

Dokumentation zum EDBS-Reader

EDBS_extra

Claus Rinner

5. April 2001

Inhaltsverzeichnis

1	Einleitung	3
2	EDBS- und ATKIS-Format	4
2.1	Einleitung	4
2.2	EDBS-Sätze	5
2.3	ATKIS-Datengruppen	6
2.4	Wiederholungsfaktor	7
2.5	ATKIS-Datenelemente	7
2.6	Beispiel	10
2.7	Umsetzung der Datenelemente in EDBS_extra -Dateien	12
3	Installation von EDBS_extra	14
4	Lesevorgang	15
4.1	Einleitung	15
4.2	Aufruf	15
4.3	Ergebnis	16
5	Daten-Aufbereitung	18
5.1	Zweck	18
5.2	Aufruf	18
5.3	Ergebnis	18
6	Automatisierung	19
6.1	Daten einlesen	19
6.2	Kartenausdehnung	19
6.3	Statistik	19

7	Einlesen in ARC/INFO	20
7.1	Einleitung	20
7.2	Erzeugen der Coverages	20
7.3	Erzeugen der Attributtabellen	21
7.4	Darstellung	22
8	Programmbeschreibung	23
9	Literatur	25

1 Einleitung

Das vorliegende Programm **EDBS_extra** wurde am Institut für Umweltsystemforschung von Claus Rinner, Student der Angewandten Systemwissenschaft an der Universität Osnabrück, im Zeitraum Februar 1994 – Juni 1995 entwickelt.

Es handelt sich um ein ANSI-C-Programm, das — unter dem Betriebssystem UNIX — EDBS-Aufträge im ATKIS- oder ALK-Format abarbeitet und die darin enthaltenen topologischen und Sachinformationen "lesbar" abspeichert. In Verbindung mit den in dieser Dokumentation ebenfalls beschriebenen Batchdateien können die EDBS-Daten in das Geo-Informationssystem ARC/INFO eingelesen werden. Die Konvertierung in die internen Datenstrukturen anderer GIS-Produkte ist mit **EDBS_extra** dann möglich, wenn das GIS eine ASCII-Schnittstelle zum Datenimport aufweist. Da der Nutzungsschwerpunkt des Programms eher im Auslesen von EDBS-Daten als in der darauffolgenden Erzeugung von GIS-Datenstrukturen liegt, spreche ich im weiteren von einem "EDBS-Reader" (statt von einem "Konverter").

Die Idee zu **EDBS_extra** ist während der Planung für das Projekt "Private Trinkwasserbrunnen im Landkreis Osnabrück" entstanden. Die vom Landkreis Osnabrück zur Verfügung gestellten ATKIS-Daten sollen in ein Fachinformationssystem unter ARC/INFO integriert werden, die Mittel der Universität erlaubten jedoch nicht die Anschaffung eines kommerziellen Konverters. Daher wurde die Entwicklung einer Behelfslösung als studentisches Projekt ausgeschrieben.

Ich danke J. Berlekamp für die Projektbetreuung und die Mitarbeit bei der Weiterentwicklung, M. Klein und S. Fuest für nützliche Hinweise, S. Ferber für die Durchsicht des vorliegenden Textes und dem Direktor des Instituts für Umweltsystemforschung, Prof. M. Matthies, für das Thema der Projektarbeit und die Beschaffung der Basisliteratur. Ein herzlicher Dank geht noch an Jan-Oliver Wagner für die Wiederbeschaffung des Source Codes und der Skripte zum Zwecke der Veröffentlichung.

Nach Ablauf der Projektzeit wurde **EDBS_extra** weiterentwickelt. Ich weise ausdrücklich darauf hin, daß diese Software mit den auf Seite 14 aufgeführten Quelldateien und allen weiteren Bestandteilen urheberrechtlich geschützt ist.

Das Programm einschließlich des Quellcodes wird Interessenten unter der GNU Public License zur Verfügung gestellt. Dies bedeutet insbesondere, daß **EDBS_extra** nicht Bestandteil anderer Software werden kann, die nicht GPL-kompatibel ist. Und daß der Autor keinerlei Garantie für die Funktionsfähigkeit und Unschädlichkeit dieser Software gibt. Näheres ist unter <http://www.rinners.de/edbs> nachzulesen.

Zur erfolgreichen Installation und Benutzung des Readers sind gute UNIX- und C-Kenntnisse nützlich, da bis zu einer zufriedenstellenden Konfiguration einige Handarbeit erforderlich sein dürfte. Die in der Anleitung empfohlenen Befehlsfolgen können nur Beispielcharakter haben ... viel Spaß und Erfolg!

2 EDBS- und ATKIS-Format

2.1 Einleitung

Das *Amtliche Topographisch-Kartographische InformationsSystem (ATKIS)* ist ein ca. 1985 von den Landesvermessungsbehörden der BRD begonnenes Projekt, mit dem im Bereich mittlerer und kleiner Maßstäbe eine einheitliche digitale Datenbasis für alle fachbezogenen Planungs- und Forschungsvorhaben aufgebaut wird. Zur Zeit befindet sich ATKIS in der ersten Aufbaustufe des *Digitalen Landschaftsmodells 1:25000 (DLM 25/1)*. Dieser Datenbestand enthält in etwa diejenigen Objekte der Erdoberfläche, die auf einer herkömmlichen TK 25 repräsentiert sind, dies aber mit einer höheren Lagegenauigkeit. Die Daten des DLM werden im Format der *Einheitlichen Datenbank-Schnittstelle (EDBS)* an die Benutzer (Behörden, Forschungseinrichtungen und Unternehmen) abgegeben.

Das vorliegende Kapitel stellt eine Zusammenfassung der EDBS- und ATKIS-Kenntnisse dar, die ich mir im Selbststudium aus der "Dokumentation zum ALK/ATKIS-Datenaustausch" der AG Hannover und später aus der "ATKIS-Gesamtdokumentation" der AdV erarbeitet habe. Diese Dokumentationen beschreiben das konzeptionelle Datenmodell von ATKIS, seine Transformation in das logische Datenmodell der ALK/ATKIS-Datenbank, die für die Übermittlung von ALK/ATKIS-Daten wesentlichen Parameter der EDBS und — im *Objektartenkatalog (ATKIS-OK)* — die semantische Gliederung der Objekte in Objektbereiche und Objektgruppen.

Die Abkürzung *ALK* steht für *Automatisierte Liegenschaftskarte*. Die ALK ergänzt ATKIS für große Maßstabsbereiche im Katasterwesen. In dieser Dokumentation gehe ich in der Regel nur auf das ATKIS-Datenmodell ein, das jedoch in Anlehnung an die ALK konzipiert wurde und große Ähnlichkeiten aufweist. Die wenigen Informationen über die ALK stammen aus dem *ALK-Objektabbildungskatalog (OBAK)*.

2.2 EDBS-Sätze

Eine Zeile einer Datei im Format der EDBS (*EDBS-Satz*) hat im wesentlichen folgenden Aufbau:

Parameter	Länge	Inhalt
Satzanfang	4	"EDBS"
Satzlänge	4	in Bytes (= Zeichen) ab dem 13. Zeichen im Satz, maximal 1988
Anfang SK	4	(Suchkriterium)
Operation OP	4	Verwendung der Daten
Satznummer	6	(interner Wert)
Zugehörigkeit	1	Leerzeichen; bei zusammengehörenden Sätzen 'A', 'F' oder 'E'
Editier- und Quittungsschlüssel	5	".0000"
Infoname IN	8	Dateninhalt
Infoinhalt II	0–1964	die eigentlichen Daten, Inhalt abhängig von OP und IN

Eine Folge von EDBS-Sätzen wird als *EDBS-Auftrag* bezeichnet, wenn sie aus folgenden Satzarten besteht (ein "." steht für ein Leerzeichen):

Satzart	OP	IN	II
Auftragskennsatz	"AKND"	"ULQA0000"	allg. Daten zum Auftrag
weitere EDBS-Sätze	(verschieden)		
Auftragsendesatz	"AEND"	"."	(leer)

Falls im Auftragskennsatz im Inhalt der Information (II) das Datenelement "Datenkennung-ALK" belegt ist, enthalten die "weiteren EDBS-Sätze" ALK-Daten. Ist die "Datenkennung-DLM" belegt, liegen ATKIS-DLM-Daten vor.

Unter die Rubrik "weitere EDBS-Sätze" der obigen Tabelle fallen die folgenden (ALK/ATKIS-spezifischen) Satzarten:

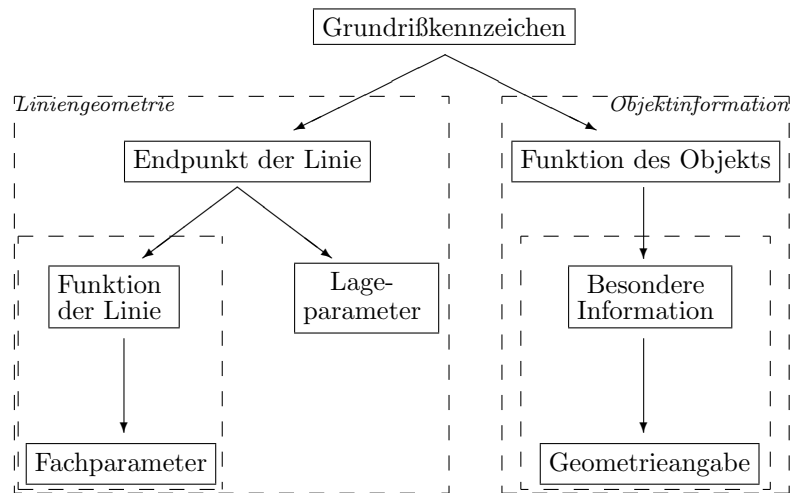
Satzart	OP	IN	II
Gebietskennzeichnung	"BKRT"	"IBENKRT."	linke untere, rechte obere Ecke des Kartenblattes
Auftragstextdaten	"OTEX"	"ULOTEX.."	Text
Grundrißdaten	"BSPE"	"ULOBNN.."	Geometrie
Attributdaten	"BSPE"	"ULTANN.."	Sachdaten

Die folgenden Ausführungen beziehen sich auf den EDBS-Parameter (= Feld) "Inhalt der Information" der beiden letzten Satztypen (vollständige Bezeichnung: "Objektdaten-Grundrißdatei" und "Objektdaten-Attributdatei"), die nahezu 100 Prozent eines EDBS-Auftrags mit ATKIS-Daten ausmachen.

2.3 ATKIS-Datengruppen

Die Grundrißdaten bestehen aus den Datengruppen *Grundrißkennzeichen* (Lage- bzw. Labelpunkt der im Satz repräsentierten Objekte; kommt pro Satz genau einmal vor), *Endpunkt der Linie* (nur bei Linien oder Flächen), *Funktion der Linie* (nur falls Endpunkt vorhanden), *Fachparameter* (Höhenangabe; nur für ATKIS vorgesehen; im DLM 25/1 noch nicht realisiert), *Lageparameter* (nur falls Endpunkt vorhanden: die Interpolations-/Stützpunkte des Linienzuges zwischen Grundrißkennzeichen und Endpunkt), *Funktion des Objekts*, *Besondere Information* (nur falls Funktion des Objekts vorhanden) und *Geometrieangabe* (nur falls Funktion des Objekts vorhanden).

Die hierarchischen Beziehungen zwischen diesen Datengruppen gehen aus dem folgenden Schema hervor ("allgemeine Struktur der Grundrißdatei" gemäß Dokumentation zum ALK/ATKIS-Datenaustausch):

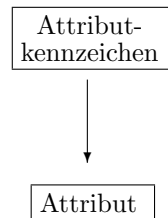


Das "Grundrißkennzeichen" ist in Verbindung mit dem Unterbaum "Liniengeometrie" als Anfangspunkt einer Linie oder eines (Teils eines) Flächenumrings zu interpretieren, beim Abarbeiten des Bereichs "Objektinformation" ist das "Grundrißkennzeichen" als Definitionsgeometrie eines Punktobjekts oder als Labelpunkt einer Linie oder Fläche zu sehen (abhängig vom Objekttyp).

Für jedes Objekt (Punkt, Linie, Fläche, komplexes Objekt) kommt im gesamten EDBS-Auftrag genau ein Grundrißdatensatz mit den objektbezogenen

Datengruppen "Funktion des Objekts", "Besondere Information" und "Geometrieangabe" vor. Für Linien und Flächen gibt es zusätzlich einen oder viele weitere Grundrißdatensätze mit den linienbezogenen Datengruppen "Endpunkt der Linie", "Funktion der Linie", "Fachparameter" und "Lageparameter", die u.a. die Stützpunkte des Linienzugs bzw. Umrings der Fläche enthalten.

Die Attributdatensätze bestehen aus den Datengruppen Attributkennzeichen und Attribut:



2.4 Wiederholungsfaktor

Bis auf das Grundrißkennzeichen können alle Datengruppen in einem EDBS-Satz mehrfach hintereinander vorkommen. Die Anzahl der Wiederholungen wird durch einen Wiederholungsfaktor (WHF, 4 Zeichen, z.B. "0003" für dreimaliges Auftreten) vor der entsprechenden Datengruppe angegeben.

Die Verknüpfung von beispielsweise *einer* Liniengeometrie (Endpunkt und Lageparameter) mit *mehreren* Funktionsangaben (WHF für "Funktion der Linie" > "0001") dient der Vermeidung der mehrfachen (redundanten) Speicherung und Übermittlung der Liniengeometrie.

Aufgrund der hierarchischen Beziehung der Datengruppen gelten die Wiederholungsfaktoren z.T. für mehrere voneinander abhängige Datengruppen (gestrichelt umrahmte Teilbäume des Schemas auf Seite 6).

Die zehn *Datengruppen* sind — gemäß dem ATKIS-Datenmodell — weiter untergliedert in *Datenelemente*. Einige dieser Datenelemente werden im folgenden näher beschrieben.

2.5 ATKIS-Datenelemente

Koordinatenrepräsentation

Die Datenelemente "Objektcoordinate" im Grundrißkennzeichen, Endpunkt der Linie, Lageparameter sowie Geometrieangabe in "Besondere Information" repräsentieren Punkte im Gauß-Krüger-Koordinatensystem. Sie sind 20stellig im Format Numerierungsbezirk (NBZ, 8 Zeichen) + Koordinaten im NBZ (12) angegeben.

Für ATKIS-Daten gilt: Der NBZ ergibt sich aus den ersten beiden Ziffern der x-Koordinate (Rechtswert), gefolgt von den ersten beiden Ziffern der y-Koordinate (Hochwert), der dritten Ziffer von x und einem Leerzeichen sowie der dritten Ziffer von y und einem Leerzeichen. Die relative Koordinate im NBZ ergibt sich aus sechs Ziffern für x und sechs für y, jeweils ab dem vierten Zeichen.

Die Zeichenfolge "34**571.9**.345689**765421**" entspricht demnach dem Punkt (3413456.89, **5797654.21**) im Gauß-Krüger-System. Die Lagegeometrie der ATKIS-Objekte ist also auf cm genau angegeben.

Folie, Objektart und -typ

Die Datengruppen "Funktion der Linie" und "Funktion des Objekts" enthalten je eine dreistellige Foliennummer, die einer Einteilung der Objekte unter fachlichen Gesichtspunkten dient, und eine vierstellige Objektart gemäß ATKIS-Objektartenkatalog. Die erste Ziffer der Objektart gibt einen der *Objektbereiche* "Festpunkte" (Objektart 1...), "Siedlung", "Verkehr", "Vegetation", "Gewässer", "Relief" und "Gebiete" (Objektart 7...) an. Die zweite Ziffer ordnet das Objekt einer *Objektgruppe* innerhalb des Objektbereichs zu.

In der Datengruppe "Funktion des Objekts" findet sich das Datenelement "Objekttyp" mit den möglichen Ausprägungen 'P' (Punkt), 'L' (Linie), 'F' (Fläche), 'R' (Raster) und 'K' (komplexes Objekt). Ein komplexes Objekt wird bei der Digitalisierung der ATKIS-Daten dort gebildet, wo mehrere einfache Objekte auch als Ganzes angesprochen werden können, z.B. Fahrbahn und Straßenkörper als "Straße (komplex)".

Objektnummer und Objektteilnummer

Die Objektnummer besteht aus sieben Ziffern *und Zeichen* und identifiziert ein Objekt eindeutig. Die dreistellige Objektteilnummer ist eindeutig innerhalb eines Objekts.

Die Objektnummer in den objektbezogenen Datengruppen der Grundrißdatensätze und in den Attributdatensätzen identifiziert das Objekt, auf das sich die übrigen Informationen dieser Datengruppen beziehen. In den linienbezogenen Datengruppen der Grundrißdatensätze gibt es zu einer Linie zwei Objektnummernfelder (rechts und links liegendes Objekt), so daß die Grenze zwischen zwei Flächenobjekten (gleicher Objektart) unter Angabe der beiden Flächen-IDs nur einmal vorkommt. Bei Linienobjekten ist nur eins der beiden Objektnummernfelder belegt.

Ein Objekt (eine Objektnummer), z.B. eine Straße, die durch ihren Namen als Einheit gekennzeichnet ist, kann in Objektteile (mit unterschiedlichen Objektteilnummern) gegliedert werden, z.B. dort, wo Straßenkreuzungen auftreten. Ein Linienzug vom Grundrißkennzeichen über die Lageparameter bis zum Endpunkt bildet dann einen solchen Objektteil, dem eigene Attribute und Text der "Besonderen Information" zugeordnet sein können.

Besondere Information - Text und Geometrie

Das Datenelement "Text" (33 Zeichen) in "Besondere Information" ist in Abhängigkeit vom Datenelement "Art der Information" auszuwerten. Mir liegt leider keine offizielle Liste der möglichen Codes für die Art der Information vor. Die folgenden Aussagen und die Behandlung der Datenelemente durch **EDBS_extra** beruhen auf Erfahrungswerten.

Der Text kann eine topographische Bezeichnung des aktuellen Objekts enthalten. In diesem Fall beginnen die 33 Zeichen mit "GN", "KN" oder "ZN", und die nachfolgende Datengruppe "Geometrieangabe" enthält eine Positionierungsangabe für die Objektbezeichnung.

Andernfalls kann das Textfeld die Objektnummer eines Objekts enthalten, das Bestandteil des aktuellen komplexen Objekts ist.

Schließlich werden auch Referenzangaben für Über- und Unterführungen im Textfeld der Besonderen Information gespeichert, indem dort zwei Objektnummern mit je einer -teilnummer auftreten. Dies sind die Nummern der übereinanderliegenden Objektteile. Die Reihenfolge der Interpretation hängt vom Wert des Datenelements "Art der Information" ab.

Bei ALK-Daten enthält der Textfeld u.a. Straßencode und Hausnummer für Gebäude (Beginn mit "HA") sowie Flurstücksnummern ("FS").

Art der Liniengeometrie und Lageparameter

Das Vorhandensein und die Bedeutung der Datengruppe "Lageparameter" hängt ab von der Ausprägung des Datenelements "Art der Liniengeometrie" beim "Endpunkt der Linie". Der Wert "11" bedeutet "Gerade", und es gibt keine Lageparameter. "15" bedeutet "Vektor", und "Lageparameter" enthält eine Folge von Koordinaten der Brechpunkte (vertices) des Linienzugs.

Im ATKIS-Modell ist auch die Möglichkeit der Repräsentation von Kreisbögen und kubischen Splines vorgesehen. Diese Geometrien werden von **EDBS_extra** (bisher) *nicht ausgewertet*.

Attributtyp und -wert

Eine Datengruppe "Attribut" besteht aus einem Attributtyp (4 Zeichen, z.B. "FKT." für eine genaue Funktionsangabe) und einem Attributwert (7 Zeichen, z.B. "...2613" für "Anlage für astronomische Zwecke" gemäß ATKIS-Objektartenkatalog).

2.6 Beispiel

Grundrißdaten – Objektinformation

```

      1      2      3      4      5      6      7      8
      ┌───┬───┬───┬───┬───┬───┬───┬───┐
      EDBS 0095 0000 FEIN 100005 .0000 UL0BNN..
      9      10     11     12     13     14
      ┌───┬───┬───┬───┬───┬───┬───┬───┐
      0001 0001 34571.6.831594598104 0 0000 0001
      15     16     17     18     19     20     21     22     23     24
      ┌───┬───┬───┬───┬───┬───┬───┬───┬───┬───┐
      0001 104 3102 01 L A999525 L1 920518 .0000
  
```

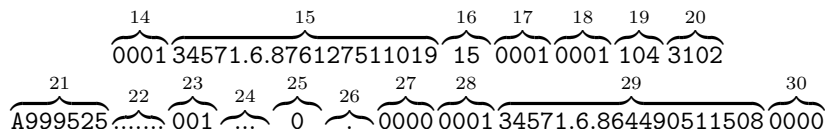
Erläuterungen (vgl. Schema auf Seite 6 und Kap. 2.5):

- 1 Satzanfang
- 2 Satzlänge ist 95 Zeichen/Bytes
- 3 (Anfangsadresse des Suchkriteriums 0)
- 4 Operationsschlüssel für Sekundärnachweis – Grundriß eintragen
- 5 (interne EDBS-Satznummer)
- 6 Zugehörigkeitsschlüssel unbelegt
- 7 (Editier- und Quittungsschlüssel)
- 8 Name der Information für Grundrißdaten
- 9 WHF für alle Datengruppen zusammen (immer 1)
- 10 WHF für Grundrißkennzeichen (immer 1)
- 11 Grundrißkennzeichen (hier: Labelpunkt für Linie)
(3418315.94,5765981.04)
- 12 (Prüfzeichen 0)
- 13 WHF für Liniengeometrie = 0
- 14 WHF für Objektinformation = 1
- 15 WHF für Funktion des Objekts = 1 (immer 1)
- 16 Folie 104
- 17 Objektart 3102 (= Weg)
- 18 (Aktualität des Objekts 01)
- 19 Objekttyp L = Linie
- 20 Objektnummer A999525
- 21 (Modelltyp L1)
- 22 (Entstehungsdatum 18.5.1992)
- 23 (Veränderungskennung)
- 24 WHF für Besondere Information und Geometrieangabe = 0

Grundrißdaten – Liniengeometrie

```

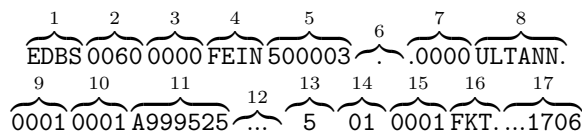
      1      2      3      4      5      6      7      8
      ┌───┬───┬───┬───┬───┬───┬───┬───┐
      EDBS 0152 0000 FEIN 100009 .0000 UL0BNN..
      9      10     11     12     13
      ┌───┬───┬───┬───┬───┬───┬───┬───┐
      0001 0001 34571.6.858334529292 0 0001
  
```



Erläuterungen (siehe auch voriger Abschnitt):

- 2 Satzlänge ist 152 Zeichen/Bytes
- 11 Grundrißkennzeichen (hier: Anfangspunkt) (3418583.34,5765292.92)
- 13 WHF für Liniengeometrie = 1
- 14 WHF für Endpunkt der Linie = 1 (immer 1)
- 15 Endpunkt (3418761.27,5765110.19)
- 16 Art der Lagegeometrie 15 (Vektor = *arc* in ARC/INFO)
- 17 WHF für Funktion der Linie und Fachparameter = 1
- 18 WHF für Funktion der Linie = 1 (immer 1)
- 19 Folie 104
- 20 Linienart (= Objektart) 3102
- 21 Objektnummer 1 (bei Flächen: linkes Objekt) A999525
- 22 Objektnummer 2 (bei Flächen: rechtes Objekt) unbelegt
- 23 Objektteilnummer 1 (vom linken Objekt) 1
- 24 Objektteilnummer 2 (vom rechten Objekt) unbelegt
- 25 (Linienteilung 1 links 0)
- 26 (Linienteilung 2 rechts unbelegt)
- 27 WHF für Fachparameter = 0
- 28 WHF für Lageparameter = 1
- 29 Lageparameter (= *vertex* in ARC/INFO) (3418644.90,5765115.08)
- 30 WHF für Objektinformation = 0

Attributdaten



Erläuterungen:

- 2 Satzlänge ist 60 Zeichen/Bytes
- 8 Name der Information für Attributdaten
- 10 WHF für Attributkennzeichen = 1
- 11 Objektnummer A999525
- 12 Objektteilnummer unbelegt
- 13 (Prüfzeichen 5)
- 14 (Aktualität 01)
- 15 WHF für Attribut = 1
- 16 Attribut FKT (genaue Funktionsangabe)
- 17 Attributwert 1706 (= Radweg)

2.7 Umsetzung der Datenelemente in EDBS_extra-Dateien

Die folgenden Tabellen geben darüber Aufschluß, welche ATKIS-Datenelemente sich nach dem Einlesen und der anschließenden Datenaufbereitung in welchen Dateien der Unterverzeichnisse `edx_files` und `dom_files` wiederfinden. Alle **EDBS_extra**-Dateien sind im ASCII-Format, d.h. als reine Textdateien, abgespeichert. Die Angabe *"nicht gelesen"* bedeutet, daß die entsprechenden Daten beim Lesevorgang zwar benutzt wurden, jedoch keine Notwendigkeit bestand, sie mit abzuspeichern.

Objektinformation:

Datengruppe	ATKIS-DLM-Datenelement	EDBS_extra-Ergebnisdatei
Grundrißkennzeichen	Objektkoordinate	<code>edx_punkte</code> für Punktobjekte <code>edx_l1?000</code> für Linienlabel <code>edx_f1?000</code> für Flächenlabel
	Prüfzeichen	<i>nicht ausgewertet</i>
Funktion des Objekts	Folie	<code>edx_objekte</code>
	Objektart	<code>edx_objekte</code>
	Aktualität des Objekts	<i>nicht ausgewertet</i>
	Objekttyp	<code>edx_objekte</code>
	Objektnummer	<code>edx_objekte</code>
	Modelltyp	<i>nicht ausgewertet</i>
	Entstehungsdatum	<i>nicht ausgewertet</i>
Besondere Information	Veränderungskennung	<i>nicht ausgewertet</i>
	Art der Information	<code>edx_referenz</code> , <code>edx_info</code>
	Kartentyp	<i>nicht ausgewertet</i>
	Signaturteilnummer	<i>nicht ausgewertet</i>
	Text	<code>edx_komplex</code> , <code>edx_referenz</code> , <code>edx_gnamen</code> , <code>edx_knamen</code> , <code>edx_znamen</code> , <code>edx_hausnr</code> , <code>edx_info</code>
Geometrieangabe	Art der Geometrieangabe	<code>edx_nampos</code>
	Objektteilnummer	<code>edx_gnamen</code> , <code>edx_knamen</code> , <code>edx_znamen</code> , <code>edx_hausnr</code> , <code>edx_info</code>
Geometrieangabe	Geometrieangabe	<code>edx_nampos</code>

Liniengeometrie:

Datengruppe	ATKIS-DLM-Datenelement	EDBS_extra-Ergebnisdatei
Grundrißkennzeichen	Anfangspunkt	edx_linien, dom_f?000, dom_l?000
	Prüfzeichen	<i>nicht ausgewertet</i>
Endpunkt der Linie	Endpunkt	edx_linien, dom_f?000, dom_l?000
	Art der Lagegeometrie	<i>bedingt ausgewertet</i>
Lageparameter	Lageparameter	edx_linien, dom_f?000, dom_l?000
Funktion der Linie	Folie	edx_objekte
	Linienart	edx_objekte
	Objektnummer 1	edx_objekte
	Objektnummer 2	edx_objekte
	Objektteilnr. 1	<i>bedingt ausgewertet</i>
	Objektteilnr. 2	<i>bedingt ausgewertet</i>
	Linienteilung 1	<i>nicht ausgewertet</i>
	Linienteilung 2	<i>nicht ausgewertet</i>
Fachparameter	Art des Fachparameters	<i>nicht ausgewertet</i>
	Kennung des Fachparam.	<i>nicht ausgewertet</i>
	Wert des Fachparameters	<i>nicht ausgewertet</i>

Attributdaten:

Datengruppe	ATKIS-DLM-Datenelement	EDBS_extra-Ergebnisdatei
Attributkennzeichen	Objektnummer	edx_attribute
	Objektteilnummer	edx_attribute
	Prüfzeichen	<i>nicht ausgewertet</i>
	Aktualität	<i>nicht ausgewertet</i>
Attribut	Attributtyp	edx_attribute
	Attributwert	edx_attribute

3 Installation von **EDBS_extra**

EDBS_extra besteht aus den C-Quellen und dem `makefile`, der Ihnen vorliegenden Kurzbeschreibung `extra.tex`, den Shell-Skripten `extra.run`, `extra.r1`, `extra.stat` und `extra.mape` und den ARC/INFO-Batchdateien `arc_topo`, `arc_tab` und `arc_display`.

Liste der C-Module

```
admini.c/.h  
atkis.c/.h  
attribut.c/.h  
domino_sort.c  
edbs.c/.h  
edx.c/.h  
fp_edx.c/.h  
grundriss.c/.h
```

Zur Installation von **EDBS_extra** werden diese Dateien auf die Festplatte des Installationsrechners kopiert. Durch den Aufruf des Befehls `make` werden die ausführbaren Dateien `edx` und `dom` erzeugt. Sollten hierbei Fehler oder Warnungen auftreten, so ist zunächst das `makefile` auf seine Verträglichkeit mit dem verwendeten System zu überprüfen (z.B. Vorhandensein des C-Kompilers). Stimmen die Meldungen vom Compiler, so muß u.U. der Programmtext angepaßt werden . . .

EDBS_extra wurde auf SUN Sparc Station IPC und Sparc Station 20 mit 32 bzw. 128 MB Hauptspeicher unter SOLARIS 2.4 entwickelt und getestet. Die im weiteren erwähnten Performance-Werte beziehen sich auf diese Maschinen. Erfahrungen mit anderen (geringeren) RAM-Größen sind bisher nicht bekannt. Für einen vollständigen Lesevorgang sollte mindestens doppelt soviel freie Festplattenkapazität zur Verfügung stehen, wie die **EDBS**-Quelldateien einnehmen!

4 Lesevorgang

4.1 Einleitung

Die Ihnen vorliegenden EDBS-Daten haben Sie — vermutlich von einer Vermessungsbehörde — als "unleserliche" Folge von ASCII-Zeichen erhalten. Aufgabe des Readers ist es, aus diesem Zeichenstrom auf Basis der EDBS- und der ATKIS-Formatbeschreibungen "lesbare" und verwertbare Daten zu machen.

In diesem Schritt werden beispielsweise aufgrund der Überprüfung der Header der EDBS-Sätze die verschiedenen Funktionen zum Lesen der Informationsinhalte aufgerufen. Diese wandeln u.a. die ALK/ATKIS-Koordinatendarstellung in Gauß-Krüger-Format um, kopieren Geometriedaten, die im EDBS-Satz zur Datenreduktion mehreren Objekten zugeordnet waren, und speichern die gewonnenen Informationen *objektweise* ab.

4.2 Aufruf

Das Einlesen der Daten erfolgt durch den Aufruf von `edx`. Als Input wird ein EDBS-Auftrag erwartet, d.h. eine Folge von EDBS-Sätzen, beginnend mit einem Auftragskenn- und abgeschlossen durch einen Auftragsendesatz. Der EDBS-Auftrag muß im ASCII-Format in einer Datei vorliegen, deren Inhalt `edx` per UNIX-Pipetechnik oder -Ausgabeumlenkung zugeführt wird: `edx < edbs_quelle`.

`edx` gibt in diesem Fall die Nummer und die Art des gerade gelesenen EDBS-Satzes auf die Standardausgabe aus. Dieser Output ist hilfreich, um Fehler beim Erkennen der Satzart herauszufinden, kann im Normalfall jedoch durch Umlenkung auf das Nulldevice "geschluckt" werden: `edx < edbs_quelle > /dev/null`.

Fehlermeldungen werden von `edx` nur auf den Standardfehlerkanal, i.d.R. den Bildschirm, geschrieben. Es erfolgt *keine generelle Überprüfung der Syntax* der EDBS/ATKIS-Quelldateien. Wenn der Input nicht mit dem von **EDBS_extra** erwarteten Format übereinstimmt, kann es zu Fehlermeldungen, aber auch zu unbrauchbaren Ergebnissen kommen!

Das Ergebnis des Lesevorgangs wird — aufgrund seiner Komplexität — in einer Reihe von Dateien des Typs `edx_*` im Unterverzeichnis `edx_files` gesichert. Falls diese Dateien vom vorhergehenden Einlesen noch existieren, *müssen Sie sie zunächst unbedingt löschen*, z. B. mit `make clean`, da die neu gelesenen Daten an ein existierendes File angehängt würden. Dies ist nur erwünscht, falls zusammengehörende EDBS-Sätze bearbeitet werden.

Das Einlesen eines EDBS-Auftrags von etwa 600 kB dauert auf einer SPARC 20 ca. fünf Sekunden Realzeit.

4.3 Ergebnis

Die Ergebnisdateien im Unterverzeichnis `edx_files` enthalten gemäß folgender Tabelle zeilenweise die Infos aus je *einem* EDBS-Satz zu je *einem* Objekt:

Datei	Inhalt einer Zeile
<code>edx_linien</code>	Objekt-ID, Objektteilnummer, Objektart, Punktfolge (x- und y-Koordinaten im Gauß-Krüger-Format). Diese Datei wird von <code>dom</code> weiterverarbeitet.
<code>edx_punkte</code>	Objekt-ID, Koordinatenpaar für Punktobjekt
<code>edx_fl?000</code>	Objekt-ID eines <i>Flächenobjekts</i> aus dem Objektbereich '?', Labelpunkt (Koordinatenpaar des Grundrißkennzeichens gemäß ATKIS-Struktur)
<code>edx_ll?000</code>	Objekt-ID eines <i>Linienobjekts</i> aus dem Objektbereich '?', Labelpunkt
<code>edx_objekte</code>	Objekt-ID, Objektnummer, Objektart, Folie, Objekttyp (P, L, F, K). Zu jedem Objekt muß es <i>genau eine</i> Zeile in <code>edx_objekte</code> geben.
<code>edx_nachbar</code>	Objekt-ID, Objekt-ID. Diese zwei Objekte teilen sich eine gemeinsame Grenzlinie. (Außer Überführungsreferenzen die einzige Topologie-Information, aber noch ungetestet!)
<code>edx_komplex</code>	Objekt-ID eines komplexen Obj., ID eines Teilobj.
<code>edx_referenz</code>	Objekt-ID, Art der Überführungsreferenz, Objekt-ID, Objektteilnummer des ersten, Objekt-ID, Objektteilnummer des zweiten referierten Objekts
<code>edx_gnamen</code>	Objekt-ID, Objektteilnummer, geographischer Name
<code>edx_knamen</code>	Objekt-ID, Objektteilnummer, Kurzname
<code>edx_znamen</code>	Objekt-ID, Objektteilnummer, Zweitname
<code>edx_nampos</code>	Objekt-ID, Objektteilnummer, Positionierungspunkte für einen Namen.
<code>edx_hausnr</code>	Objekt-ID, Objektteilnr., Straße, Hausnr. (ALK)
<code>edx_info</code>	Objekt-ID, Objektteilnummer, Art der Information, Text der "besonderen Information", soweit nicht anders ausgewertet
<code>edx_attribute</code>	Objekt-ID, Objektteilnummer, Attributtyp, Attributwert. I.d.R. gibt es zu einem Objekt mehrere Attributdatensätze (= Zeilen in <code>edx_attribute</code>).
<code>edx_auftragskennung</code>	Dienststelle, Text für die Ausgabe, Anzahl der weiteren EDBS-Sätze, Datumsangaben aus dem Auftragskennsatz.
<code>edx_gebiet</code>	Koordinaten der linken unteren und der rechten oberen Ecke des Kartenblattes.

In Abschnitt 2.7 wurde bereits aus Sicht des ALK/ATKIS-Datenmodells aufgelistet, welche Datenelemente in welcher Ergebnisdatei gesichert werden. Dort können Sie auch feststellen, welche Datenelemente von **EDBS_extra** *nicht* ausgewertet werden.

5 Daten-Aufbereitung

5.1 Zweck

Zusammengehörige Geometrieangaben können im EDBS-Auftrag auf viele, weit auseinanderliegende Sätze verstreut sein. Insbesondere sind Objektdaten (Geometrie und Objektart) und Attributdaten voneinander getrennt. Die Verbindung ist lediglich durch die eindeutige ATKIS-Objektnummer gesichert, die zur späteren Weiterverarbeitung im GIS von **EDBS_extra** in eine ganzzahlige ID umgewandelt wird (vgl. Hinweis unter 7.2 auf Seite 20).

Um vielgliedrige Linienzüge sowie Flächenumrandungen wieder zusammenzufügen, wird zunächst die gelesene Liniengeometrie (`edx_linien`) nach den Objekt-IDs sortiert. Anschließend versucht das Modul `dom`, Linienzüge mit gleicher Objekt-ID zu verketteten. Abhängig von der Objektart und vom Erfolg der Verkettung werden alle zu einem Objekt gehörenden Linienzüge unter ihrer ID zusammen abgespeichert.

Da das Modul versucht, neue Linienzüge an allen Enden und in allen Richtungen zu verknüpfen, nenne ich diesen Prozess "*Domino-Sortierung*". (Eine ausführliche Beschreibung des verwendeten Algorithmus soll später erfolgen.)

5.2 Aufruf

Das UNIX-Kommando `sort -o edx_linien_sortiert edx_linien` sortiert die Linienzüge. Die Verkettung wird mit `dom < edx_linien_sortiert` gestartet.

Sortierung und Verkettung dauern auf dem Entwicklungsrechner bei einem typischen EDBS-Auftrag wiederum etwa fünf Sekunden Echtzeit.

5.3 Ergebnis

Die Dateien `dom_f?000` enthalten — getrennt nach ATKIS-Objektbereich — die Objekte, für die die Verkettung der einzelnen Linienzüge einen geschlossenen Umring ergeben hat. Auf die Objekt-ID folgen zeilenweise die Stützpunkte des Umrings. Ein Objekt wird mit einer "END"-Zeile abgeschlossen.

Die Dateien `dom_l?000` enthalten unter ihrer ID die verketteten Linienzüge der Objekte, bei denen kein geschlossener Umring entstanden ist. Falls bei der Verkettung für eine Objekt-ID nicht alle Linienzüge miteinander verknüpft werden konnten, werden diese Linienzüge getrennt hintereinander unter der gleichen ID abgespeichert.

`dom_objekte` enthält zu jedem bearbeiteten Objekt die Information, wieviele Linien (von insgesamt wievielen Objektteilen) verkettet wurden.

6 Automatisierung

6.1 Daten einlesen

Das Shell-Skript `extra_run` zeigt exemplarisch, wie das Einlesen der Daten mit **EDBS_extra** automatisiert werden kann. Wenn Sie mehrere EDBS-Aufträge (= Quelldateien) zu bearbeiten haben, so schreiben Sie die Dateinamen in eine Liste: `ls x-v34* > extra_quellen`. Wenn Sie diesem Beispiel folgen, dürfen Sie nicht vergessen, den Namen "extra_quellen" selbst wieder aus der Datei `extra_quellen` zu löschen. Sie können natürlich auch nur einen oder wenige Dateinamen (von Hand) in die Liste schreiben.

`extra_run` ruft für jede Datei in dieser Liste einmal das Hilfsskript `extra_r1` auf, das einen einzelnen EDBS-Auftrag bearbeitet (Einlesen, Sortierung, Verkettung). Die neuen Daten werden dabei jeweils an die bestehenden Dateien *angehängt*. Daher muß `extra_run` vor dem ersten Aufruf von `extra_r1` alle Überbleibsel von früheren Aufrufen löschen — *sichern Sie also noch benötigte Daten* unter anderen Dateinamen als den vom Programm vergebenen (`edx_*` und `dom_*` in den entsprechenden Unterverzeichnissen)!

Nach der Schleife, d.h. nach Bearbeitung des letzten EDBS-Auftrags, können noch Aufräumarbeiten durchgeführt werden. Zur Weiterverarbeitung in ARC/INFO müssen z.B. die Dateien, die Geometriedaten enthalten, mit einem "END" abgeschlossen werden: `echo 'END' >> dom_12000` etc.

Die vollständige Bearbeitung aller ATKIS-Quellen des Landkreises Osnabrück dauerte ca. 45 Minuten.

6.2 Kartenausdehnung

Das Skript `extra_mape` wertet die Datei `edx_gebiet` mit den Blattecken der bearbeiteten EDBS-Quellen insoweit aus, daß die kleinste vorkommende x- und die kleinste y-Koordinate aller linken unteren Ecken und die größte x- und y-Koordinate aller rechten oberen Ecken in die Datei `edx_mapextent` ausgegeben werden, die damit den kleinstmöglichen rechteckigen Ausschnitt definiert, der alle gelesenen ATKIS-Objekte enthält. Diese Koordinaten können Sie im `mape`-Befehl in ARC/INFO verwenden.

6.3 Statistik

Schließlich stellt das Skript `extra_stat` Ihnen die Möglichkeit zur Verfügung, einige statistische Angaben über den letzten Lesevorgang abzufragen. Die Berechnung erfolgt auf Basis der `edx_*`- und `dom_*`-Dateien erst mit Aufruf von `extra_stat`; daher dauert die Ausführung des Befehls eine Weile. Benutzt werden die UNIX-Kommandos `grep` und `wc`, indem das Muster "END" in den Ergebnisdateien gesucht wird und die gefundenen Zeilen gezählt werden.

7 Einlesen in ARC/INFO

7.1 Einleitung

In diesem Kapitel wird erläutert, wie die `edx.*`- und `dom.*`-Files mit Objektgeometrie und Sachdaten in die Datenbank von ARC/INFO umgesetzt werden können. Ziel ist dabei die Erzeugung von nach ATKIS-Objektbereich und nach Objekttyp getrennten Coverages für Linien- bzw. Flächenobjekte sowie eines Coverages für alle Punktobjekte. Die Unterscheidung nach Objekttypen ermöglicht also ein getrenntes Einlesen in coverages für die feature types in ARC/INFO (point, line und area), während die Unterscheidung nach Objektbereichen diese coverages übersichtlicher macht.

Bei ALK-Daten werden (wg. der zu erwartenden geringeren Datenmengen) alle Objekte in einem Coverage abgespeichert. Desweiteren werden Objektinformation und Attributangaben (ATKIS) in INFO-Tabellen eingelesen.

Die Skripte wurden unter ARC/INFO Version 6.1.2. und 7.0.2. getestet. Es handelt sich um Dateien mit Befehlssequenzen, die ARC/INFO als Batchdateien zugeführt werden: `cat arc_topo | arc`.

Der Import der Daten ins GIS ist *wesentlich zeitaufwendiger* als das zuvor beschriebene Einlesen und Aufbereiten. Es handelt sich bei der Konvertierung der Daten des Landkreises Osnabrück (ca. 2000 km²) um mehrere Stunden.

7.2 Erzeugen der Coverages

Das Einlesen der Geometrieminformation von Linien und Flächen aus den ASCII-Dateien in ARC/INFO geschieht folgendermaßen (vgl. Skript `arc_topo`):

```
GENERATE <polygon_cover> bzw. <linien_cover>
      INPUT /.../DOM_FILES/DOM_F?000 bzw. DOM_L?000
      POLYGONS bzw. LINES
      INPUT /.../EDX_FILES/EDX_FL?000 bzw. EDX_LL?000
      POINTS
QUIT
```

Die erste Zeile startet von `arc`: aus das `GENERATE`-Kommando, das als Parameter einen neuen Coverage-Namen erhält, z.B. "`see`" für ein Coverage, das die flächenhaften Objekte des ATKIS-Objektbereichs 5 enthalten soll.

Innerhalb des Generate-Moduls wird die Inputdatei angegeben, z.B. `dom_files/dom_p5000` für die Gewässerflächen. Nach dem Einlesen der Polygone wird als zweiter Input die Datei mit den Labelpunkten angegeben, im Beispiel das File `edx_files/edx_p15000`. Das Einlesen der Labelpunkte ist notwendig, damit ARC/INFO die von `EDBS_extra` übergebene ID als User-ID im feature attribute table einträgt.

Die User-ID (Objekt-ID) wird von **EDBS_extra** aus der ATKIS-Objektnummer berechnet, da Arc/Info für dieses Item keine Kombination aus Ziffern und Buchstaben zuläßt. Aufgrund des maximal möglichen Werts für die User-ID können bei der Berechnung von insgesamt sieben Stellen nur die letzten fünf Stellen berücksichtigt werden. Überprüfen Sie daher beim Auftreten von Inkonsistenzen, ob die Datei **edx_objekte** Zeilen mit verschiedener ATKIS-Objektnummer aber gleicher Objekt-ID enthält (**sort, uniq**).

Nach dem Einlesen der Definitionsgeometrie wird die Topologie erzeugt. Bei Liniencoverages wird **BUILD** benutzt, da die ATKIS-Objekte bei Überschneidungen die Objekt*teil*nummer wechseln, so daß es keine Intersections geben sollte. Dies wird bei der Verkettung durch das **EDBS_extra**-Modul **dom** auch berücksichtigt, indem Objektteile von Linienobjekten getrennt abgespeichert werden, so daß bei **GENERATE LINES** dort ein Knoten (node) entsteht, wo zwei Objektteile aneinanderstoßen.

Bei Polygoncoverages muß jedoch der Befehl **CLEAN** benutzt werden, da bei benachbarten Flächen die Grenze zweimal vorhanden ist, und Arc/Info auf dem Mittelpunkt jeder Grenzlinie eine Intersection diagnostiziert.

Nach dem Erzeugen der **.aat/.pat**-Files werden diesen zwei weitere Items hinzugefügt, um die Selektion von arcs bzw. polygons nach Objekt-ID und Objektart zu erleichtern: **ADDITEM L_2000.PAT 1_2000.PAT ID 4 10 B** und **ADDITEM L_2000.PAT 1_2000.PAT ART 4 5 B**. Sobald die INFO-Tabelle "OBJEKTE" existiert (vgl. Abschnitt 7.3), werden die neuen Items im INFO-Modul wie folgt mit Werten belegt:

```
SELECT L_2000.PAT
      CALCULATE ID = L_2000-ID
      SORT ON ID
      RELATE OBJEKTE BY ID
      CALCULATE ART = $1ART
```

Die ID wird also gleich der User-ID (auch Cover-ID) gesetzt. Dies dient ausschließlich der Verkürzung des Itemnamens. Die Objektart wird aus dem gleichnamigen Feld der Objekttabelle kopiert.

Falls Sie feststellen, daß sie häufig Selektionen nach Objektarten durchführen, empfiehlt es sich, einen Index auf das Item **ART** erstellen zu lassen:

```
INDEX L_2000 LINE
      INDEXITEM L_2000.PAT ART
```

7.3 Erzeugen der Attributtabellen

Die Batchdatei **arc_tab** wird dazu benutzt, die tabellarischen Ergebnisse von **EDBS_extra** in INFO-Tabellen einzulesen. In der folgenden Übersicht sind die **edx_***-Inputdateien den INFO-Tabellen gegenübergestellt:

Inputdatei	INFO-Tabelle	Items (vgl. Text)
<code>edx_attribute</code>	ATTRIBUTE	ID, OTN, ATYP, AWERT
<code>edx_objekte</code>	OBJEKTE	ID, ONR, ART, TYP, FOLIE
<code>edx_gnamen</code>	GNAMEN	ID, GNAME
<code>edx_knamen</code>	KNAMEN	ID, KNAME
<code>edx_znamen</code>	ZNAMEN	ID, ZNAME
<code>edx_hausnr</code>	HAUSNR	ID, OTN, STRASSE, NUMMER
<code>edx_komplex</code>	KOMPLEX	ID, TEILID
<code>edx_referenz</code>	REFERENZ	ID, RTYP, ID1, OTN1, ID2, OTN2

Die wichtigste Tabelle ist sicherlich **OBJEKTE**, denn sie erlaubt die Zuordnung von Objektart und Folie sowie Objektnummer und -typ zu einzelnen Objekten. Der Zugriff erfolgt über **RELATE** mit der **ID** als Relate-Item.

Die Tabellen **?NAMEN** enthalten geographische, Kurz- und Zweitnamen, die nur für ALK-Daten sinnvolle **HAUSNR**-Tabelle enthält Straßencode und Hausnummer für Gebäude-Polygone.

KOMPLEX speichert zeilenweise die ID eines komplexen Objekts und die ID eines Objekts, welches Bestandteil des ersten ist. Jedes komplexe Objekt müßte daher in mindestens zwei Zeilen auftreten — mit jeweils einem seiner Teilobjekte.

Die **REFERENZ**-Tabelle verknüpft übereinanderliegende Objekte (z.B. Autobahn und Eisenbahnlinie unter einer Autobahnbrücke). Das Item **RTYP** unterscheidet zwischen Über- und Unterführung (codiert gemäß ATKIS-Referenztabelle). Die Referenz bezieht sich auf zwei Objekt*teile*.

ATTRIBUTE listet schließlich die Sachdaten als Kombination von Attribut ("Attributtyp") und Ausprägung ("Attributwert") auf.

7.4 Darstellung

Die Erarbeitung einer Darstellungsform für die gelesenen ATKIS-Daten hängt von der jeweiligen Anwendung ab und muß dem ARC/INFO-Fachmann vorbehalten bleiben. I.d.R. werden dabei Lookup-Tabellen benutzt werden, die den ATKIS-Objektarten bestimmte Signaturen (Symbole) und Bezeichnungen zuordnen.

8 Programmbeschreibung

Die nachfolgende Tabelle zeigt die Aufgabenverteilung innerhalb der C-Module von **EDBS.extra**:

Modul	Aufgabe
<code>edx.c/.h</code>	Hauptprogramm. Globale Konstanten, Funktion <code>nummer2id()</code> zur Konvertierung von ATKIS-Objektnummern in Integer
<code>fp_edx.c/.h</code>	Konstante für Unterverzeichnis <code>edx_files</code> . Filepointer für Ergebnisdateien. Funktionen zum Öffnen und Schließen der Dateien.
<code>admini.c/.h</code>	ALK/ATKIS-Administratives. Funktionen zum Abspeichern von Gebietskennzeichnung, Auftrags-text und Auftragskennung. Variable <code>datenmodell</code> für ALK/ATKIS-Unterscheidung.
<code>atkis.c/.h</code>	ATKIS-Spezifika (i. Ggs. zu ALK). Typ für Gauß-Krüger-Punkt und Funktion zum Lesen und Konvertieren eines Punkts aus ATKIS in Gauß-Krüger-Format.
<code>edbs.c/.h</code>	EDBS-Administratives. Funktionen zum Lesen eines EDBS-Satzes und ggfs. Fortsetzungssätze aus <code>stdin</code> in Variable <code>satz</code> und zum Lesen einzelner Datentypen (String, Integer, WHF, Zeichen) aus <code>satz</code> . Funktion zum Hochsetzen der Position <code>pos</code> im aktuellen Satz, ohne zu lesen. Check auf Art des EDBS-Satzes
<code>grundriss.c/.h</code>	Lesen der Geometrien. Konstanten für Obergrenzen für Anzahl Objektarten pro Linie und Anzahl Interpolationspunkte. Variablen <code>anfang</code> , <code>ende</code> , <code>lapa[]</code> ; <code>anzahl_lapa</code> , <code>anzahl_doppel</code> ; <code>objektart[]</code> , <code>folie[]</code> , <code>objekt1/2[]</code> , <code>objektteil1/2[]</code> , <code>richtung</code> , <code>objekttyp</code> , <code>art_info</code> , <code>text[]</code> , <code>id</code> .
<code>domino_sort.c</code> <code>attribut.c/.h</code>	"Domino-Sortierung" der abgespeicherten Linien. Extraktion der Sachdaten. Funktionen zum Lesen und Abspeichern von Attributtypen und -werten aus <code>satz</code> .

In Analogie zu Tabelle 4.3 seien noch die in **EDBS_extra** verwendeten Filepointer und die in den Files jeweils abgespeicherten Variablen aufgelistet:

Filepointer	Output-Variable
fp	nummer2id(objekt1/2[]), objektteil1/2, objektart[], anfang.x/y, lapa[] .x/y, ende.x/y
fp_points	id, anfang.x, anfang.y
fp_pl?000	id, anfang.x, anfang.y
fp_ll?000	id, anfang.x, anfang.y
fp_objects	id, objekt1[0], objektart[0], objekttyp, folie[0]
fp_nachbar	nummer2id(objekt1), nummer2id(objekt2)
fp_complex	id, nummer2id(nummer)
fp_refer	id, art_info, nummer2id(text[0..6]), text[7..9], nummer2id(text[10..16]), text[17..19]
fp_gname	id, -, text[2..33]
fp_kname	id, -, text[2..33]
fp_zname	id, -, text[2..33]
fp_npos	id, adg, gp.x, gp.y
fp_ha_alk	id, -, text[2..14], text[15..18]
fp_noname	id, -, art_info, text[0..33]
fp_att	nummer2id(obj_nr), atoi(teil_nr), att_typ, att_wert
fp_auftrag	-, -, -, -
fp_gebiet	lu.x, lu.y, ro.x, ro.y

Weitere Details zur Implementation von **EDBS_extra** sind den Kommentaren im Programmtext zu entnehmen. Die Erläuterung des Zwecks einzelner Funktionen findet sich dabei in den .c-Modulen, nicht in den .h-Dateien.

9 Literatur

1. AG Hannover in den Vorhaben ALK/ATKIS: *Dokumentation zum ALK/ATKIS-Datenaustausch*. Nds. Landesverwaltungsamt – Landesvermessung –, Hannover 1993
2. Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland (AdV): *Gesamtdokumentation ATKIS*. Landesvermessungsamt NRW, Bonn 1989
3. Environmental Systems Research Institute (ESRI): *ARC/INFO 6.1 User's Guide: ARC/PLOT Command References*. Redlands, CA 1992
4. Krämer & Dr. Schüller: *Kommunikationswerkzeuge mit unterschiedlichen GIS-Plattformen zum Austausch raumbezogener Daten*. (Studie im Auftrag des UBA) Krefeld 1993