



## Objektkatalog für das Straßen- und Verkehrswesen Modellierung "Grunderwerbsverzeichnis"

Version: n/a  
Datum: 28.07.2000  
Status: in Bearb.  
Dateiname: N0009.doc  
Verantwortlich: D. König

### **OKSTRA-Pflegestelle**

interactive instruments GmbH  
Trierer Straße 70-72  
53115 Bonn

<http://www.okstra.de/>

Herr Clemens Portele  
Tel. 0228 91410 73  
Fax 0228 91410 90  
Email [portele@interactive-instruments.de](mailto:portele@interactive-instruments.de)

### **Im Auftrag von**

Bundesanstalt für Straßenwesen  
ZD - OKSTRA  
Brüderstraße 53  
51427 Bergisch Gladbach

Herr Alfred Stein  
Tel. 02204 43 354  
Fax 02204 43 673  
Email [stein@bast.de](mailto:stein@bast.de)



## 0 Allgemeines

### 0.1 Inhaltsverzeichnis

<b>0 Allgemeines</b> .....	<b>2</b>
0.1 Inhaltsverzeichnis .....	2
0.2 Abkürzungen und Definitionen .....	2
0.3 Abbildungsverzeichnis .....	2
0.4 Tabellenverzeichnis .....	2
0.5 Bezüge .....	2
0.6 Änderungen .....	3
0.7 Bearbeitungsvermerke .....	3
<b>1 Zweck des Dokuments</b> .....	<b>4</b>
1.1 Leserkreis .....	4
1.2 Kernaussagen des Inhalts .....	4
<b>2 NIAM-Modellierung</b> .....	<b>5</b>
<b>3 Erläuterungen</b> .....	<b>7</b>
3.1 Begriffsdefinition und Grundlagen .....	7
3.2 Objektdefinitionen .....	7
3.3 Diskussionspunkte .....	9
<b>4 EXPRESS-Schema</b> .....	<b>11</b>
<b>5 SQL-Schema</b> .....	<b>20</b>
<b>6 Anhang: Referenzen anderer Schemata</b> .....	<b>27</b>

### 0.2 Abkürzungen und Definitionen

GE	Grunderwerb
GEV	Grunderwerbsverzeichnis

### 0.3 Abbildungsverzeichnis

Abbildung 1 - NIAM-Diagramm "Grunderwerbsverzeichnis" .....	6
---	---

### 0.4 Tabellenverzeichnis

Tabelle 1 - Historie des NIAM-Diagramms "Grunderwerbsverzeichnis" .....	5
---	---

### 0.5 Bezüge

Dokument	Bemerkungen
----------	-------------



Dokument	Bemerkungen
OKSTRA-Web-Seite	<a href="http://www.okstra.de/">http://www.okstra.de/</a>
D002	Teilschema 002: Administration - NIAM, zu finden auf der OKSTRA-Web-Seite
E002	Teilschema 002: Administration - EXPRESS, zu finden auf der OKSTRA-Web-Seite
S002	Teilschema 002: Administration - SQL, zu finden auf der OKSTRA-Web-Seite

## 0.6 Änderungen

Name	Datum	Kapitel	Bemerkungen	Bearbeiter
N0009	20.06.2000	alle	Dokument erstellt, in Bearbeitung	D. König
N0009	07.07.2000	alle	Dokument überarbeitet	D. König
N0009	27.07.2000	alle	Dokument erneut überarbeitet	D. König

## 0.7 Bearbeitungsvermerke

- Die Eintragung in die ACCESS-DB steht noch aus. Dies erfolgt nach der Abstimmung.



# 1 Zweck des Dokuments

## 1.1 Leserkreis

Das Dokument richtet sich an die BG10 und die BG24 sowie an alle Experten des Bereichs "Grunderwerbsverzeichnis" im Straßen- und Verkehrswesen.

Vorausgesetzt werden Kenntnisse

- der grundlegenden OKSTRA-Standards (NIAM, EXPRESS und SQL),
- zum OKSTRA und seinen Regularien (siehe auch <http://www.okstra.de/>).

## 1.2 Kernaussagen des Inhalts

Der Auftrag Nr. 17 der BG10 vom 05.06.2000 umfasst die Modellierung eines Schemas zum Grunderwerbsverzeichnis im OKSTRA.

In dem Dokument wird die Umsetzung dieses Auftrags dokumentiert, d.h. die Modellierung

- in NIAM,
- in EXPRESS und
- in SQL.



## 2 NIAM-Modellierung

Grundlage der Modellierung waren die fachlichen Vorgaben aus dem Anhang zum Änderungsantrag A0002 sowie der dazugehörigen MS-ACCESS-Datenbank (GEVProto.mdb), die uns Herr Feser (AKG) im Auftrag der BG24 übermittelt hat. Die Informationen dieser Vorgaben wurden umgesetzt. Dabei enthält das NIAM-Diagramm zur Erhöhung der Übersichtlichkeit keine einfachen Attribute, diese sind aber selbstverständlich im EXPRESS-Schema wie auch im SQL-Schema enthalten.

Zur Historie des Diagramms:

Grunderverbsverzeichnis			dxxx.cfl
Datum	Versionsnr.	Beschreibung der Änderungen	
20.06.2000	0.01	?.??	Schema modelliert
07.07.2000	0.02	?.??	Schema (nach Antworten von AKG) überarbeitet

**Tabelle 1 - Historie des NIAM-Diagramms "Grunderverbsverzeichnis"**





## 3 Erläuterungen

### 3.1 Begriffsdefinition und Grundlagen

Im Schema Grunderwerb werden die Objekttypen definiert, die zur Beschreibung der für ein Bauvorhaben (im Straßenwesen) zu erwerbenden Flächen notwendig sind.

Grundlage der vorliegenden im Rahmen der OKSTRA-Pflege durchgeführten Neu-Modellierung waren die fachlichen Vorgaben der BG24, übermittelt durch Herrn Feser (AKG) in Form eines Änderungsantrages (A0002).

### 3.2 Objektdefinitionen

Die folgende Tabelle erläutert die in Abbildung 1 - NIAM-Diagramm "Grunderwerbsverzeichnis" definierten Objekte:

Objekt	Definition, Erläuterung
Erwerbsfläche	Dieser Begriff beschreibt die Teilfläche(n) eines Flurstückes oder die gesamte Fläche des Flurstückes, die für eine Baumaßnahme in einer bestimmten Erwerbsart benötigt werden. Das Ergebnis kommt durch die Planung zustande. Ein Flurstück kann also mehrere verschiedene Erwerbsflächen haben.
Erwerbsart	EA1 = Erwerb: Die Erwerbsfläche soll für immer dem Eigentümer entzogen werden. Beispielsweise wird dort die Straße gebaut o.ä. Der Vorgang mündet in einen Kaufvertrag mit dem Baulastträger. EA2 = Anmietung: Diese Erwerbsflächen werden für eine gewisse Zeit während der Baumaßnahme benötigt (Bauschuttablagerung, Baufahrzeuge, Zufahrtswege ...). Danach bekommt der Eigentümer sie wieder. Für die Nutzung erhält er Entschädigungen und Pachtgebühren. EA3 = zukünftige Belastung oder Dienstbarkeit (nicht zu verwechseln mit den momentanen Belastungen des Flurstückes). Beispiel: Es muss wegen der Notrufsäulen der Autobahn ein Kabel durch ein Grundstück gelegt werden. Deswegen darf an bestimmten Stellen nichts gebaut werden. Ansonsten kann der Eigentümer das Grundstück für landwirtschaftliche Zwecke nutzen. Er bekommt dafür eine Entschädigung. Oder ein Grünplaner beschließt auf dem Grundstück eine Baumreihe mit Büschen zu pflanzen. Diese Flächen werden aber nicht gekauft, sondern der Eigentümer erhält beispielsweise die Auflage, diese mit seinem Traktor nicht zu beschädigen und einmal im Monat einen Gärtner auf sein Grundstück zu lassen. Dafür erhält er Entschädigungszahlungen.
Erwerbzweck	Nähere Erläuterung der Erwerbsart: Erwerb: EZ = für den Baulastträger: also der klassische Erwerb. Erwerb: EZ = für Dritte: beispielsweise wird für eine Imbisskette der Erwerb mit getätigt, da diese dann an der Raststätte eine Filiale errichten will. Erwerb: EZ = für LBP: hier wird Land von der Straßenbauverwaltung gekauft, um Landschaftsbegleitende Pflegemaßnahmen zu realisieren. Anmietung: EZ = für private Bauarbeiterversorgungseinrichtungen. Nicht alle Kombinationen von EA und EZ sind sinnvoll, aber die Möglichkeiten vielfältig.



Nutzungsart	Die Nutzungsart beschreibt, wie ein Flurstück genutzt wird, bzw. wie die Nutzung eines Flurstücksteiles ist, das von einer Erwerbsfläche betroffen ist. Dies ist wichtig für die spätere Entschädigungshöhe.
GE_Massnahme	Die GE_Massnahme beschreibt die Grunderwerbs-Maßnahme näher, z.B. Maßnahme mit Kurztext „L15.3a“ - Erläuterung „Erhaltung/Erweiterung einer Biotopfläche“.
Nutzungsartteilflächen	Ein Flurstück kann mehrere Teilflächen mit verschiedenen Nutzungsarten haben, z.B. eine Hälfte Wald, ein Viertel Acker, der Rest Obstplantage. Dies ist relevant für den Grunderwerb, da die Höhe der Entschädigung von der (ehemaligen) Nutzung abhängt.
Belastung	Beispiel: Das Flurstück eines Bauern hat eine dingliche Belastung (Belastung Abt. 2) vom Energiebetrieb, weil durch sein Grundstück ein dickes Kabel geht. Er kann an diesen Stellen nicht alles mit seinem Grundstück machen. Oder er hat ein Wegerecht, damit der Nachbar zu seinem Grundstück gelangen kann. Oder er hat eine Hypothek (Belastung Abt. 3) von einer Bank. Die Belastungen zum Flurstück sind im Grundbuch eingetragen. All das hat Einfluss auf die Planung und die Kaufverhandlungen.
Belastungsberechtigter	Die Namen und Adressen der o.g. Personen (Energiebetrieb und Bank)
Flurstück	Grundbesitz von Eigentümern an Grund und Boden - im Grundbuch formuliert. Ein Flurstück hat eine eindeutige Nummer innerhalb einer Flur bzw. Gemarkung.
Attribut Flurstück	Dies ist ein informativer textlicher Zusatz zu einem Flurstück, vom Nutzer selbst zu definieren. Es ist kein direkter Bestandteil eines GEV's, eher eine Erweiterung aus der praktischen Arbeit, einem Flurstück freie Informationen zuordnen zu können.
Person Flurstück	Beschrieben werden keine Eigentümer, Pächter oder Mieter, sondern: Hausmeister, Verwaltungen, Gärtner, Makler uvm. Diese sind nur im erweiterten Sinne direkt dem GEV zugehörig.
Grundbuch	Dokument auf dem Grundbuchamt, in dem die Flurstücke, deren Eigentümer, Belastungen u.a. eingetragen sind.
Flurstück Grundbuch	Ein Grundbuch kann mehrere Flurstücke enthalten (also ein oder mehrere Eigentümer des Grundbuches haben mehrere Flurstücke). Zu einem Flurstück können aber auch mehrere Grundbücher existieren. Das ist immer der Fall bei Wohneigentum oder Erbbaurecht. Jede(r) Wohnungseigentümer hat sein eigenes Grundbuch und es gibt i.a. mehrere Wohnungseigentümer auf dem selben Flurstück. Es wird hier eine beidseitig multiple Relation zwischen Flurstück und Grundbuch verwendet, die weitere Attribute zum Besitzverhältnis trägt (objektifizierte Relation).
Eigentümer Grundbuch	In einem Grundbuch (mit einem/mehreren Flurstücken) können ein oder mehrere Eigentümer enthalten sein (z.B. Mann zu 2/5, Frau zu 2/5 und Kind zu 1/5). Natürlich kann auch ein Eigentümer mehrere Grundbücher besitzen. Es wird hier eine beidseitig multiple Relation zwischen Eigentümer und Grundbuch verwendet, die weitere Attribute zum Besitzverhältnis trägt (objektifizierte Relation).
Pächter/Mieter	Flurstückseigentümer können ihr Eigentum verpachten oder vermieten, falls sie selbst es nicht nutzen wollen (z.B. an Bauern)
Person	In diesem Objekttyp werden Details zu allen Personen, die im Grunderwerbsverzeichnis benutzt werden abgelegt, also Eigentümer, Pächter,



	Belastungsberechtigte, weitere Personen ...
Gemarkung	<p>Eine bestimmte Landfläche. Kleiner als der Gemeindebereich und größer als die Flur (wenn eine vorhanden ist).</p> <p>Ein Bundesland hat mehrere Landkreise Ein Landkreis hat mehrere Gemeinden Eine Gemeinde hat mehrere Gemarkungen Eine Gemarkung hat mehrere Flure Eine Flur hat mehrere Flurstücke (in einigen Regionen, z.B. Bayern oder Teile von Thüringen) gibt es keine Flur, da sind die Flurstücke innerhalb einer Gemarkung eindeutig.</p> <p>Hinweis: im allgemeinen Sprachgebrauch ist die Flurstücksnummer die Bezeichnung für die kleinste Zelle in dieser o.g. Einteilung (also Zähler und ggf. Nenner).</p>

### 3.3 Diskussionspunkte

- Objektklassen mit fachlich fest vorgegebenem Wertebereich werden im OKSTRA als Schlüsseltabellen abgebildet. Diese Werte werden dann Teil des Standards selbst. Es ist nicht vorgesehen, dass die Inhalte dieser Objektklassen vom Anwender erweitert oder verändert werden. Hier ist zu diskutieren, für welche Objektklassen eine fachliche Festlegung möglich ist. Mögliche Kandidaten für Schlüsseltabellen sind:
  - Land
  - Personenklasse
  - Eigentumsart
  - Eigentumsverhaeltnis
  - Erwerbsart
  - Erwerbzweck
  - GE\_Massnahme
  - GEV\_Ausgabespalte
  - Verhaeltnis
  - Vertragsart

Die Wertebereiche sollten verbindlich für den OKSTRA festgelegt werden, wo dies möglich ist. Es ist zu prüfen ob die im Datenschema gegebenen Wertebereiche korrekt und vollständig sind. Ist keine Festlegung möglich müssten diese Werte als normale Attribute abgebildet werden.

Nach Rückmeldung von AKG werden die Nutzungsarten länderspezifisch verschlüsselt. Das verhindert die Verwendung einer Schlüsseltabelle, obwohl dies strukturell die korrekte Lösung wäre. Es sollte mittelfristig überlegt werden, ob hier nicht eine Vereinheitlichung möglich ist.

- Nach Aussage von AKG ist eine Historisierung im GEV nicht vorgesehen. Dies sollte meiner Ansicht nach noch einmal diskutiert werden. Zeitliche Gültigkeitsbedingungen sind offensichtlich erforderlich und werden derzeit über gesonderte Attribute dargestellt oder sollen nach Rücksprache mit AKG als „gelöscht“-Informationen abgebildet werden. Eine Integration der Historisierung, d.h. zeitlicher Gültigkeit und evtl. zeitlichen Vorgänger-Nachfolger-Beziehungen, wäre



hier eine umfassende und einheitliche Lösung. Das Konzept dazu ist im OKSTRA bereits beschrieben und wird in den meisten anderen Fachbereichen eingesetzt.

- Die Anbindung des GEV an die übrigen im OKSTRA abgebildeten Fachbereiche erscheint mir etwas schwach ausgebildet. Da es sich um einen Teil von Planung und Entwurf handelt sollte die Möglichkeit einer engeren Anbindung an die Objektklassen des TP2 geprüft werden.



## 4 EXPRESS-Schema

Die Umsetzung des NIAM-Diagramms unter Berücksichtigung der Festlegungen der Bezugsdokumente ergibt das folgende EXPRESS-Schema. Eine Historisierung ist in diesem Schema vorerst nicht vorgesehen. Die untenstehenden EXPRESS-Definitionen bilden nach der Abstimmung ein neues EXPRESS-Schema Grunderwerb.

```
SCHEMA Grunderwerb;
```

```
(*  
Historie:
```

```
26.07.2000 - 0.002  
Modellierungsvorschlag überarbeitet
```

```
19.06.2000 - 0.001  
Modellierungsvorschlag erstellt
```

```
29.06.2000 - 0.002  
Antworten AKG eingearbeitet  
*)
```

```
REFERENCE FROM Administration (Bundes-  
land,Kreis_kreisfreie_Stadt,Gemeindebezirk);
```

```
REFERENCE FROM Allgemeine_Objekte (Datum,Kilometer,Quadratmeter,Euro);
```

```
ENTITY Flurstueck;
```

```
--- Attribute :
```

```
Flurstueck_Zaehler           : STRING(5);  
Flurstueck_Nenner           : OPTIONAL STRING(3);  
Flaeche                       : OPTIONAL Quadratmeter;  
Lage                          : OPTIONAL STRING(100);  
Baukilometer                 : OPTIONAL Kilometer;  
Liegenschaftsbuch           : OPTIONAL STRING(5);  
Flurkarte                     : OPTIONAL STRING(50);  
Bemerkung                    : OPTIONAL STRING(100);
```

```
--- Relationen :
```

```
hat_Attribute                : OPTIONAL SET[1:?] OF Attribut_Flurstueck;  
hat_Erwerbsflaechen         : OPTIONAL SET[1:?] OF Erwerbsflaeche;  
hat_Nutzungsartteilflaechen : OPTIONAL SET[1:?] OF Nutzungsartteilflaeche;  
hat_Belastungen             : OPTIONAL SET[1:?] OF Belastung;  
in_Gemarkung                : Gemarkung;  
in_Flur                     : OPTIONAL Flur;
```

```
INVERSE
```

```
in_Grundbuch                : SET[0:?] OF Flurstueck_Grundbuch  
                             FOR Flurstueck;  
zu_Person                   : SET[0:?] OF Person_Flurstueck  
                             FOR Flurstueck;  
hat_Paechter_Mieter         : SET[0:?] OF Paechter_Mieter  
                             FOR von_Flurstueck;
```

```
END_ENTITY;
```

```
ENTITY Flur;
```

```
--- Attribute :
```

```
Kennung                      : STRING(3);
```

```
--- Relationen :
```

```
INVERSE
```

```
hat_Flurstuecke             : SET[0:?] OF Flurstueck FOR in_Flur;
```

```
END_ENTITY;
```

```
ENTITY Attribut_Flurstueck;
```



```
--- Attribute :
  Attributname           : STRING(100);
  Attributdatum         : OPTIONAL Datum;
  Attributstatus        : Boolean;
  Attributtext          : OPTIONAL STRING;
--- Relationen :
INVERSE
  von_Flurstueck        : Flurstueck FOR hat_Attribute;
END_ENTITY;

ENTITY Erwerbsflaeche;
--- Attribute :
  Erwerbsart            : Erwerbsart;
  Erwerbszweck         : OPTIONAL Erwerbszweck;
  GE_Massnahme         : OPTIONAL GE_Massnahme;
  Bedarfsflaeche       : OPTIONAL Quadratmeter;
  GE_Plannummer        : OPTIONAL STRING(12);
  laufende_Nummer_Flurstueck : OPTIONAL STRING(4);
  laufende_Nummer_Teilflaeche : OPTIONAL STRING(2);
  Bemerkung            : OPTIONAL STRING(50);
--- Relationen :
  hat_Nutzungsart      : OPTIONAL Nutzungsart;
INVERSE
  in_Flurstueck       : Flurstueck FOR hat_Erwerbsflaechen;
END_ENTITY;

ENTITY Erwerbsart;
  (* KEY_NAME Erwerbsartschluessel *)
  (* KEY_TYP CHAR(1) *)
  Erwerbsartschluessel : STRING(1);
  Erwerbsart           : STRING(50);
  GEV_Ausgabespalte    : GEV_Ausgabespalte;
UNIQUE
  Kennung_eindeutig    : Erwerbsartschluessel;
END_ENTITY;

(* SQL :

INSERT INTO Erwerbsart VALUES ('?', 'unbekannt', 1)
INSERT INTO Erwerbsart VALUES ('1', 'zu erwerbende Fläche', 1)
INSERT INTO Erwerbsart VALUES ('2', 'vorübergehend in Anspruch zu nehmende Fläche', 2)
INSERT INTO Erwerbsart VALUES ('3', 'dauernd zu belastende Fläche', 3)
INSERT INTO Erwerbsart VALUES ('4', 'Restflächenerwerb', 1)
INSERT INTO Erwerbsart VALUES ('5', 'rückständiger Grunderwerb', 1)

END_SQL
*)

ENTITY GEV_Ausgabespalte;
  (* KEY_NAME GEV_Ausgabespalte *)
  GEV_Ausgabespalte : INTEGER;
  Beschreibung      : STRING;
UNIQUE
  Kennung_eindeutig : GEV_Ausgabespalte;
END_ENTITY;

(* SQL :

INSERT INTO GEV_Ausgabespalte VALUES (1, 'zu erwerbende Fläche')
INSERT INTO GEV_Ausgabespalte VALUES (2, 'vorübergehend in Anspruch zu nehmende Fläche')
INSERT INTO GEV_Ausgabespalte VALUES (3, 'dauernd zu belastende Fläche')

END_SQL
*)

ENTITY Erwerbszweck;
  (* KEY_NAME Erwerbszweckschluessel *)
```



```
(* KEY_TYP CHAR(1) *)
Erwerbszweckschlüssel      : STRING(1);
Erwerbsart                 : STRING(50);
UNIQUE
Kennung_eindeutig         : Erwerbszweckschlüssel;
END_ENTITY;

(* SQL :

INSERT INTO Erwerbszweck VALUES ('?', 'unbekannt')
INSERT INTO Erwerbszweck VALUES ('A', 'für Baulastträger der Baumaßnahme Straße')
INSERT INTO Erwerbszweck VALUES ('B', 'für Nebenanlagen und Nebenbetriebe')
INSERT INTO Erwerbszweck VALUES ('C', 'für Dritte')
INSERT INTO Erwerbszweck VALUES ('D', 'für Baulastträger der Baumaßnahme LBP')
INSERT INTO Erwerbszweck VALUES ('X', 'Flächen-/Planungsänderung')
INSERT INTO Erwerbszweck VALUES ('Y', 'Fläche entfällt')

END_SQL
*)

ENTITY Nutzungsart;
--- Attribute :
Nutzungsartschlüssel      : INTEGER;
Kurztext                  : OPTIONAL STRING(4);
Bezeichnung               : STRING(50);
--- Relationen :
INVERSE
von_Nutzungsartteilflaeche : SET[0:?] OF Nutzungsartteilflaeche
                           FOR hat_Nutzungsart;
von_Erwerbsflaeche        : SET[0:?] OF Erwerbsflaeche
                           FOR hat_Nutzungsart;
von_Paechter_Mieter       : SET[0:?] OF Paechter_Mieter
                           FOR hat_Nutzungsart;
END_ENTITY;

ENTITY GE_Massnahme;
(* KEY_NAME GE_Massnahme *)
(* KEY_TYP CHAR(10) *)
GE_Massnahme              : STRING(10);
Massnahmetext             : STRING(200);
UNIQUE
Kennung_eindeutig         : GE_Massnahme;
END_ENTITY;

ENTITY Nutzungsartteilflaeche;
--- Attribute :
Teilflaeche               : OPTIONAL Quadratmeter;
Bodenwert                 : OPTIONAL Euro;
--- Relationen :
hat_Nutzungsart           : Nutzungsart;
INVERSE
von_Flurstueck            : Flurstueck FOR hat_Nutzungsartteilflaechen;
END_ENTITY;

ENTITY Belastung;
--- Attribute :
Abteilung                 : STRING(1);
laufende_Nummer           : OPTIONAL STRING(4);
Belastungstext            : OPTIONAL STRING(200);
Erlaeuterungen           : OPTIONAL STRING;
Eintragung                : OPTIONAL Datum;
Abloesung                 : OPTIONAL Datum;
Bemerkung                 : OPTIONAL STRING(200);
--- Relationen :
von_Belastungsberechtigten : OPTIONAL SET[1:?] OF Belastungsberechtigter;
INVERSE
von_Flurstueck            : SET[0:1] OF Flurstueck FOR hat_Belastungen;
in_Grundbuch              : SET[0:1] OF Grundbuch FOR hat_Belastungen;
```



```
END_ENTITY;

ENTITY Belastungsberechtigter;
--- Attribute :
    Bemerkung                : OPTIONAL STRING(200);
--- Relationen :
    ist_Person                : OPTIONAL Person;
INVERSE
    zu_Belastung             : SET[0:1] OF Belastung
                              FOR von_Belastungsberechtigten;
END_ENTITY;

ENTITY Person_Flurstueck;
--- Attribute :
    Verhaeltnis              : OPTIONAL Verhaeltnis;
    Bemerkung                : OPTIONAL STRING(200);
--- Relationen :
    Flurstueck               : Flurstueck;
    Person                   : Person;
END_ENTITY;

ENTITY Verhaeltnis;
    (* KEY_NAME Kennung *)
    (* KEY_TYP CHAR(1) *)
    Kennung                  : STRING(1);
    Verhaeltnis              : STRING;
UNIQUE
    Kennung_eindeutig        : Kennung;
END_ENTITY;

(* SQL :

INSERT INTO Verhaeltnis VALUES ('V','Verfügungsberechtigter')
INSERT INTO Verhaeltnis VALUES ('A','Alteigentümer')

    END_SQL
*)

ENTITY Flurstueck_Grundbuch;
--- Attribute :
    GB_Anteil_Zaehler        : INTEGER;
    GB_Anteil_Nenner        : INTEGER;
    Eigentumsart             : OPTIONAL Eigentumsart;
    Bestandsnummer          : OPTIONAL STRING(8);
    Bemerkung                : OPTIONAL STRING(50);
--- Relationen :
    Flurstueck               : OPTIONAL Flurstueck;
    Grundbuch               : OPTIONAL Grundbuch;
END_ENTITY;

ENTITY Eigentumsart;
    (* KEY_NAME Kennung *)
    (* KEY_TYP CHAR(1) *)
    Kennung                  : STRING(1);
    Eigentumsart             : STRING;
UNIQUE
    Kennung_eindeutig        : Kennung;
END_ENTITY;

(* SQL :

INSERT INTO Eigentumsart VALUES ('W','Wohneigentum')
INSERT INTO Eigentumsart VALUES ('E','Erbbaurecht')

    END_SQL
*)

ENTITY Grundbuch;
--- Attribute :
```



```
Band                : OPTIONAL STRING(3);
Blatt               : STRING(6);
Bemerkung          : OPTIONAL STRING(100);
--- Relationen :
zu_Gemarkung       : Gemarkung;
hat_Belastungen    : OPTIONAL SET[1:?] OF Belastung;
INVERSE
in_Flurstueck_Grundbuch : SET[0:?] OF Flurstueck_Grundbuch
                    FOR Grundbuch;
hat_Paechter_Mieter : SET[0:?] OF Paechter_Mieter
                    FOR zu_Grundbuch;
hat_Eigentuemer    : SET[0:?] OF Eigentuemer_Grundbuch
                    FOR von_Grundbuch;
END_ENTITY;

ENTITY Gemarkung;
--- Attribute :
Schluessel         : STRING(4);
--- Relationen :
in_Bundesland     : Bundesland;
in_Kreis          : OPTIONAL SET[1:?] OF Kreis_kreisfreie_Stadt;
in_Gemeinde       : OPTIONAL SET[1:?] OF Gemeindebezirk;
INVERSE
hat_Flurstuecke   : SET[0:?] OF Flurstueck FOR in_Gemarkung;
hat_Grundbuch    : SET[0:?] OF Grundbuch FOR zu_Gemarkung;
END_ENTITY;

ENTITY Paechter_Mieter;
--- Attribute :
Pachtbeginn       : OPTIONAL Datum;
Pachtende         : OPTIONAL Datum;
Pachtzins         : OPTIONAL Euro;
Pachtflaeche      : OPTIONAL Quadratmeter;
Vertragsart       : OPTIONAL Vertragsart;
Bemerkung        : OPTIONAL STRING(100);
--- Relationen :
von_Flurstueck    : OPTIONAL Flurstueck;
zu_Grundbuch     : OPTIONAL Grundbuch;
ist_Person        : OPTIONAL Person;
hat_Nutzungsart   : OPTIONAL Nutzungsart;
END_ENTITY;

ENTITY Vertragsart;
(* KEY_NAME Kennung *)
(* KEY_TYP CHAR(1) *)
Kennung           : STRING(1);
Vertragsart       : STRING;
UNIQUE
Kennung_eindeutig : Kennung;
END_ENTITY;

(* SQL :

INSERT INTO Vertragsart VALUES ('s','schriftlich')
INSERT INTO Vertragsart VALUES ('m','mündlich')

END_SQL
*)

ENTITY Eigentuemer_Grundbuch;
--- Attribute :
Eigentumsverhaeltnis : OPTIONAL Eigentumsverhaeltnis;
ET_Anteil_Zaehler     : INTEGER;
ET_Anteil_Nenner     : INTEGER;
Bemerkung            : OPTIONAL STRING(50);
--- Relationen :
von_Grundbuch       : Grundbuch;
ist_Person           : Person;
END_ENTITY;
```



```
ENTITY Eigentumsverhaeltnis;
  (* KEY_NAME Kennung *)
  (* KEY_TYP CHAR(1) *)
  Kennung                : STRING(1);
  Eigentumsverhaeltnis  : STRING;
UNIQUE
  Kennung_eindeutig      : Kennung;
END_ENTITY;

(* SQL :

INSERT INTO Eigentumsverhaeltnis VALUES ('A','Alleineigentümer')
INSERT INTO Eigentumsverhaeltnis VALUES ('M','Miteigentümer')
INSERT INTO Eigentumsverhaeltnis VALUES ('U','Ungeteilte Gemeinschaft')
INSERT INTO Eigentumsverhaeltnis VALUES ('G','Geteilte Gemeinschaft')

  END_SQL
*)

ENTITY Person;
--- Attribute :
  Personenklasse         : OPTIONAL Personenklasse;
  Titel                  : OPTIONAL STRING(32);
  Name                    : STRING(64);
  Vorname                 : OPTIONAL STRING(48);
  Strasse                 : OPTIONAL STRING(48);
  Plz                     : OPTIONAL STRING(10);
  Ort                     : OPTIONAL STRING(48);
  Land                    : OPTIONAL Land;
  Geburtsdatum           : OPTIONAL Datum;
  Geburtsname            : OPTIONAL STRING(64);
  Anrede                  : OPTIONAL STRING(50);
  Telefon                 : OPTIONAL STRING(20);
  Telefax                 : OPTIONAL STRING(20);
  Bemerkung               : OPTIONAL STRING(100);
--- Relationen :
INVERSE
  zu_Flurstueck          : SET[0:?] OF Person_Flurstueck
                          FOR Person;
  ist_Eigentuemer_Grundbuch : SET[0:?] OF Eigentuemer_Grundbuch
                          FOR ist_Person;
  ist_Paechter_Mieter     : SET[0:?] OF Paechter_Mieter
                          FOR ist_Person;
  ist_Belastungsberechtigter : SET[0:?] OF Belastungsberechtigter
                          FOR ist_Person;
END_ENTITY;

ENTITY Land;
  (* KEY_NAME Kennung *)
  (* KEY_TYP CHAR(1) *)
  Kennung                : STRING(1);
  Land                    : STRING;
UNIQUE
  Kennung_eindeutig      : Kennung;
END_ENTITY;

(* SQL :

INSERT INTO Land VALUES ('AFG','Afghanistan')
INSERT INTO Land VALUES ('AL','Albanien')
INSERT INTO Land VALUES ('AND','Andora')
INSERT INTO Land VALUES ('RA','Argentinien')
INSERT INTO Land VALUES ('AZ','Aserbaidtschan')
INSERT INTO Land VALUES ('AUS','Australien')
INSERT INTO Land VALUES ('BS','Bahamas')
INSERT INTO Land VALUES ('BRN','Bahrain')
INSERT INTO Land VALUES ('BD','Bangladesch')
INSERT INTO Land VALUES ('BDS','Barbados')
```



```
INSERT INTO Land VALUES ('BY','Belarus')
INSERT INTO Land VALUES ('B','Belgien')
INSERT INTO Land VALUES ('BH','Belize')
INSERT INTO Land VALUES ('BOL','Bolivien')
INSERT INTO Land VALUES ('BIH','Bosnien-Herzegowina')
INSERT INTO Land VALUES ('BR','Brasilien')
INSERT INTO Land VALUES ('BRU','Brunei Darussalam')
INSERT INTO Land VALUES ('BG','Bulgarien')
INSERT INTO Land VALUES ('BF','Burkina Faso')
INSERT INTO Land VALUES ('CA','Canada')
INSERT INTO Land VALUES ('DK','Dänemark')
INSERT INTO Land VALUES ('D','Deutschland')
INSERT INTO Land VALUES ('FIN','Finnland')
INSERT INTO Land VALUES ('F','Frankreich')
INSERT INTO Land VALUES ('GR','Griechenland')
INSERT INTO Land VALUES ('GB','Großbritannien')
INSERT INTO Land VALUES ('IRL','Irland')
INSERT INTO Land VALUES ('IS','Island')
INSERT INTO Land VALUES ('IL','Israel')
INSERT INTO Land VALUES ('I','Italien')
INSERT INTO Land VALUES ('J','Japan')
INSERT INTO Land VALUES ('CDN','Kanada')
INSERT INTO Land VALUES ('C','Kuba')
INSERT INTO Land VALUES ('LT','Litauen')
INSERT INTO Land VALUES ('L','Luxenburg')
INSERT INTO Land VALUES ('M','Malta')
INSERT INTO Land VALUES ('MA','Marokko')
INSERT INTO Land VALUES ('MEX','Mexiko')
INSERT INTO Land VALUES ('MC','Monaco')
INSERT INTO Land VALUES ('N','Norwegen')
INSERT INTO Land VALUES ('A','Österreich')
INSERT INTO Land VALUES ('PL','Polen')
INSERT INTO Land VALUES ('P','Portugal')
INSERT INTO Land VALUES ('S','Schweden')
INSERT INTO Land VALUES ('CH','Schweiz')
INSERT INTO Land VALUES ('SLO','Slowenien')
INSERT INTO Land VALUES ('E','Spanien')
INSERT INTO Land VALUES ('ZA','Südafrika')
INSERT INTO Land VALUES ('TR','Türkei')
INSERT INTO Land VALUES ('H','Ungarn')
INSERT INTO Land VALUES ('USA','Vereinigte Staaten')
```

END\_SQL

\*)

```
ENTITY Personenklasse;
```

```
(* KEY_NAME Kennung *)
```

```
(* KEY_TYP CHAR(1) *)
```

```
Kennung : STRING(1);
```

```
Personenklasse : STRING;
```

```
UNIQUE
```

```
Kennung_eindeutig : Kennung;
```

```
END_ENTITY;
```

```
(* SQL :
```

```
INSERT INTO Personenklasse VALUES ('?', 'unbekannt')
```

```
INSERT INTO Personenklasse VALUES ('G', 'Gemeindeverwaltung')
```

```
INSERT INTO Personenklasse VALUES ('J', 'juristische Person')
```

```
INSERT INTO Personenklasse VALUES ('L', 'Landwirtschaftsamt')
```

```
INSERT INTO Personenklasse VALUES ('N', 'natürliche Person')
```

```
INSERT INTO Personenklasse VALUES ('Ö', 'öffentlicher Bedarfsträger')
```

```
INSERT INTO Personenklasse VALUES ('V', 'verstorben')
```

END\_SQL

\*)

```
END_SCHEMA; -- Grunderwerb
```



SCHEMA Allgemeine\_Objekte;

(\*  
Auszug zur Verwendung im Schema Grunderwerb  
\*)

TYPE Datum = STRING(10) FIXED;  
WHERE  
    Datums\_Format                          : SELF LIKE '##.##.####';  
    (\* BEDINGUNG SELF LIKE '\_\_.\_\_.\_\_\_\_' \*)  
END\_TYPE;

TYPE Groesse = REAL;  
WHERE  
    Groesse\_nicht\_negativ                : SELF >= 0;  
END\_TYPE;

TYPE Kilometer = Groesse;  
END\_TYPE;

TYPE Quadratmeter = Groesse;  
END\_TYPE;

TYPE Euro = Groesse;  
END\_TYPE;

END\_SCHEMA; -- Allgemeine\_Objekte

SCHEMA Administration;

(\*  
Auszug zur Verwendung im Schema Grunderwerb  
\*)

REFERENCE FROM Grunderwerb (Gemarkung);

ENTITY ASB\_Bezeichnung  
ABSTRACT SUPERTYPE OF (Verwaltungsbezirk);  
    (\* VERERBEN DOWN \*)  
--- Attribute :  
    Bezeichnung\_Lang                      : OPTIONAL STRING(29);  
    Bezeichnung\_Mittel                    : OPTIONAL STRING(19);  
    Bezeichnung\_Kurz                      : OPTIONAL STRING(11);  
--- Relationen :  
END\_ENTITY;

ENTITY Verwaltungsbezirk                  -- ASB 97, 4.2 & 9.2  
ABSTRACT SUPERTYPE OF (ONEOF(Bundesland,Regierungsbezirk,  
                                  Kreis\_kreisfreie\_Stadt,Gemeindebezirk,Ortsteil))  
SUBTYPE OF (ASB\_Bezeichnung);  
    (\* ARTEN J \*)  
--- Attribute :  
--- Relationen :  
END\_ENTITY;

ENTITY Bundesland  
SUBTYPE OF (Verwaltungsbezirk);  
--- Attribute :  
    Kennung\_Bundesland                   : STRING(2) FIXED;  
--- Relationen :  
INVERSE  
    enthaelt\_Regierungsbezirk            : SET [0:?] OF Regierungsbezirk  
  FOR ist\_in\_Bundesland;  
    enthaelt\_Gemarkung                   : SET[0:?] OF Gemarkung FOR in\_Bundesland;  
END\_ENTITY;

ENTITY Regierungsbezirk



```
SUBTYPE OF (Verwaltungsbezirk);
--- Attribute :
    Kennung_Regierungsbezirk      : STRING(3) FIXED;
--- Relationen :
    ist_in_Bundesland             : SET [1:?] OF Bundesland; (* D *)
INVERSE
    enthaelt_Kreis_kreisfr_Stadt  : SET [0:?] OF Kreis_kreisfreie_Stadt
                                   FOR ist_in_Regierungsbezirk;
END_ENTITY;

ENTITY Kreis_kreisfreie_Stadt
SUBTYPE OF (Verwaltungsbezirk);
--- Attribute :
    Kennung_Kreis                 : STRING(5) FIXED;
    Kreisart                       : OPTIONAL Kreisart;
--- Relationen :
    ist_in_Regierungsbezirk       : SET [1:?] OF Regierungsbezirk; (* D *)
INVERSE
    enthaelt_Gemeindebezirk      : SET [0:?] OF Gemeindebezirk
                                   FOR ist_in_Kreis_kreisfreie_Stadt;
    enthaelt_Gemarkung           : SET[0:?] OF Gemarkung FOR in_Kreis;
END_ENTITY;

ENTITY Kreisart;
    (* KEY_NAME Kennung *)
    (* KEY_TYP CHAR(1) *)
    Kennung                       : STRING(1);
    Langtext                       : STRING;
UNIQUE
    Kennung_eindeutig             : Kennung;
END_ENTITY;

(* SQL :

INSERT INTO Kreisart VALUES ('S','Stadtkreis')
INSERT INTO Kreisart VALUES ('L','Landkreis')

    END_SQL
*)

ENTITY Gemeindebezirk
SUBTYPE OF (Verwaltungsbezirk);
--- Attribute :
    Kennung_Gemeinde               : STRING(8) FIXED;
--- Relationen :
    ist_in_Kreis_kreisfreie_Stadt : SET [1:?] OF Kreis_kreisfreie_Stadt; (* D *)
INVERSE
    enthaelt_Ortsteil             : SET [0:?] OF Ortsteil
                                   FOR ist_in_Gemeindebezirk;
    enthaelt_Gemarkung           : SET[0:?] OF Gemarkung FOR in_Gemeinde;
END_ENTITY;

ENTITY Ortsteil
SUBTYPE OF (Verwaltungsbezirk);
--- Attribute :
    Kennung_Ortsteil               : STRING(10) FIXED;
--- Relationen :
    ist_in_Gemeindebezirk         : SET [1:?] OF Gemeindebezirk; (* D *)
END_ENTITY;

END_SCHEMA; -- Administration
```



## 5 SQL-Schema

```
-- grunderwerb.SQL
```

```
CREATE SCHEMA Grunderwerb
```

```
CREATE TABLE Flurstueck (  
  Flurstueck_ID  
    INTEGER NOT NULL PRIMARY KEY,  
  Flurstueck_Zaehler  
    VARCHAR(5) NOT NULL,  
  Flurstueck_Nenner  
    VARCHAR(3),  
  Flaechen  
    REAL, -- Quadratmeter  
  CHECK ( Flaechen >= 0 ),  
  Lage  
    VARCHAR(100),  
  Baukilometer  
    REAL, -- Kilometer  
  CHECK ( Baukilometer >= 0 ),  
  Liegenschaftsbuch  
    VARCHAR(5),  
  Flurkarte  
    VARCHAR(50),  
  Bemerkung  
    VARCHAR(100),  
  in_Gemarkung  
    INTEGER NOT NULL REFERENCES Gemarkung,  
  in_Flur  
    INTEGER REFERENCES Flur  
)  
  
CREATE TABLE Flur (  
  Flur_ID  
    INTEGER NOT NULL PRIMARY KEY,  
  Kennung  
    VARCHAR(3) NOT NULL  
)  
  
CREATE TABLE Attribut_Flurstueck (  
  Attribut_Flurstueck_ID  
    INTEGER NOT NULL PRIMARY KEY,  
  Attributname  
    VARCHAR(100) NOT NULL,  
  Attributdatum  
    CHAR(10), -- Datum  
  CHECK ( Attributdatum LIKE '__.__.____' ),  
  Attributstatus  
    INTEGER NOT NULL REFERENCES Logical_Types.Boolean,  
  Attributtext  
    VARCHAR(255),  
  von_Flurstueck  
    INTEGER NOT NULL REFERENCES Flurstueck  
)  
  
CREATE TABLE Erwerbsflaeche (  
  Erwerbsflaeche_ID  
    INTEGER NOT NULL PRIMARY KEY,  
  Erwerbsart  
    CHAR(1) NOT NULL REFERENCES Erwerbsart,  
  Erwerbszweck
```



```
        CHAR(1) REFERENCES Erwerbzsweck,
GE_Massnahme
        CHAR(10) REFERENCES GE_Massnahme,
Bedarfsflaeche
        REAL, -- Quadratmeter
CHECK ( Bedarfsflaeche >= 0 ),
GE_Plannummer
        VARCHAR(12),
laufende_Nummer_Flurstueck
        VARCHAR(4),
laufende_Nummer_Teilflaeche
        VARCHAR(2),
Bemerkung
        VARCHAR(50),
hat_Nutzungsart
        INTEGER REFERENCES Nutzungsart,
in_Flurstueck
        INTEGER NOT NULL REFERENCES Flurstueck
)

CREATE TABLE Erwerbsart (
    Erwerbsartschluessel
        CHAR(1) NOT NULL PRIMARY KEY,
    Erwerbsart
        VARCHAR(50) NOT NULL,
    GEV_Ausgabespalte
        INTEGER NOT NULL REFERENCES GEV_Ausgabespalte
)

INSERT INTO Erwerbsart VALUES ('?', 'unbekannt', 1)
INSERT INTO Erwerbsart VALUES ('1', 'zu erwerbende Fläche', 1)
INSERT INTO Erwerbsart VALUES ('2', 'vorübergehend in Anspruch zu nehmende Fläche', 2)
INSERT INTO Erwerbsart VALUES ('3', 'dauernd zu belastende Fläche', 3)
INSERT INTO Erwerbsart VALUES ('4', 'Restflächenerwerb', 1)
INSERT INTO Erwerbsart VALUES ('5', 'rückständiger Grunderwerb', 1)

CREATE TABLE GEV_Ausgabespalte (
    GEV_Ausgabespalte
        INTEGER NOT NULL PRIMARY KEY,
    Beschreibung
        VARCHAR(255) NOT NULL
)

INSERT INTO GEV_Ausgabespalte VALUES (1, 'zu erwerbende Fläche')
INSERT INTO GEV_Ausgabespalte VALUES (2, 'vorübergehend in Anspruch zu nehmende Fläche')
INSERT INTO GEV_Ausgabespalte VALUES (3, 'dauernd zu belastende Fläche')

CREATE TABLE Erwerbzsweck (
    Erwerbzsweckschluessel
        CHAR(1) NOT NULL PRIMARY KEY,
    Erwerbsart
        VARCHAR(50) NOT NULL
)

INSERT INTO Erwerbzsweck VALUES ('?', 'unbekannt')
INSERT INTO Erwerbzsweck VALUES ('A', 'für Baulastträger der Baumaßnahme Straße')
INSERT INTO Erwerbzsweck VALUES ('B', 'für Nebenanlagen und Nebenbetriebe')
INSERT INTO Erwerbzsweck VALUES ('C', 'für Dritte')
INSERT INTO Erwerbzsweck VALUES ('D', 'für Baulastträger der Baumaßnahme LBP')
INSERT INTO Erwerbzsweck VALUES ('X', 'Flächen-/Planungsänderung')
INSERT INTO Erwerbzsweck VALUES ('Y', 'Fläche entfällt')

CREATE TABLE Nutzungsart (
    Nutzungsart_ID
        INTEGER NOT NULL PRIMARY KEY,
    Nutzungsartschluessel
```



```
        INTEGER NOT NULL,  
        Kurztext  
        VARCHAR(4),  
        Bezeichnung  
        VARCHAR(50) NOT NULL  
    )  
  
CREATE TABLE GE_Massnahme (  
    GE_Massnahme  
        CHAR(10) NOT NULL PRIMARY KEY,  
    Massnahmetext  
        VARCHAR(200) NOT NULL  
)  
  
CREATE TABLE Nutzungsartteilflaeche (  
    Nutzungsartteilflaeche_ID  
        INTEGER NOT NULL PRIMARY KEY,  
    Teilflaeche  
        REAL, -- Quadratmeter  
    CHECK ( Teilflaeche>=0 ),  
    Bodenwert  
        REAL, -- Euro  
    CHECK ( Bodenwert>=0 ),  
    hat_Nutzungsart  
        INTEGER NOT NULL REFERENCES Nutzungsart,  
    von_Flurstueck  
        INTEGER NOT NULL REFERENCES Flurstueck  
)  
  
CREATE TABLE Belastung (  
    Belastung_ID  
        INTEGER NOT NULL PRIMARY KEY,  
    Abteilung  
        CHAR(1) NOT NULL,  
    laufende_Nummer  
        VARCHAR(4),  
    Belastungstext  
        VARCHAR(200),  
    Erlaeuterungen  
        VARCHAR(255),  
    Eintragung  
        CHAR(10), -- Datum  
    CHECK ( Eintragung LIKE '___..____' ),  
    Abloesung  
        CHAR(10), -- Datum  
    CHECK ( Abloesung LIKE '___..____' ),  
    Bemerkung  
        VARCHAR(200),  
    von_Flurstueck  
        INTEGER REFERENCES Flurstueck,  
    in_Grundbuch  
        INTEGER REFERENCES Grundbuch  
)  
  
CREATE TABLE Belastungsberechtigter (  
    Belastungsberechtigter_ID  
        INTEGER NOT NULL PRIMARY KEY,  
    Bemerkung  
        VARCHAR(200),  
    ist_Person  
        INTEGER REFERENCES Person,  
    zu_Belastung  
        INTEGER REFERENCES Belastung  
)  
  
CREATE TABLE Person_Flurstueck (  
    Person_Flurstueck_ID  
        INTEGER NOT NULL PRIMARY KEY,  
    Verhaeltnis
```



```
        CHAR(1) REFERENCES Verhaeltnis,
Bemerkung
        VARCHAR(200),
Flurstueck
        INTEGER NOT NULL REFERENCES Flurstueck,
Person
        INTEGER NOT NULL REFERENCES Person
)

CREATE TABLE Verhaeltnis (
    Kennung
        CHAR(1) NOT NULL PRIMARY KEY,
    Verhaeltnis
        VARCHAR(255) NOT NULL
)

INSERT INTO Verhaeltnis VALUES ('V','Verfügungsberechtigter')
INSERT INTO Verhaeltnis VALUES ('A','Alteigentümer')

CREATE TABLE Flurstueck_Grundbuch (
    Flurstueck_Grundbuch_ID
        INTEGER NOT NULL PRIMARY KEY,
    GB_Anteil_Zaehler
        INTEGER NOT NULL,
    GB_Anteil_Nenner
        INTEGER NOT NULL,
    Eigentumsart
        CHAR(1) REFERENCES Eigentumsart,
    Bestandsnummer
        VARCHAR(8),
    Bemerkung
        VARCHAR(50),
    Flurstueck
        INTEGER REFERENCES Flurstueck,
    Grundbuch
        INTEGER REFERENCES Grundbuch
)

CREATE TABLE Eigentumsart (
    Kennung
        CHAR(1) NOT NULL PRIMARY KEY,
    Eigentumsart
        VARCHAR(255) NOT NULL
)

INSERT INTO Eigentumsart VALUES ('W','Wohneigentum')
INSERT INTO Eigentumsart VALUES ('E','Erbbaurecht')

CREATE TABLE Grundbuch (
    Grundbuch_ID
        INTEGER NOT NULL PRIMARY KEY,
    Band
        VARCHAR(3),
    Blatt
        VARCHAR(6) NOT NULL,
    Bemerkung
        VARCHAR(100),
    zu_Gemarkung
        INTEGER NOT NULL REFERENCES Gemarkung
)

CREATE TABLE Gemarkung (
    Gemarkung_ID
        INTEGER NOT NULL PRIMARY KEY,
    Schluessel
        VARCHAR(4) NOT NULL,
    in_Bundesland
        INTEGER NOT NULL REFERENCES Administration.Bundesland
)
```



```
CREATE TABLE Gemarkung__in_Kreis (
  Gemarkung
    INTEGER NOT NULL REFERENCES Gemarkung,
  Kreis_kreisfreie_Stadt
    INTEGER NOT NULL REFERENCES Administration.Kreis_kreisfreie_Stadt
)

CREATE TABLE Gemarkung__in_Gemeinde (
  Gemarkung
    INTEGER NOT NULL REFERENCES Gemarkung,
  Gemeindebezirk
    INTEGER NOT NULL REFERENCES Administration.Gemeindebezirk
)

CREATE TABLE Paechter_Mieter (
  Paechter_Mieter_ID
    INTEGER NOT NULL PRIMARY KEY,
  Pachtbeginn
    CHAR(10), -- Datum
  CHECK ( Pachtbeginn LIKE '__.__.____' ),
  Pachtende
    CHAR(10), -- Datum
  CHECK ( Pachtende LIKE '__.__.____' ),
  Pachtzins
    REAL, -- Euro
  CHECK ( Pachtzins>=0 ),
  Pachtflaeche
    REAL, -- Quadratmeter
  CHECK ( Pachtflaeche>=0 ),
  Vertragsart
    CHAR(1) REFERENCES Vertragsart,
  Bemerkung
    VARCHAR(100),
  von_Flurstueck
    INTEGER REFERENCES Flurstueck,
  zu_Grundbuch
    INTEGER REFERENCES Grundbuch,
  ist_Person
    INTEGER REFERENCES Person,
  hat_Nutzungsart
    INTEGER REFERENCES Nutzungsart
)

CREATE TABLE Vertragsart (
  Kennung
    CHAR(1) NOT NULL PRIMARY KEY,
  Vertragsart
    VARCHAR(255) NOT NULL
)

INSERT INTO Vertragsart VALUES ('s','schriftlich')
INSERT INTO Vertragsart VALUES ('m','mündlich')

CREATE TABLE Eigentuemer_Grundbuch (
  Eigentuemer_Grundbuch_ID
    INTEGER NOT NULL PRIMARY KEY,
  Eigentumsverhaeltnis
    CHAR(1) REFERENCES Eigentumsverhaeltnis,
  ET_Anteil_Zaehler
    INTEGER NOT NULL,
  ET_Anteil_Nenner
    INTEGER NOT NULL,
  Bemerkung
    VARCHAR(50),
  von_Grundbuch
    INTEGER NOT NULL REFERENCES Grundbuch,
  ist_Person
    INTEGER NOT NULL REFERENCES Person
)
```



```
)

CREATE TABLE Eigentumsverhaeltnis (
  Kennung
    CHAR(1) NOT NULL PRIMARY KEY,
  Eigentumsverhaeltnis
    VARCHAR(255) NOT NULL
)

INSERT INTO Eigentumsverhaeltnis VALUES ('A','Alleineigentümer')
INSERT INTO Eigentumsverhaeltnis VALUES ('M','Miteigentümer')
INSERT INTO Eigentumsverhaeltnis VALUES ('U','Ungeteilte Gemeinschaft')
INSERT INTO Eigentumsverhaeltnis VALUES ('G','Geteilte Gemeinschaft')

CREATE TABLE Person (
  Person_ID
    INTEGER NOT NULL PRIMARY KEY,
  Personenklasse
    CHAR(1) REFERENCES Personenklasse,
  Titel
    VARCHAR(32),
  Name
    VARCHAR(64) NOT NULL,
  Vorname
    VARCHAR(48),
  Strasse
    VARCHAR(48),
  Plz
    VARCHAR(10),
  Ort
    VARCHAR(48),
  Land
    CHAR(1) REFERENCES Land,
  Geburtsdatum
    CHAR(10), -- Datum
  CHECK ( Geburtsdatum LIKE '____.____.____' ),
  Geburtsname
    VARCHAR(64),
  Anrede
    VARCHAR(50),
  Telefon
    VARCHAR(20),
  Telefax
    VARCHAR(20),
  Bemerkung
    VARCHAR(100)
)

CREATE TABLE Land (
  Kennung
    CHAR(1) NOT NULL PRIMARY KEY,
  Land
    VARCHAR(255) NOT NULL
)

INSERT INTO Land VALUES ('AFG','Afghanistan')
INSERT INTO Land VALUES ('AL','Albanien')
INSERT INTO Land VALUES ('AND','Andora')
INSERT INTO Land VALUES ('RA','Argentinien')
INSERT INTO Land VALUES ('AZ','Aserbaidzhan')
INSERT INTO Land VALUES ('AUS','Australien')
INSERT INTO Land VALUES ('BS','Bahamas')
INSERT INTO Land VALUES ('BRN','Bahrain')
INSERT INTO Land VALUES ('BD','Bangladesch')
INSERT INTO Land VALUES ('BDS','Barbados')
INSERT INTO Land VALUES ('BY','Belarus')
INSERT INTO Land VALUES ('B','Belgien')
INSERT INTO Land VALUES ('BH','Belize')
INSERT INTO Land VALUES ('BOL','Bolivien')
```



```
INSERT INTO Land VALUES ('BIH','Bosnien-Herzegowina')
INSERT INTO Land VALUES ('BR','Brasilien')
INSERT INTO Land VALUES ('BRU','Brunei Darussalam')
INSERT INTO Land VALUES ('BG','Bulgarien')
INSERT INTO Land VALUES ('BF','Burkina Faso')
INSERT INTO Land VALUES ('CA','Canada')
INSERT INTO Land VALUES ('DK','Dänemark')
INSERT INTO Land VALUES ('D','Deutschland')
INSERT INTO Land VALUES ('FIN','Finnland')
INSERT INTO Land VALUES ('F','Frankreich')
INSERT INTO Land VALUES ('GR','Griechenland')
INSERT INTO Land VALUES ('GB','Großbritannien')
INSERT INTO Land VALUES ('IRL','Irland')
INSERT INTO Land VALUES ('IS','Island')
INSERT INTO Land VALUES ('IL','Israel')
INSERT INTO Land VALUES ('I','Italien')
INSERT INTO Land VALUES ('J','Japan')
INSERT INTO Land VALUES ('CDN','Kanada')
INSERT INTO Land VALUES ('C','Kuba')
INSERT INTO Land VALUES ('LT','Litauen')
INSERT INTO Land VALUES ('L','Luxemburg')
INSERT INTO Land VALUES ('M','Malta')
INSERT INTO Land VALUES ('MA','Marokko')
INSERT INTO Land VALUES ('MEX','Mexiko')
INSERT INTO Land VALUES ('MC','Monaco')
INSERT INTO Land VALUES ('N','Norwegen')
INSERT INTO Land VALUES ('A','Österreich')
INSERT INTO Land VALUES ('PL','Polen')
INSERT INTO Land VALUES ('P','Portugal')
INSERT INTO Land VALUES ('S','Schweden')
INSERT INTO Land VALUES ('CH','Schweiz')
INSERT INTO Land VALUES ('SLO','Slowenien')
INSERT INTO Land VALUES ('E','Spanien')
INSERT INTO Land VALUES ('ZA','Südafrika')
INSERT INTO Land VALUES ('TR','Türkei')
INSERT INTO Land VALUES ('H','Ungarn')
INSERT INTO Land VALUES ('USA','Vereinigte Staaten')
```

```
CREATE TABLE Personenklasse (
  Kennung
    CHAR(1) NOT NULL PRIMARY KEY,
  Personenklasse
    VARCHAR(255) NOT NULL
)
```

```
INSERT INTO Personenklasse VALUES ('?', 'unbekannt')
INSERT INTO Personenklasse VALUES ('G', 'Gemeindeverwaltung')
INSERT INTO Personenklasse VALUES ('J', 'juristische Person')
INSERT INTO Personenklasse VALUES ('L', 'Landwirtschaftsamt')
INSERT INTO Personenklasse VALUES ('N', 'natürliche Person')
INSERT INTO Personenklasse VALUES ('Ö', 'öffentlicher Bedarfsträger')
INSERT INTO Personenklasse VALUES ('V', 'verstorben')
```



## 6 Anhang: Referenzen anderer Schemata

Im folgenden sind einige Entities und Types zusammengestellt, die aus dem Schema Grunderwerb referenziert werden.

Achtung: Diese Objekte sind teilweise auf die Anforderungen für den Grunderwerb angepasst. Die Zusammenstellung hat nur informativen Charakter und gibt die dargestellten Schemata des OKSTRA nicht notwendig vollständig wieder. Ebenso sind zu erwartende Änderungen in den betroffenen Schemata als Folge der Modellierung des Grunderwerbsverzeichnisses bereits integriert.

```
SCHEMA Allgemeine_Objekte;
```

```
(*
```

```
Auszug zur Verwendung im Schema Grunderwerb
```

```
*)
```

```
TYPE Datum = STRING(10) FIXED;
```

```
WHERE
```

```
  Datums_Format          : SELF LIKE '##.##.####';
```

```
  (* BEDINGUNG SELF LIKE '__.__.____' *)
```

```
END_TYPE;
```

```
TYPE Groesse = REAL;
```

```
WHERE
```

```
  Groesse_nicht_negativ : SELF >= 0;
```

```
END_TYPE;
```

```
TYPE Kilometer = Groesse;
```

```
END_TYPE;
```

```
TYPE Quadratmeter = Groesse;
```

```
END_TYPE;
```

```
TYPE Euro = Groesse;
```

```
END_TYPE;
```

```
END_SCHEMA; -- Allgemeine_Objekte
```

```
SCHEMA Administration;
```

```
(*
```

```
Auszug zur Verwendung im Schema Grunderwerb
```

```
*)
```

```
REFERENCE FROM Grunderwerb (Gemarkung);
```

```
ENTITY ASB_Bezeichnung
```

```
ABSTRACT SUPERTYPE OF (Verwaltungsbezirk);
```

```
  (* VERERBEN DOWN *)
```

```
--- Attribute :
```

```
  Bezeichnung_Lang          : OPTIONAL STRING(29);
```

```
  Bezeichnung_Mittel        : OPTIONAL STRING(19);
```

```
  Bezeichnung_Kurz          : OPTIONAL STRING(11);
```

```
--- Relationen :
```

```
END_ENTITY;
```

```
ENTITY Verwaltungsbezirk          -- ASB 97, 4.2 & 9.2
```

```
ABSTRACT SUPERTYPE OF (ONEOF(Bundesland, Regierungsbezirk,  
                             Kreis_kreisfreie_Stadt, Gemeindebezirk, Ortsteil))
```

```
SUBTYPE OF (ASB_Bezeichnung);
```

```
  (* ARTEN J *)
```



```
--- Attribute :
--- Relationen :
END_ENTITY;

ENTITY Bundesland
SUBTYPE OF (Verwaltungsbezirk);
--- Attribute :
    Kennung_Bundesland      : STRING(2) FIXED;
--- Relationen :
INVERSE
    enthaelt_Regierungsbezirk : SET [0:?] OF Regierungsbezirk
                                FOR ist_in_Bundesland;
    enthaelt_Gemarkung       : SET[0:?] OF Gemarkung FOR in_Bundesland;
END_ENTITY;

ENTITY Regierungsbezirk
SUBTYPE OF (Verwaltungsbezirk);
--- Attribute :
    Kennung_Regierungsbezirk : STRING(3) FIXED;
--- Relationen :
    ist_in_Bundesland        : SET [1:?] OF Bundesland; (* D *)
INVERSE
    enthaelt_Kreis_kreisfr_Stadt : SET [0:?] OF Kreis_kreisfreie_Stadt
                                FOR ist_in_Regierungsbezirk;
END_ENTITY;

ENTITY Kreis_kreisfreie_Stadt
SUBTYPE OF (Verwaltungsbezirk);
--- Attribute :
    Kennung_Kreis           : STRING(5) FIXED;
    Kreisart                : OPTIONAL Kreisart;
--- Relationen :
    ist_in_Regierungsbezirk : SET [1:?] OF Regierungsbezirk; (* D *)
INVERSE
    enthaelt_Gemeindebezirk : SET [0:?] OF Gemeindebezirk
                                FOR ist_in_Kreis_kreisfreie_Stadt;
    enthaelt_Gemarkung      : SET[0:?] OF Gemarkung FOR in_Kreis;
END_ENTITY;

ENTITY Kreisart;
    (* KEY_NAME Kennung *)
    (* KEY_TYP CHAR(1) *)
    Kennung                 : STRING(1);
    Langtext                : STRING;
UNIQUE
    Kennung_eindeutig       : Kennung;
END_ENTITY;

(* SQL :

INSERT INTO Kreisart VALUES ('S','Stadtkreis')
INSERT INTO Kreisart VALUES ('L','Landkreis')

    END_SQL
*)

ENTITY Gemeindebezirk
SUBTYPE OF (Verwaltungsbezirk);
--- Attribute :
    Kennung_Gemeinde        : STRING(8) FIXED;
--- Relationen :
    ist_in_Kreis_kreisfreie_Stadt : SET [1:?] OF Kreis_kreisfreie_Stadt; (* D *)
INVERSE
    enthaelt_Ortsteil       : SET [0:?] OF Ortsteil
                                FOR ist_in_Gemeindebezirk;
    enthaelt_Gemarkung      : SET[0:?] OF Gemarkung FOR in_Gemeinde;
END_ENTITY;

ENTITY Ortsteil
```



```
SUBTYPE OF (Verwaltungsbezirk);  
--- Attribute :  
    Kennung_Ortsteil          : STRING(10) FIXED;  
--- Relationen :  
    ist_in_Gemeindebezirk     : SET [1:?] OF Gemeindebezirk; (* D *)  
END_ENTITY;  
  
END_SCHEMA; -- Administration
```