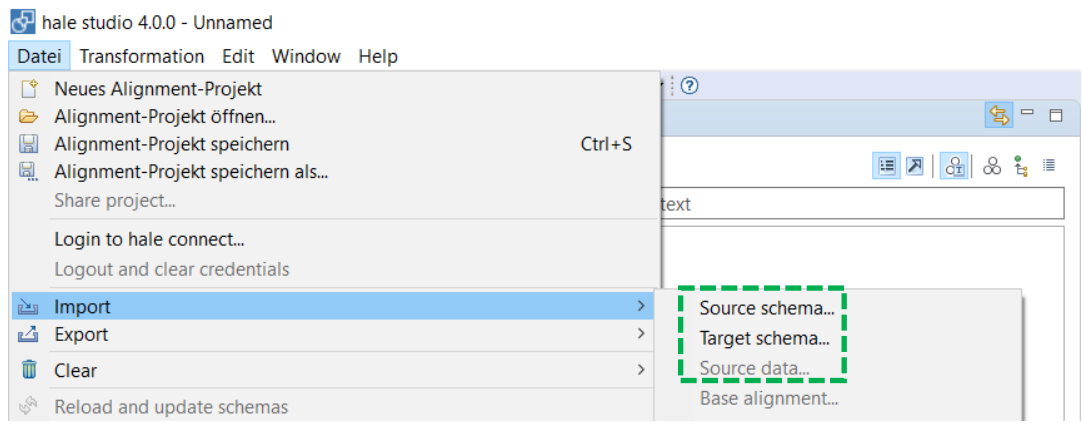
ANMERKUNG: Für den Datenimport im QGIS und anderen GIS Clients muss das Applikationsschema über das Web verfügbar sein. Dieses wird beim Import aufgerufen und beinhaltet alle Informationen die der Client benötigt um das Datenbankschema zu erstellen in dem dann die GML Instanz gespeichert wird = Schemabezogener Datentransfer

Für das Beispiel wurde das Schema unter den unten angeführten URL verfügbar gemacht (Ersichtlich im XML-Prolog der mit diesem Projekt erzeugten GML Instanz)

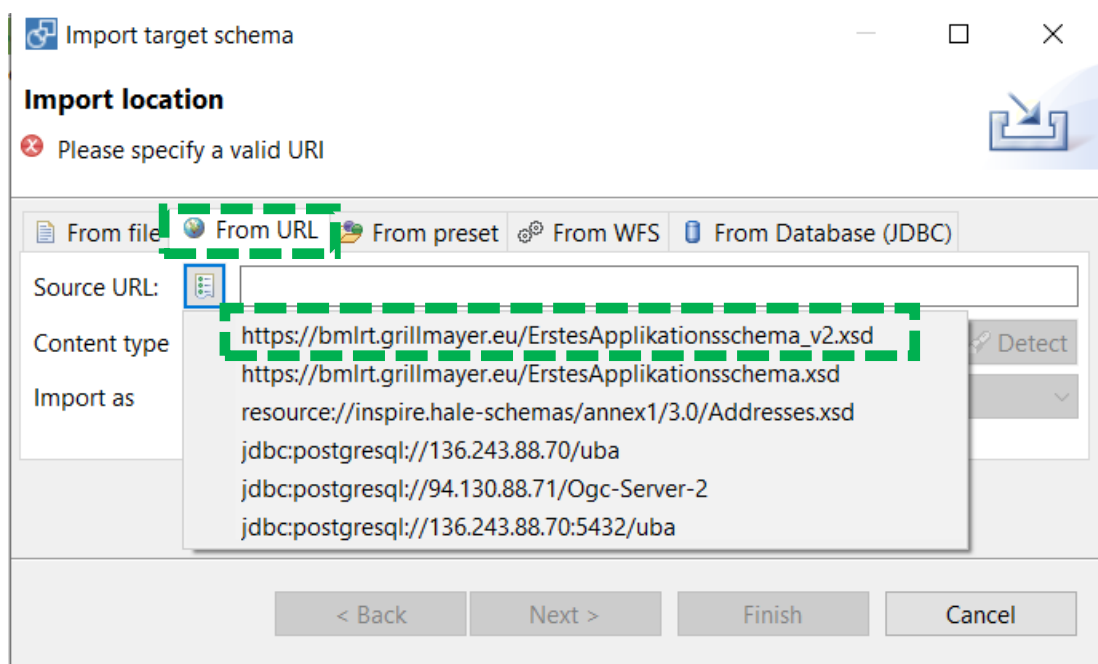
```

<?xml version="1.0" ?>
<gml:FeatureCollection xmlns:ama="http://ama.at/gml/erstesApplikationsschema"
xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:gml:exr="http://www.opengis.net/gml/3.3/exr"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:gmd="http://www.isotc211.org/2005/gmd"
xmlns:gco="http://www.isotc211.org/2005/gco"
xmlns:gss="http://www.isotc211.org/2005/gss"
xmlns:gstr="http://www.isotc211.org/2005/gstr"
xmlns:gts="http://www.isotc211.org/2005/gts"
xmlns:hfp="http://www.w3.org/2001/XMLSchema-hasFacetAndProperty"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:schemaCollection="http://ama.at/gml/erstesApplikationsschema"
https://bmlrt.grillmayer.eu/ErstesApplikationsschema_v2.xsd
http://www.opengis.net/gml/3.2 http://schemas.opengis.net/gml/3.2.1/deprecatedTypes.xsd">
  <gml:featureMember>
    <ama:BeispielFeatureTypeA gml:id="AMA.a26d01e5-0b72-4ad8-addc-aa8f90d72154.ama.BeispielFeatureTypeA.1">
      <gml:identifizier codeSpace="http://inspire.ec.europa.eu/ids">https://geodaten.ama.at/a26d01e5-0b72-4ad
      <ama:eigenschaftA_Kardinalitaet_1>Eigenschaft A mit Kardinalitaet = 1</ama:eigenschaftA_Kardinalitaet_

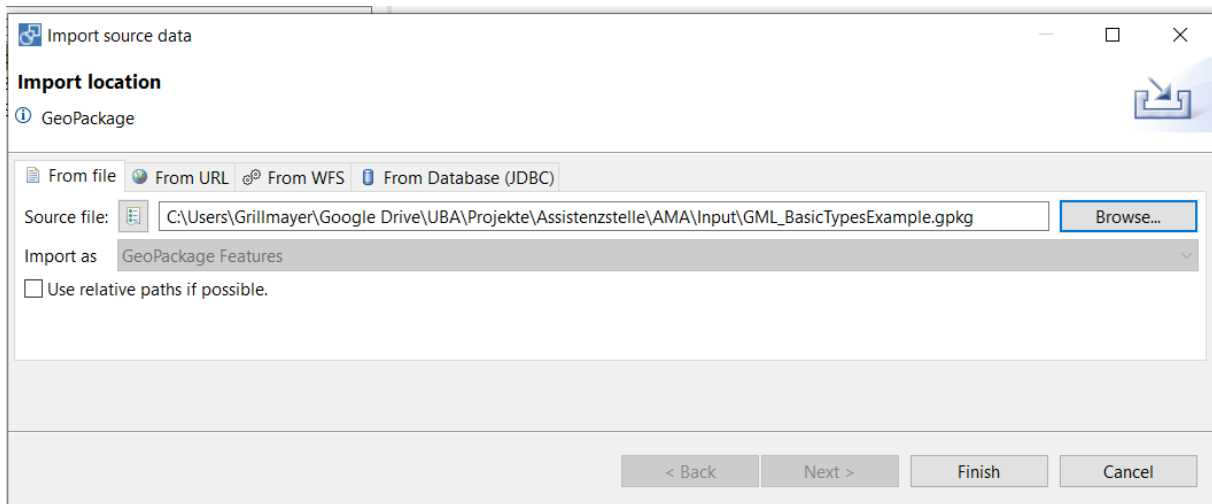
```



Laden der Source data... → [GeoPackage \(GML BasicTypesExample.gpkg\)](#)

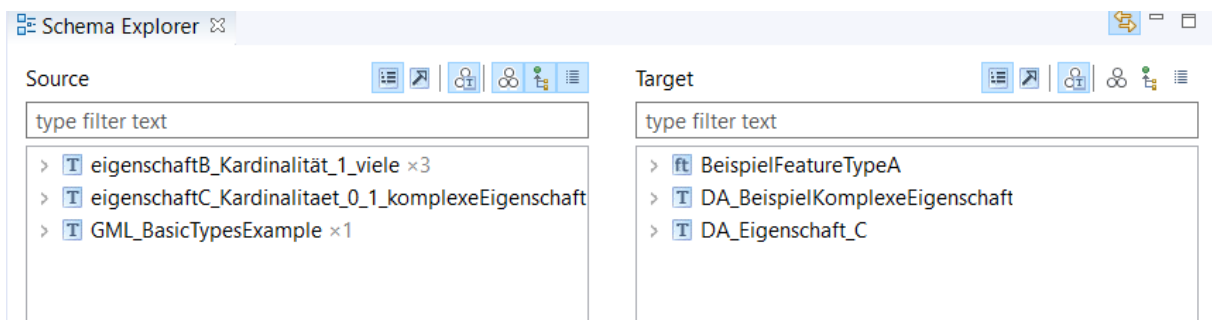


Laden des Target schema... → https://bmlrt.grillmayer.eu/ErstesApplikationsschema_v2.xsd



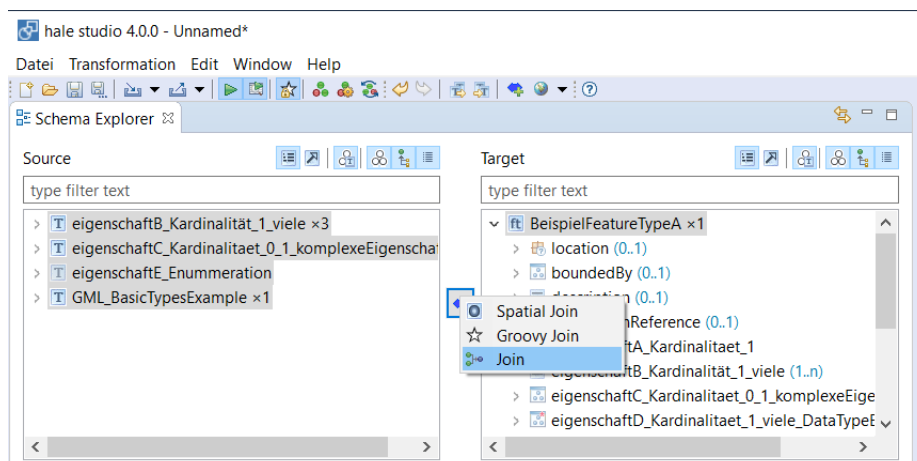
Laden der Source data..... → [GeoPackage \(GML_BasicTypesExample.gpkg\)](#)

Anschließend sollte im Schema Explorer folgende Inhalte verfügbar sein.

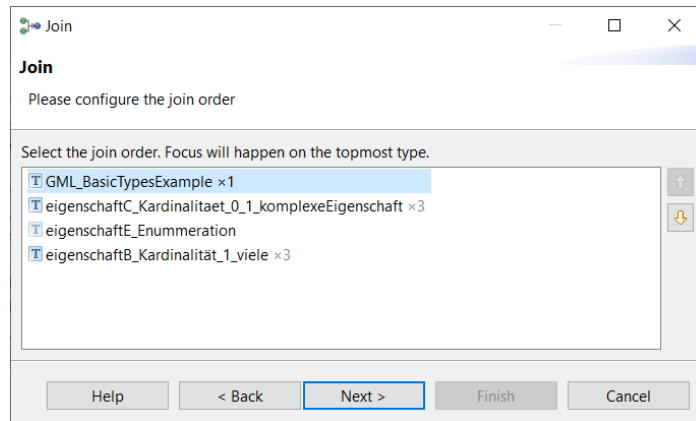


Mapping der Tabellen mit den Informationen auf den Feature Type. Falls mehrere Tabelle (1:n – Relationen) benötigt werden müssen die ID der Haupttabelle und Forekeys der Nebentabellen mittels der HALE Studio Funktion „Join“ konfiguriert werden

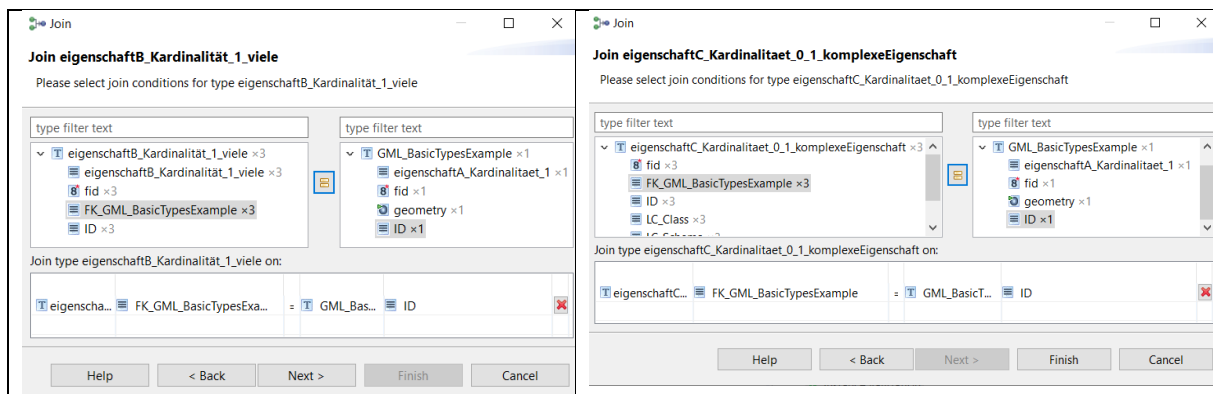
Markieren der Tabellen auf der linken Seite in den Ausgangsdaten (=Haupttabelle mit der Geometrie und Nebentabellen) und des Feature Types auf der rechten Seite des Zielmodells – anschließend die entsprechende Funktion auswählen. In Beispiel sind die Informationen für den Feature Type in den drei Ausgangstabellen der Source Daten beinhaltet. Deshalb werden mit der HALE Studio Funktion „Join“ diese auf den Feature Type gemappt und die ID der Haupttabelle mit den FK in den assoziierten Tabellen konfiguriert.



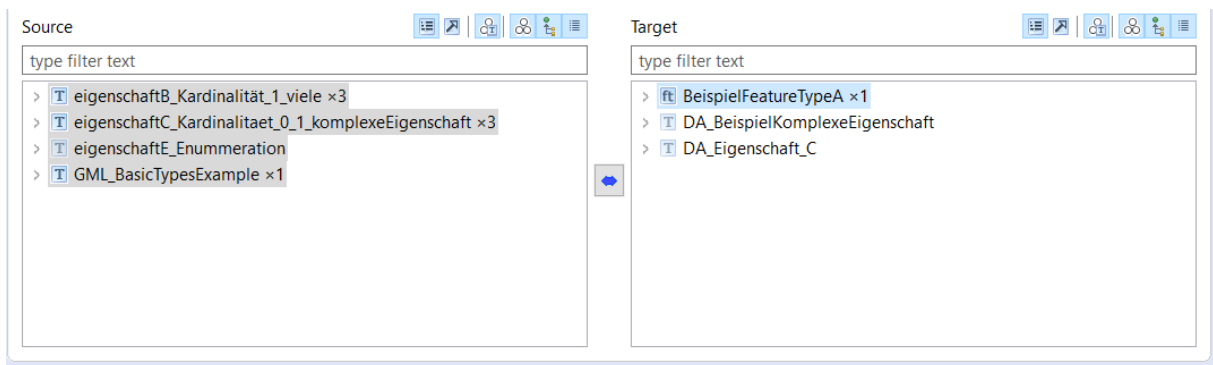
Mappen der 4 Tabellen auf der Source Daten auf den Feature Type in Zielmodell



Richtige Reihenfolge konfigurieren



ID Haupttabelle und FK der assoziierten Tabellen konfigurieren



Ergebnis des Mappings – es muss die selbe Instanzenanzahl der Haupttabelle im gemappten Feature Type vorhanden sein – falls nicht wurde die Join Reihenfolge nicht richtig konfiguriert.

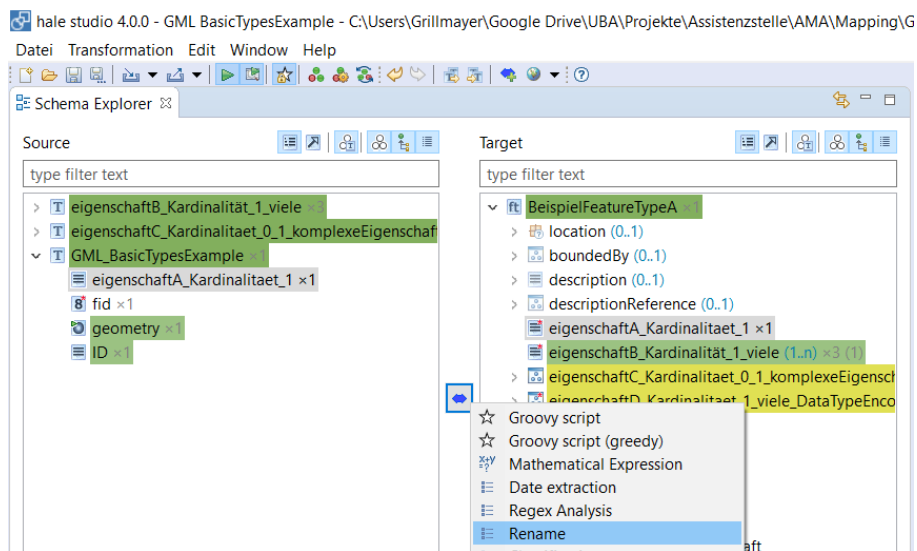
MAPPING DER FEATURE TYPE EIGENSCHAFTEN (FEATURE PROPERTIES)

Anschließend kann mit dem Mapping der Eigenschaften (=Feature Properties) begonnen werden.

Für dieses Mapping stehen unterschiedliche HALE STUDIO Funktionen zur Verfügung, die auch in Anhängigkeit zum Zieldatentype variieren. So kann man bei Geometrien zum Beispiel die Bounding Box oder den Centroid berechnen lassen, für ein Datum eine Datumstransformation durchführen lassen ect.

Rename: Die einfachste und am meiste angewendete Funktion ist das sogenannte „Rename“ – diese wird für ein 1:1 Mapping einer Eigenschaft aus dem Source Datensatz auf eine Feature Property des Zielmodells angewendet, was mehr oder weniger einer Umbenennung der Eigenschaft entspricht (=Rename)

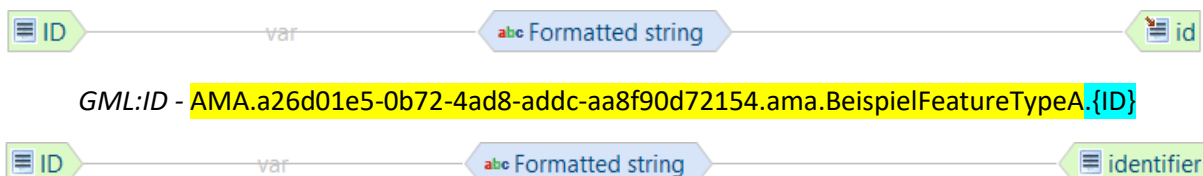
Auswahl der Eigenschaft des Ausgangsdatsatzes (eigenschaftA_Kardinalitaet_1) und der Feature Property des Zieldatenmodells (eigenschaftA_Kardinalitaet_1) und zuweisen der HALE Studio Funktion „Rename“



Im vorliegenden Projekt wurden des Weiteren die folgenden HALE STUDIO Funktionen verwendet

Assign: Zuweisen von fixen Werten

FormattedString: Zuweisen einer Stringkette in der Eigenschaft des Ausgangsdatsatz eingebettet werden. Beispiele sind die gml:id und der gml:identifier. Das in diesem Beispiel gewählte Muster für diese beiden Identifikatoren berücksichtigt den österreichischen Vorschlag der INSPIRE Assistenzstelle für die Erstellung von Identifikatoren..



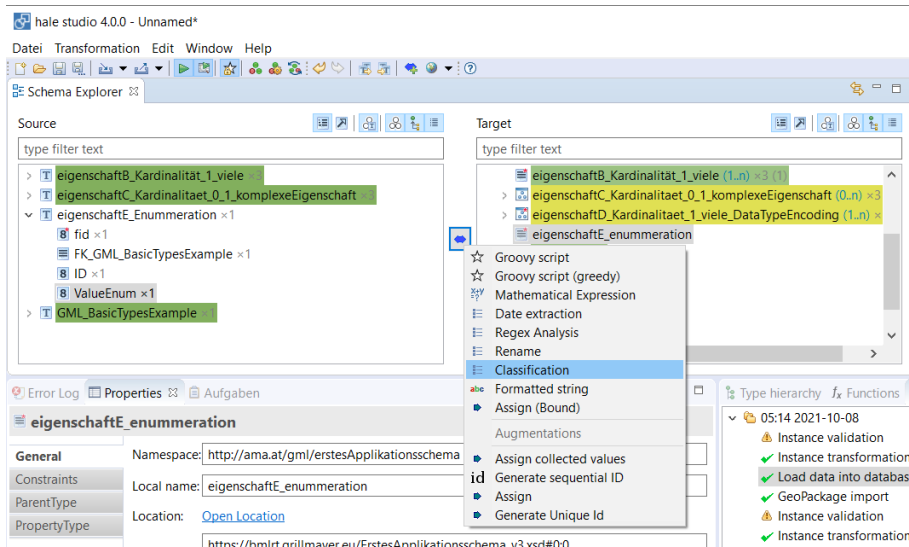
GML:ID - AMA.a26d01e5-0b72-4ad8-addc-aa8f90d72154.ama.BeispielFeatureTypeA.{ID}



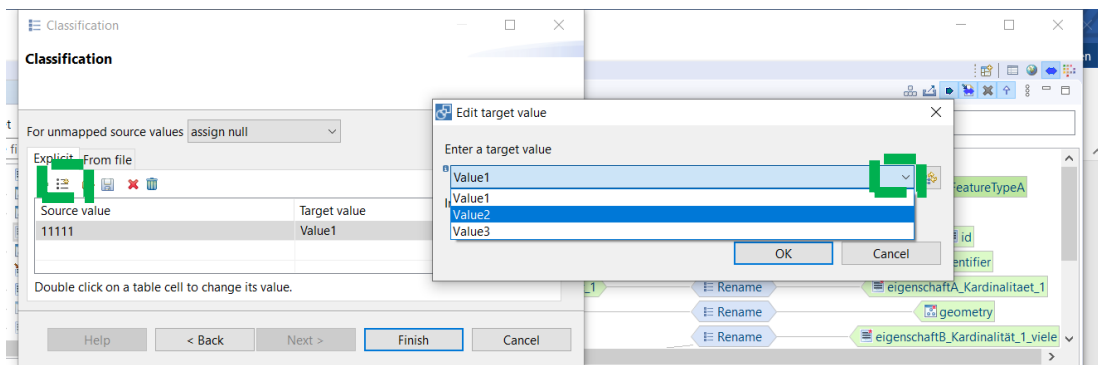
GMI:IDENTIFIER: <https://geodaten.ama.at/a26d01e5-0b72-4ad8-addc-aa8f90d72154/ama.BeispielFeatureTypeA/{ID}>

In der Funktion definierte Stringkette | Variable aus den Ausgangsdatsatz

Classification: Für die eigenschaftE_Enumeration wurde die Funktion „Classification“ verwendet. Mit dieser können Werte aus dem Ausgangsdatensatz einen anderen Wert im Zieldatensatz zugewiesen werden. Normalerweise werden die Werte des Ausgangsdatensatz beim GML Encoding dann auf Werte einer Codeliste welche in einer Codelistenregistry verwaltet werden zugeordnet. Im Beispiel wurde im Applikationsschema eine Enumeration definiert (=Codeliste mit fixen, nicht erweiterbaren Werten). Durch die Klassifikation wird der Wert des Ausgangsdatensatzes einem Wert der Enumeration zugewiesen



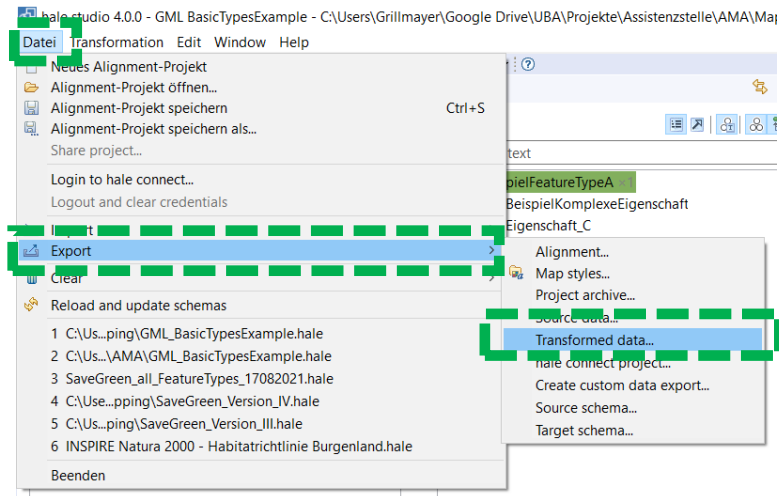
ValueEnum wird eigenschaftE_enumeration mittels der Funktion „Classification“ zugewiesen



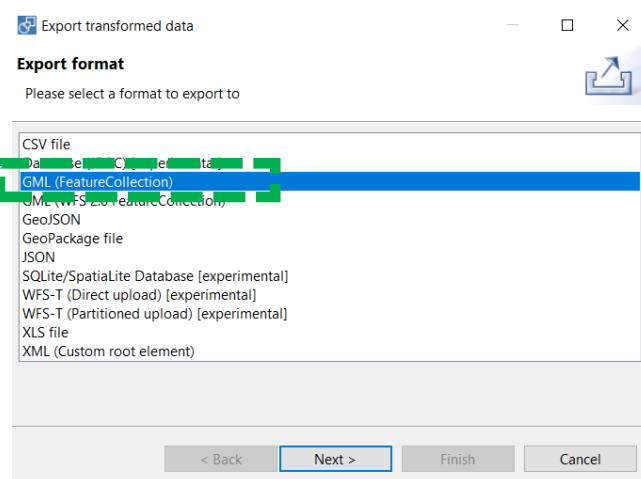
Laden der Werteausprägungen aus dem Ausgangsdatensatz und Zuweisung (=Klassifikation) entsprechend der in der Enumeration vorgesehenen Werteliste

Erstellen des GML-Instanz Dokuments

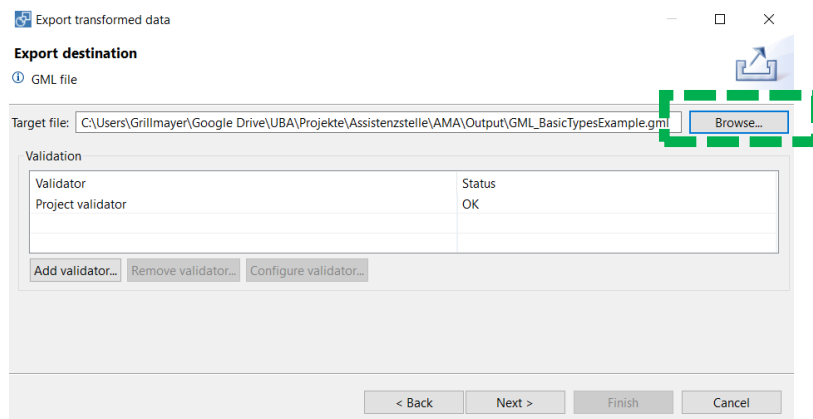
Abschließend kann eine GML Instanz erstellt werden. Für die Erstellung wurden die folgenden Einstellungen vorgenommen



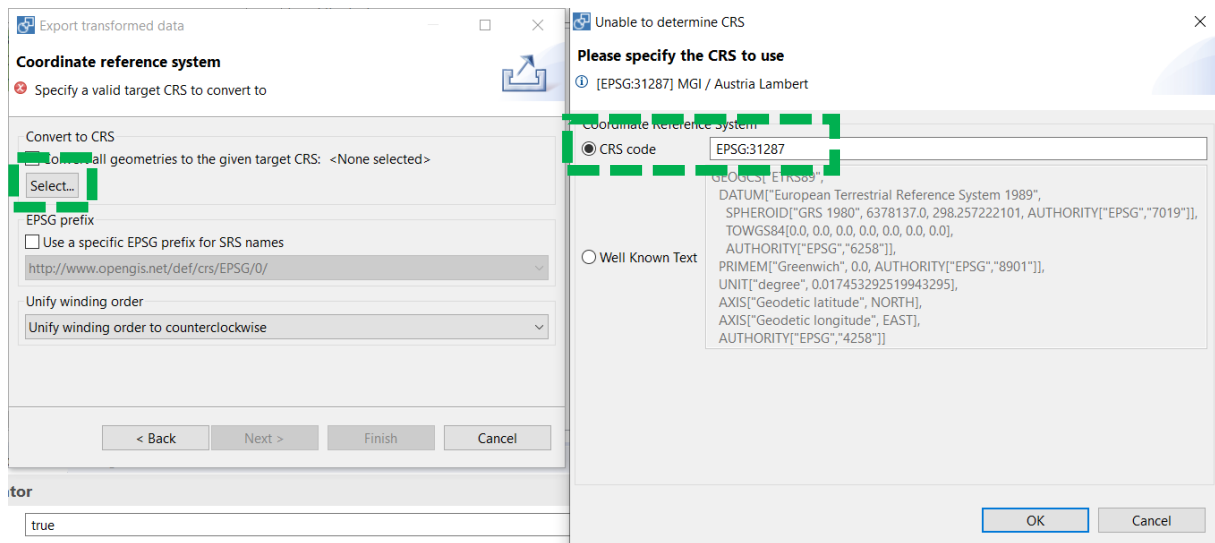
Transformieren der Daten



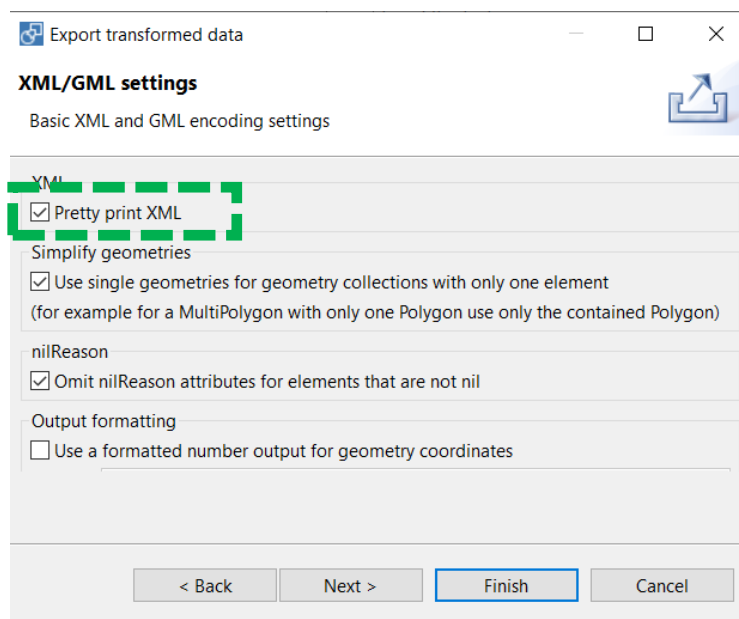
Daten werden als GML Feature Collection encodiert



Dateiname der GML Datei und Ordner in dem diese abgelegt wird



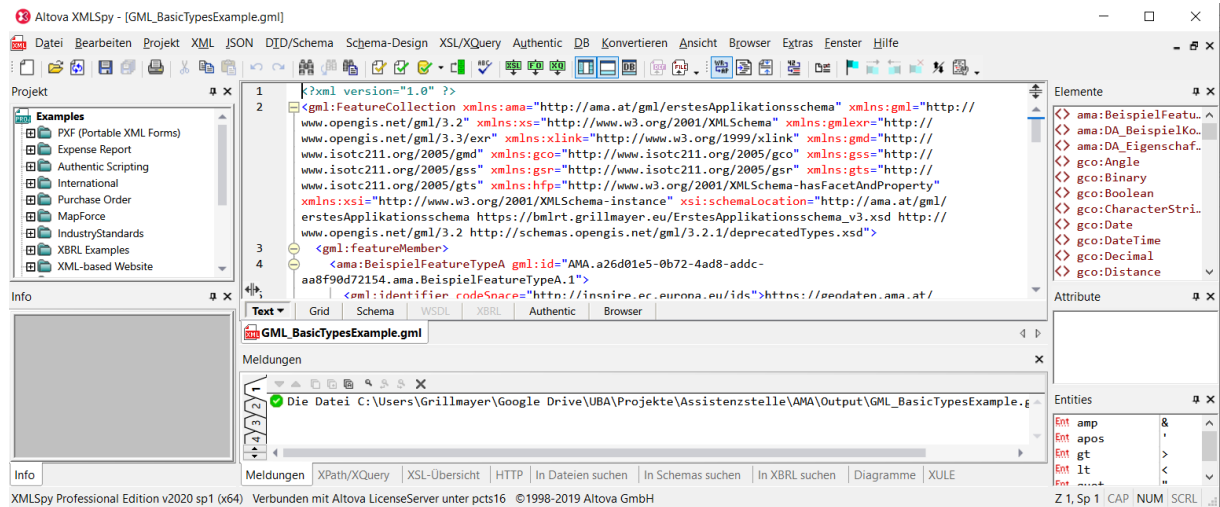
Auswahl des CRS für die GML Instanz



Ausgabe mit Zeilenumbrüchen falls als Encoding Beispiel für Entwickler

Validieren des GML Instanzen Dokuments

Abschließend soll das erstellte Dokument in einem gängigen XML Client validiert werden



GML-Instanz Dokument

```
<?xml version="1.0" ?>
<gml:FeatureCollection xmlns:ama="http://ama.at/gml/erstesApplikationsschema" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:gml:extr="http://www.opengis.net/gml/3.3/exr" xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:gmd="http://www.isotc211.org/2005/gmd" xmlns:gco="http://www.isotc211.org/2005/gco" xmlns:gss="http://www.isotc211.org/2005/gss"
xmlns:gtr="http://www.isotc211.org/2005/gtr" xmlns:gts="http://www.isotc211.org/2005/gts" xmlns:hfp="http://www.w3.org/2001/XMLSchema-
hasFacetAndProperty" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://ama.at/gml/erstesApplikationsschema
https://bmlrt.grillmayer.eu/ErstesApplikationsschema_v3.xsd http://www.opengis.net/gml/3.2 http://schemas.opengis.net/gml/3.2.1/deprecatedTypes.xsd">
  <gml:featureMember>
    <ama:BeispielFeatureTypeA gml:id="AMA.a26d01e5-0b72-4ad8-addc-aa8f90d72154.ama.BeispielFeatureTypeA.1">
      <gml:identifizier codeSpace="http://inspire.ec.europa.eu/ids">https://geodaten.ama.at/a26d01e5-0b72-4ad8-addc-
aa8f90d72154/ama.BeispielFeatureTypeA/1</gml:identifizier>
      <ama:eigenschaftA_Kardinalitaet_1_viele>Eigenschaft A mit Kardinalität = 1</ama:eigenschaftA_Kardinalitaet_1_viele>
      <ama:eigenschaftB_Kardinalitaet_1_viele>Eigenschaft B - zweite Ausprägung</ama:eigenschaftB_Kardinalitaet_1_viele>
      <ama:eigenschaftB_Kardinalitaet_1_viele>Eigenschaft B - dritte Ausprägung</ama:eigenschaftB_Kardinalitaet_1_viele>
      <ama:eigenschaftB_Kardinalitaet_1_viele>Eigenschaft B - erste Ausprägung</ama:eigenschaftB_Kardinalitaet_1_viele>
      <ama:eigenschaftC_Kardinalitaet_0_1_komplexeEigenschaft>
        <ama:DA_BeispielKomplexeEigenschaft>
          <ama:lc_class xlink:title="Landinformationssystem Österreich Nomenklatur (LISA)"></ama:lc_class>
          <ama:lc_schema>https://registry.inspire.gv.at/codelist/LISALandCoverClassValue</ama:lc_schema>
          <ama:percentage>20</ama:percentage>
        </ama:DA_BeispielKomplexeEigenschaft>
      </ama:eigenschaftC_Kardinalitaet_0_1_komplexeEigenschaft>
      <ama:eigenschaftC_Kardinalitaet_0_1_komplexeEigenschaft>
        <ama:DA_BeispielKomplexeEigenschaft>
          <ama:lc_class xlink:title="Landinformationssystem Österreich Nomenklatur (LISA)"></ama:lc_class>
          <ama:lc_schema>https://registry.inspire.gv.at/codelist/LISALandCoverClassValue</ama:lc_schema>
          <ama:percentage>70</ama:percentage>
        </ama:DA_BeispielKomplexeEigenschaft>
      </ama:eigenschaftC_Kardinalitaet_0_1_komplexeEigenschaft>
      <ama:eigenschaftD_Kardinalitaet_1_viele_DataTypeEncoding>
        <ama:DA_Eigenschaft_C>
          <ama:eigenschaftC>Eigenschaft B - zweite Ausprägung</ama:eigenschaftC>
        </ama:DA_Eigenschaft_C>
      </ama:eigenschaftD_Kardinalitaet_1_viele_DataTypeEncoding>
      <ama:eigenschaftD_Kardinalitaet_1_viele_DataTypeEncoding>
        <ama:DA_Eigenschaft_C>
          <ama:eigenschaftC>Eigenschaft B - dritte Ausprägung</ama:eigenschaftC>
        </ama:DA_Eigenschaft_C>
      </ama:eigenschaftD_Kardinalitaet_1_viele_DataTypeEncoding>
      <ama:eigenschaftD_Kardinalitaet_1_viele_DataTypeEncoding>
        <ama:DA_Eigenschaft_C>
          <ama:eigenschaftC>Eigenschaft B - erste Ausprägung</ama:eigenschaftC>
        </ama:DA_Eigenschaft_C>
      </ama:eigenschaftD_Kardinalitaet_1_viele_DataTypeEncoding>
      <ama:eigenschaftE_enumeration>Value1</ama:eigenschaftE_enumeration>
      <ama:geometry>
        <gml:Polygon srsName="http://www.opengis.net/def/crs/EPSG/0/31287" srsDimension="2">
          <gml:exterior>
            <gml:LinearRing>
              <gml:posList>474818.93796269153 629845.5684895432 474791.5592761907 629848.3063581933 474762.3553439232 629853.3257840518
474727.2193629138 629864.277258652 474698.01543064625 629897.5879938947 474694.8212505545 629956.9084813131 474699.3843649713
630048.1707696493 474703.9474793881 630153.1224012356 474706.6853480382 630217.4623145126 474708.9669052466 630234.3458378548
```

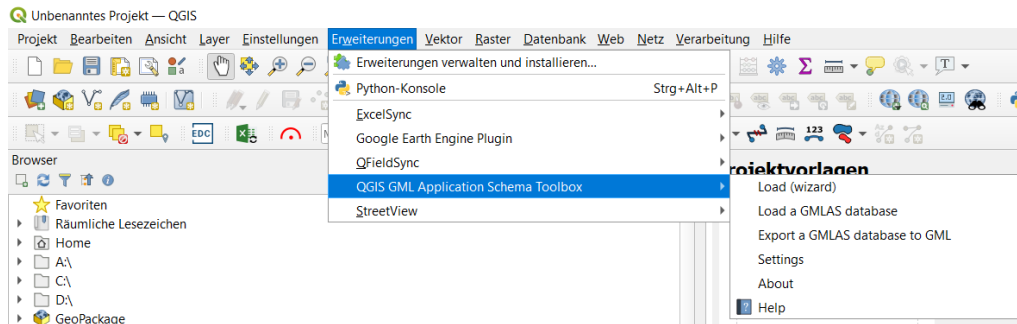
```

474735.8892803057 630237.5400179466 474765.54952401493 630236.1710836214 474771.0252613151 630234.8021492965 474775.5883757319
630191.4525623368 474781.0641130321 630142.6272380771 474783.80198168213 630078.2873248002 474789.734030424 630006.6464284563
474792.92821051576 629944.1317609462 474800.68550502433 629896.2190595698 474806.6175537662 629878.8792247858 474806.6175537662
629878.8792247858 474818.93796269153 629845.5684895432 </gml:posList>
</gml:LinearRing>
</gml:exterior>
</gml:Polygon>
</ama:geometry>
</ama:BeispielFeatureTypeA>
</gml:featureMember>
</gml:FeatureCollection>

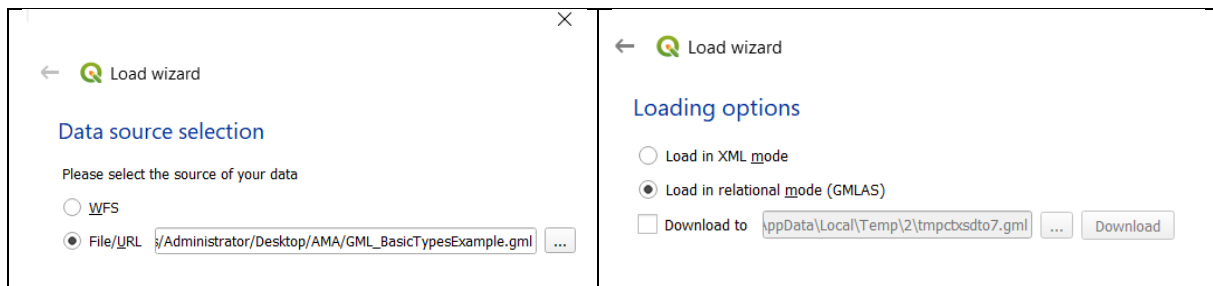
```

Einlesen der GML Instanz im QGIS

Da es sich bei der Beispielinstantz um ein GML mit komplexen Datentypen handelt – muss dieses zwingend mit der Erweiterung QGIS GML Application Schema Toolbox eingelesen werden. Diese Erweiterung lest in einen ersten Schritt das Applikationsschema ein – erzeugt das Datenbankschema (sqlite oder PostGIS) und ladet dann die GML Instanz in dieses Datenbankschema. Diese Art des Datenaustausches, in dem das Datenbankschema vorangestellt wird nennt man den „schemabezogener Datentransfer“ und stellt ein wesentliches Konzept von GML dar. Somit können mit GML komplexe Datenstrukturen wie diese in jeder normalen Datenbank vorhanden sind ausgetauscht und in GIS verfügbar gemacht werden



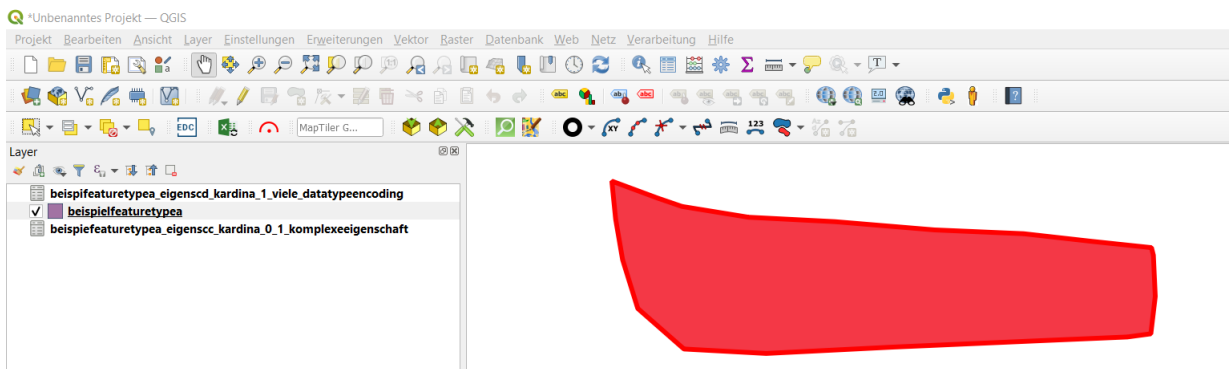
Aufrufen des Plugins über den Reiter „Erweiterungen“ – zum Laden von GML Instanzen verwendet man am besten den „Load (Wizard)“



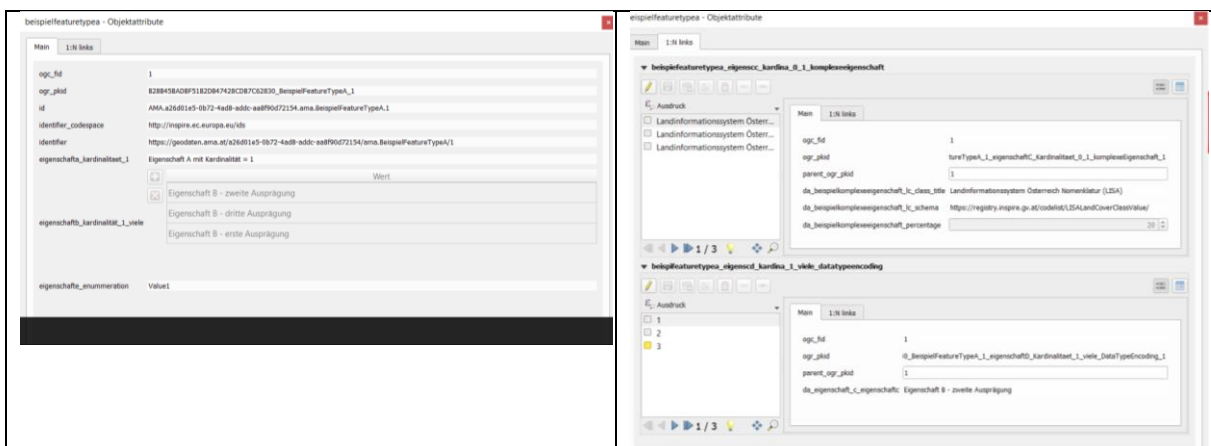
GML Instanz für den Import auswählen | Option „Load in relational mode“ auswählen

GMLAS Options

Es gibt zahlreiche Optionen für den Import – für das vorliegende GML File einfach mit den default Einstellungen den Import abschließen



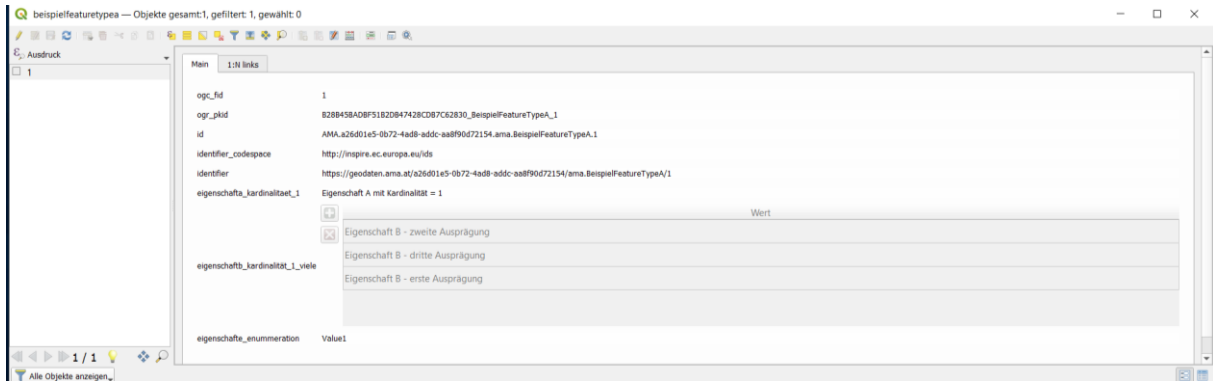
Haupttabelle und assoziierte Tabellen der komplexen Datentypen werden geladen



Formularansicht der Haupttabelle und assoziierten (1:n) Tabellen

Erörterung der importierten Datei

Nicht komplexe Eigenschaften werden immer in einer Spalte in der Haupttabelle abgebildet. Liegt eine Kardinalität von mehr als 1 wie zum Beispiel bei der „eigenschaftB_kardinalitaet_1_viele“ vor – so wird diese in Form eines Arrays abgebildet.



eigenschaftA_kardinalitaet_1	eigenschaftB_kardinalitaet_1_viele	eigenschaftC_enumeration
Eigenschaft A mit Kardinalität = 1	Eigenschaft B - zweite Ausprägung, Eigenschaft B - dritte Ausprägung, Eigenschaft B - erste Ausprägung	Value1

Abbildung einer Eigenschaft mit Kardinalität 1...* im importierten Datensatz

Will man diese Art der Repräsentation im **importierten Datensatz** vermeiden will und Eigenschaften welche eine Kardinalität von 1...* auf weisen in assoziierten Tabellen repräsentiert werden sollen – muss man das für die eigenschaftC verwendeten GML Encoding anwenden. Für die eigenschaftC wurde ein komplexer Datentyp definiert (obwohl es sich um einen einfachen Datentyp handelt der nur eine Eigenschaft repräsentiert). Dadurch wird dieser wie ein komplexer Datentyp beim Import encodiert und in Form einer assoziierten Tabelle abgebildet.

ogc_fid	ogr_pkid	parent_ogr_pkid	da_eigenschaft_c_eigenschaftc
1	B288458ADB51B2DB47428CD87C62830_BeispielFeatureTypeA_1_eigenschaftD_Kardinalitaet_1_viele_DataTypeEncoding_1	B288458ADB51B2DB47428CD87C62830_BeispielFeatureTypeA_1	Eigenschaft B - zweite Ausprägung
2	B288458ADB51B2DB47428CD87C62830_BeispielFeatureTypeA_1_eigenschaftD_Kardinalitaet_1_viele_DataTypeEncoding_2	B288458ADB51B2DB47428CD87C62830_BeispielFeatureTypeA_1	Eigenschaft B - dritte Ausprägung
3	B288458ADB51B2DB47428CD87C62830_BeispielFeatureTypeA_1_eigenschaftD_Kardinalitaet_1_viele_DataTypeEncoding_3	B288458ADB51B2DB47428CD87C62830_BeispielFeatureTypeA_1	Eigenschaft B - erste Ausprägung

Assoziierte Tabelle der eigenschaftC des importierten Datensatzes