



**Kanton Zürich  
Baudirektion  
AWEL Energie**

# Anleitung

**Erstellung der kommunalen Energieplanung nach Vorgaben des kantonalen Geodatenmodells (KGDM) in ArcMap Desktop**

# Informieren

Orientieren Sie sich vorgängig über Modellvorgaben und die Unterscheidung der Objektklassen:

- **Modelldokumentation:** [Link](#)
- **Klassenübersicht:** [Link](#)



Die vorliegende Anleitung liefert **notwendige Angaben** zur korrekten Erfassung der Klassen und insbesondere deren Beziehungen zueinander (siehe *T\_Id* und *GemeindeRef* / *VerbundgebietRef*)

# Datenbank vs. Projekt

## Datenbank

*(kgdm\_komm\_ep\_arcmap\_desktop.gdb)*

- Klassen
- Attribute

## Projekt

*(kgdm\_komm\_ep\_arcmap\_desktop.mxd)*

- Datenbank integriert
- Darstellung (Symbolisierung)

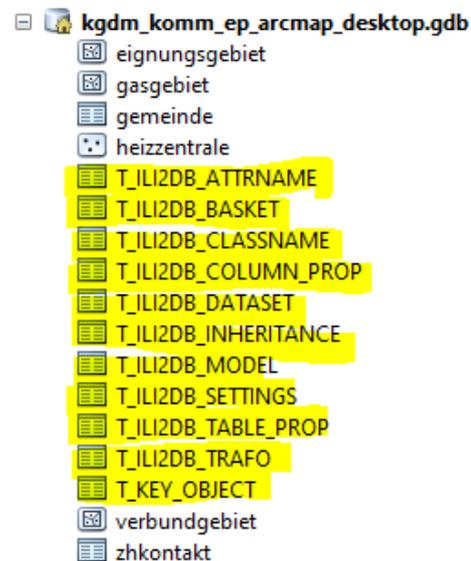


Es wird empfohlen, direkt im Vorlage-Projekt zu arbeiten. Die Datenbank ist dem Projekt bereits hinzugefügt und die dem KGDM entsprechende Symbolisierung ist schon erstellt.

# Arbeiten mit der Geodatabase

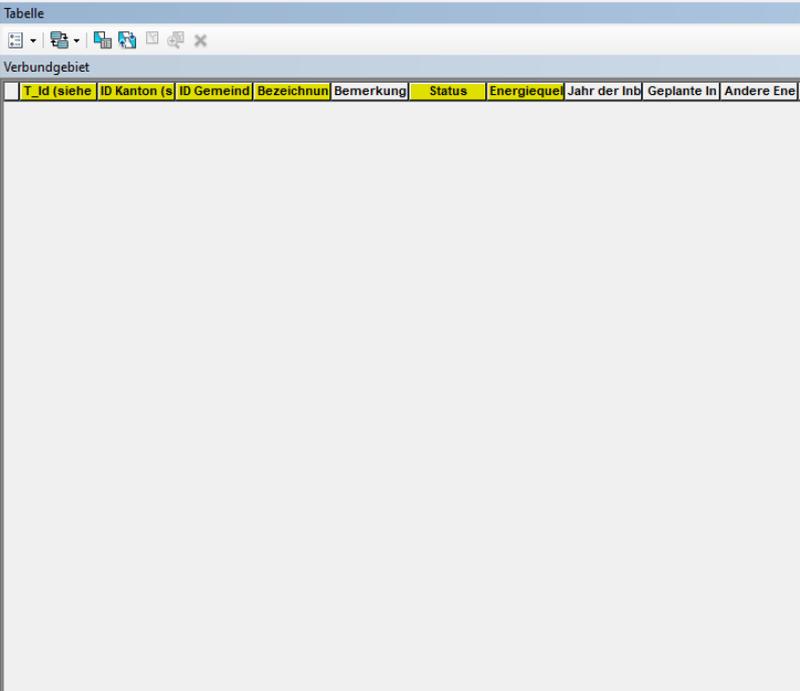
# Features erstellen

Die hervorgehobenen Tabellen der File-Geodatabase **sind leer zu lassen**. Sie sind notwendig für die Umwandlung ins Interlis.



# Features erstellen

In der Attributtabelle sind jeweils diejenigen Attribute gelb markiert, die in jedem Fall ausgefüllt werden müssen.



T_Id (siehe)	ID Kanton (s)	ID Gemeind	Bezeichnung	Bemerkung	Status	Energieque	Jahr der Inb	Geplante In	Andere Ene
--------------	---------------	------------	-------------	-----------	--------	------------	--------------	-------------	------------

# T\_Id

Das Attribut *T\_Id* ist für die Identifikation und die Umwandlung ins Interlis notwendig. Daher müssen die Werte für dieses Attribut zwingend richtig und pro Feature einzigartig sein. Dazu wird nach folgendem Schlüssel vorgegangen (TID Vorgabe pro Klasse mit BFS-Nr. als Präfix):

*bfsnr \* 1000 (Bsp. Gemeinde Zürich: 261000 plus 100-600 plus 1 - 99)*



siehe Einteilung der Klassen auf der nächsten Seite

# T\_Id

Die T\_Id setzt sich für die verschiedenen Klassen folgendermassen zusammen:

- **Gemeinde:** *bfsnr101*
- **Verbundgebiet:** *bfsnr201* für erstes Objekt, *bfsnr202* für zweites Objekt
- **Heizzentrale:** *bfsnr301* für erstes Objekt, *bfsnr302* für zweites Objekt
- **Eignungsgebiet:** *bfsnr401* für erstes Objekt, *bfsnr402* für zweites Objekt
- **Gasgebiet:** *bfsnr501* für erstes Objekt, *bfsnr502* für zweites Objekt
- **ZHKontakt:** *bfsnr601* (pro Gemeinde kann nur eine Kontaktangabe gemacht werden)



Die gewählte Nummerierung ermöglicht 99 Einträge pro Klasse

# T\_Id – Beispiel Gemeinde Zürich

Klasse	Objekt / Eintrag	Identifikator ( <i>T_Id</i> )
Gemeinde	Zürich 	261101
Verbundgebiet	Gebiet 1 (in_Betrieb) 	261201
	Gebiet 2 (in_Planung) 	261202
Heizzentrale	Zentrale 1 	261301
	Zentrale 2 	261302
Eignungsgebiet	Gebiet 1 (Erdwärme) 	261401
	Gebiet 2 (Abwärme) 	261402
Gasgebiet	Gebiet 1 (Stilllegung) 	261501
	Gebiet 2 (Fortbestand) 	261502
ZHKontakt	Eintrag 	261601

# ID\_Gemeinde

Für das Attribut *ID\_Gemeinde* kann die Gemeinde einen eigenen Identifikationsschlüssel definieren. Zum Beispiel folgendermassen:

Gebiet		Identifikator ( <i>ID_Gemeinde</i> )
Verbundgebiet 1 (in_Betrieb)		VG1
Verbundgebiet 2 (in_Planung)		VG2
Gasgebiet 1		GG1

 Jedem Feature muss unter *ID\_Gemeinde* ein einzigartiger Identifikationswert zugewiesen werden.

# ID\_Kanton

Der Wert für das Attribut *ID\_Kanton* wird zusammengesetzt aus der BFS-Nr. der Gemeinde, einem Underscore <\_> und dem jeweiligen Wert für *ID\_Gemeinde*.  
Zum Beispiel folgendermassen:

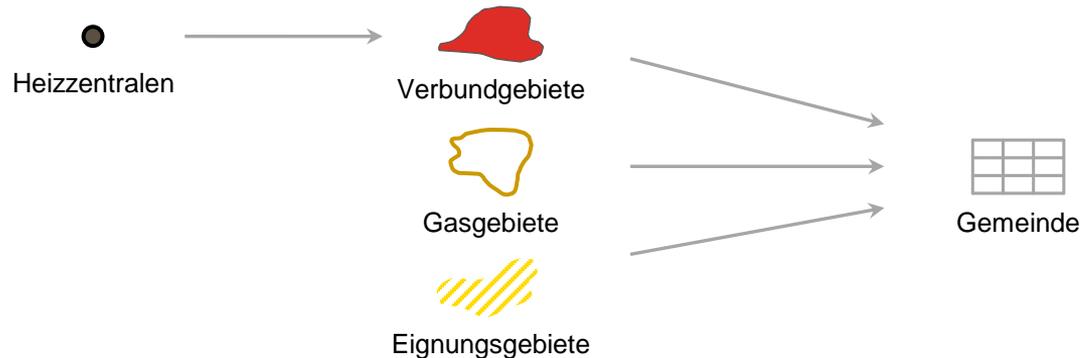
Gebiet		Identifikator ( <i>ID_Kanton</i> )
Verbundgebiet 1 (in_Betrieb)		230_VG1
Verbundgebiet 2 (in_Planung)		230_VG2
Gasgebiet 1		230_GG1



Das Hinzufügen der BFS-Nummer stellt die Eindeutigkeit aller im kantonalen Geodatenmodell (also im gesamten Kanton) definierten Features sicher.

# GemeindeRef / VerbundgebietRef

Verbund-, Gas- und Eignungsgebiete müssen immer einer Gemeinde zugeordnet werden. Heizzentralen müssen immer einem Verbundgebiet (in Betrieb) zugeordnet werden.



Da sich alle Gebiete auf eine Gemeinde beziehen müssen, ist die Erfassung der Gemeinde in der Gemeindetabelle zwingend.

# GemeindeRef / VerbundgebietRef

In den Attributen *GemeindeRef* und *VerbundgebietRef* ist die *T\_Id* des Bezugsobjekts (Gemeinde, respektive Verbundgebiet) einzutragen.

Gemeinde	← Verbundgebiet	← Heizzentrale
		
T_Id	261201	261301
BFSNr	261_VG1	261_HZ1
Gemeindename	VG1	HZ1
...	Status	VerbundgebietRef
	in_Betrieb	261201
	in_Planung	261201
	GemeindeRef	261202
	261101	261302
	261101	261_HZ2
		HZ2
		261303
		261_HZ3
		HZ3
		...

! Aufgrund der Beziehungen ist **IMMER** zuerst eine Gemeinde zu definieren und entsprechende Verbundgebiete vor deren jeweiligen Heizzentralen.

# ZHKontakt

- Die Tabelle **ZHKontakt** ermöglicht es, verschiedene Angaben zur jeweiligen Kontaktstelle zu machen.
- Eine Telefonnummer und eine Mailadresse sind bei der Erstellung eines ZHKontakt-Eintrages zwingend.
- Es kann modellbedingt maximal eine Zeile ausgefüllt werden.
- Im Feld *T\_Seq* muss immer der Wert «1» stehen.
- Der **ZHKontakt**-Eintrag muss immer über das *GemeindeRef* Attribut mit der entsprechenden Gemeinde verknüpft werden (*T\_Id* der Gemeinde).



Das Attribut *GemeindeRef* heisst eigentlich *Gemeinde\_Kontakt*. Der Aliasname wurde aber einfachheitshalber analog zu den anderen Klassen zu *GemeindeRef* umbenannt.

# Datumsangaben Gemeinde

## ***Kantonales\_Verfuegungsdatum***

- Da das kantonale Verfügungsdatum nach Modellvorgabe zwingend anzugeben ist, soll bei der Erstellung der Energieplanung für dieses Attribut ein Platzhalter-Wert eingetragen werden.
- Das korrekte Verfügungsdatum wird durch die Baudirektion des Kantons Zürich eingetragen.

## ***Letzte\_Aktualisierung***

- Unter diesem Attribut ist das Datum anzugeben, an welchem das letzte Mal Änderungen am Geodatensatz vorgenommen wurden.



Der Platzhalter-Wert für das Verfügungsdatum muss zwischen 1582-01-01 und 2999-12-31 liegen.

# Vorlauftemperaturniveau

- Die Attribute **Vorlauftemperaturniveau\_1** und **Vorlauftemperaturniveau\_2** sind optional und geben Auskunft über die Temperatur im Verbund und darüber, ob gegebenenfalls auch Kälte geliefert wird.
- Es ist nicht vorgegeben, welches Attribut das höhere Temperaturniveau abbildet.



Bei Netzen mit Warm- und Kaltleitern können die jeweiligen Temperaturen in beliebiger Reihenfolge den beiden Attributen zugewiesen werden.

# Abschluss der Datenerfassung

- Im ArcMap Desktop gibt es teilweise Probleme mit Attributwerten, die eigentlich NULL enthalten sollten, aber effektiv einfach leer sind.
- Nach Erfassung aller Daten und vor Umwandlung ins XTF (Interlis) ist **zwingend** ein Python-Befehl (s. folgende Seite) zur Behebung dieses Problems auszuführen.

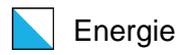
Python

```
>>>
```

F1 zeigt Hilfe für aktuelle Cursorposition an.  
 F2 prüft die Syntax der aktuellen Zeile (oder des Code-Blocks beim Mehrzeilenmodus).  
 Mit ESC wird der aktuelle Vorgang abgebrochen.  
 Umschalttaste oder Strg+Eingabetaste drücken wechseln in den Mehrzeilenmodus. Zum Ausführen klicken

Katalog Python Features erstellen

Code aus dem mitgelieferten Textdokument (Abschluss\_Code.txt) in das Python Fenster im ArcMap kopieren und mit (ggf. 2x) ENTER ausführen



Abschluss\_Code.txt - Editor

Datei Bearbeiten Format Ansicht Hilfe

```
import arcpy

# Get the current workspace
workspace = arcpy.env.workspace

# If the workspace is not set, use the default geodatabase
if workspace is None:
    workspace = arcpy.GetParameterAsText(0) # If running as a script
    tool, this gets the default geodatabase
    if workspace == '' or workspace.lower() == 'none':
        workspace = arcpy.env.scratchGDB # If still not set, use the
        scratch geodatabase

# Set the workspace to the determined geodatabase
arcpy.env.workspace = workspace

# Get a list of all feature classes and tables in the geodatabase
datasets = arcpy.ListFeatureClasses() + arcpy.ListTables()

# Iterate through each dataset
for dataset in datasets:
    # Iterate through each field in the dataset
    fields = arcpy.ListFields(dataset)
    for field in fields:
        # Check if the field is editable (not OID or geometry)
        if not field.editable:
            continue
        # Update the field values to NULL if empty
        with arcpy.da.UpdateCursor(dataset, field.name) as cursor:
            for row in cursor:
                if row[0] == '' or row[0] == ' ':
                    row[0] = None
                    cursor.updateRow(row)
```

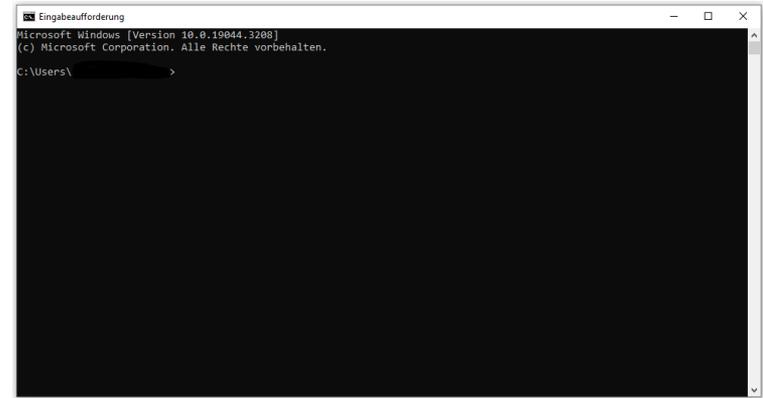
Zeile 1, Spalte 1 100% Windows (CRLF) UTF-8

# Umwandlung File-Geodatabase zu XTF

# ili2fgdb

Datei: ili2fgdb-4.11.0.jar

Benötigtes Programm: [Java](#)



! Das Umwandlungstool *ili2db* unterstützt noch keine grafische Benutzeroberfläche (GUI), weshalb die Umwandlung ins xtf über die Kommandozeile (Windows: z.B. «Eingabeaufforderung») gemacht werden muss.

# ili2fgdb - Umwandlungscode

Folgender Code wird für die Umwandlung benötigt:

```
java.exe -jar „“ --export --models  --modeldir „“ --dbfile „“ „“
```

: Pfad zur ili2db-Anwendung → C:\Musterpfad\ili2fgdb-4.11.0.jar

: Modellname → Energieplanung\_kommunal\_ZH\_V1\_LV95

: Pfad zum Modellordner → C:\Musterpfad\ilimodels

: Pfad zur Geodatabase → C:\Musterpfad\kgdm\_komm\_ep\_arcgis\_pro.gdb

: Gewünschter Speicherort der XTF-Datei: C:\Musterpfad\kgdm\_komm\_ep\_arcgis\_pro.xtf

 Falls der Befehl java.exe nicht gefunden werden kann, muss der gesamte Pfad zur Java-Anwendung angegeben werden

 Der Modellname ist **nicht** Energieplanung\_kommunal\_ZH\_V1\_LV95c.ili oder ähnlich. Er wird aus der Modelldatei gelesen und lautet immer **Energieplanung\_kommunal\_ZH\_V1\_LV95**

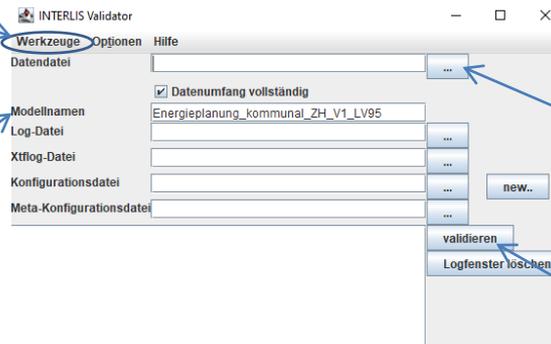
# Validierung der XTF-Datei

# ilinvalidator

- Im Gegensatz zu *ili2db* unterstützt der *ilinvalidator* eine grafische Benutzeroberfläche (GUI) und kann somit ganz einfach per Doppelklick geöffnet werden.
- Da der *ilinvalidator* bereits im *ili2db* enthalten ist, können die Daten grundsätzlich gar nicht in ein XTF umgewandelt werden, sollten die Modellvorgaben nicht eingehalten werden.
  - Es ist trotzdem **in jedem Fall** eine Prüfung mit dem *ilinvalidator* zu machen.

# ilvalidator

Unter «Werkzeuge» ist **zwingend**  
das Modellverzeichnis anzupassen  
→ C:\Musterpfad\ilmodels



Das Modell lautet immer  
«Energieplanung\_kommunal\_ZH\_V1\_LV95»

Die entsprechende  
XTF-Datei auswählen

XTF-Datei validieren



Bei einer erfolgreichen Validierung sollte in der letzten Zeile des Logfensters *Info: ...validation done* stehen.

# **Einreichung der GIS-Daten beim Kanton**

# Einreichung

Nach erfolgreicher Validierung mit dem ilvalidator ist zur Publikation auf dem GIS-Browser einzig die XTF-Datei dem Kanton abzugeben.