

ADIS/ADED Handbuch

Abschrift der Quelle <http://ian.lkv-nrw.de>

Basis ist der Standard (siehe S. 27)

Inhaltsverzeichnis

1	ADIS/ADED Einführung	4
1.1	Wo wird ADIS eingesetzt?	4
2	Beschreibung ADIS	5
2.1	Allgemein	5
2.2	Dateibezeichnung	5
2.3	Zeilentypen	6
2.4	Statuszeichen	7
2.5	Kombination von Zeilentypen und Statuszeichen	9
2.6	Ereignisnummer	9
2.7	Abfrage	10
2.8	Feldlänge und Auflösung	12
2.9	Rückstellung von Nachrichteneinstellungen	12
2.10	DDI-Nummer	12
2.11	Datenfelder	12
2.12	Daten und Zeitformat	13
2.13	Kontrollsummen	13
2.14	Unbekannte DDI-Nummern	14
2.15	Kommentar	14
2.16	Dateistruktur	15
3	Beschreibung ADED	16
3.1	Allgemein	16
3.2	Datenelemente	16
3.3	Datenobjekt(Entity)	17
3.4	Header	19
4	Deutsches Data Dictionary	20
4.1	Anwendungsbereich	20
4.2	Durchführen des Datanaustausches	20
4.2.1	Allgemeines	20
4.2.2	Standarverzeichnis und Dateiname	21
4.2.3	Interfaceprogramm	21
4.2.4	Dumbfile	21
4.2.5	Errorfile	21
4.3	Datenelemente	22
4.4	Entities (Datenobjekte)	22

4.5	Verwaltung des Data Dictionary	23
5	Beispiel	24
5.1	Datenblock	24
5.2	Abruf des Data Dictionarys mit einer Query vom ADISSERVER	24
5.2.1	Abfrage per Telnet	24
5.2.2	Benutzung eines Client Programms	25
5.2.3	Generierung von XML aus ADS-Daten über den Server des LKV (Dienstleistung)	26
6	ADIS/ADED Standards	27
7	ADIS Header	28
7.1	Was gehört in einen ADIS-Header zum Datenaustausch nach DIN 11800?	28
7.1.1	Gemäß internationaler ISO-Norm:	28
7.1.2	Gemäß nationaler Festlegung (DIN):	29
7.2	Beispiel eines "Minimal-Headers"	29
8	Beschreibung des Clients	30
8.1	Mini Query-Client für Abfragen auf dem ADISSERVER des LKV Westfalen- Lippe.	30
8.2	ADIS-Dateien auf dem ADISSERVER des LKV Westfalen-Lippe verar- beiten (XML)	31

1 ADIS/ADED Einführung

1.1 Wo wird ADIS eingesetzt?

Mit ADIS-ADED kann ein standardisierter elektronischer Datenaustausch zwischen:

- einem Prozeßrechner (z.B. Fütterungscomputer) und einem Anwendungsprogramm auf dem Betriebs-PC (z.B. Herdenmanagement)
- zwei verschiedenen Prozeßrechnern (z.B. Melkcomputer und Fütterungscomputer)
- einem Anwendungsprogramm auf dem Hof-PC (z.B. Herdenmanagement) und einem externen Dienstleister (z.B. Landeskontrollverbände) usw.

durchgeführt werden.

Theoretisch ist der Anwendungsbereich nicht begrenzt. Für den kommerziellen Bereich hat sich jedoch EDIFACT als Standard zur Datenübertragung etabliert, so daß die Anwendung von ADIS-ADED überwiegend im landwirtschaftlichen Bereich stattfindet.

2 Beschreibung ADIS

2.1 Allgemein

Die ADIS-Datei besteht aus ASCII-Zeichen (ISO 8-bit Code) und jede Datenzeile endet mit <CR><LF>. Um die Daten der ADIS-Datei zu verschlüsseln bzw. zu entschlüsseln, muß sowohl der Sender als auch der Empfänger der ADIS-Datei die gleiche Version des Data Dictionaries (ADED) benutzen. Das Data Dictionary enthält die genaue Definition der übermittelten Merkmale. Jedes Merkmal ist durch eine 8-stellige Nummer eindeutig kodiert. Die folgenden Zeichen werden im folgenden verwendet, um eine bessere Beschreibung der ADIS-Syntax zu geben:

<...> enthält die Beschreibung eines Feldes;

[...] wählbare (nicht obligatorische) Teile sind in eckigen Klammern aufgeführt;

(...) Teile zwischen Klammern können einige Male wiederholt auftreten, mindestens einmal.

Die Mindestzeilenlänge, die ein Programm unterstützen muß, ist 256 Zeichen, einschließlich <CR><LF>. Die Maximalzeilenlänge in einer ADIS-Datei ist nicht spezifiziert, sollte aber durch die Nachrichtengruppe, die die Syntax verwendet, abgestimmt sein.

ZEICHEN	NAME	DEC	HEX	Funktion
M	CR	13	0D	Wagenrücklauf
J	LF	10	0A	Zeilenvorschub
I	vertikaler Balken	124	7C	Fehlercode für unbekannte DDI-Nummern
?	Fragezeichen	63	3F	Nullwert (nicht initialisierte Felder)

ANMERKUNG – ASCII definiert die Zeichen 0 bis 31 und 127 als Steuerzeichen (oder nicht druckbare Zeichen). Die gezeigten Befehlsfolgen können verwendet werden, um diese Werte mit den meisten Systemkeyboards einzugeben. Abgesehen von diesen belegten Zeichen können alle Zeichen mit ASCII-Werten von 32 bis 255 (ausgenommen 127) verwendet werden.

ANMERKUNG – In einigen Fällen können ASCII-Werte über 127 Probleme verursachen, weil sie vom Land abhängig sind.

2.2 Dateibezeichnung

Der Dateiname kann frei vergeben werden.

2.3 Zeilentypen

Jede Zeile in der ADIS-Datei gehört zu einem bestimmten **Zeilentyp**. Mit dem ersten Zeichen in einer Zeile wird der Zeilentyp definiert und es gibt an, welche Daten in der Zeile enthalten sind. Der Zeilentyp muß standardmäßig mit einem Großbuchstaben angegeben werden. Es gibt folgende Standard-Zeilentypen:

Zeilentyp	Bezeichnung
D	Definition
V	Value
E	End of logical file
C	Comment
S	Search
R	Request
F	File
I	Include
O	Output
T	Terminate
Z	Physical end of file

Andere Zeilentypen sind herstellerspezifisch und müssen zwischen sendenden und empfangenden Teilnehmer abgestimmt sein. Wenn der Zeilentyp unbekannt ist, wird die Zeile ignoriert.

Die ADIS-Eigenschaften können schrittweise verwendet werden. Daher sind die Zeilentypen in vier verschiedene Klassen eingeteilt:

Klasse A: Verwendung der Zeilentypen **D, V, C, E** und **Z**;

Klasse B: Klasse A plus zusätzlich Verwendung von **F-, I- und T-Zeilen**;

Klasse C: Klasse A plus zusätzlich Verwendung von **R- und O-Zeilen**;

Klasse D: Klasse A plus zusätzlich Verwendung von **S-, R- und O-Zeilen**.

Klasse A ist obligatorisch für die Verwendung von ADIS. Die Klassen B, C und D sind erlaubte Erweiterungen.

Das heißt nicht, daß, wenn z.B. Klasse C unterstützt wird, automatisch Klasse B unterstützt wird. Beim Synchronisieren (Initialisieren) müssen Sender und Empfänger sich gegenseitig mitteilen, in welcher Klasse sie kommunizieren können. Dies kann z.B. auf Papier erfolgen, aber ebenso automatisch durch Verwendung einer DDI-Nummer in der ADIS-Datei für die Einteilung des Zeilentyps.

D = "Definition"- Zeile

In einer D-Zeile wird der Inhalt der nachfolgenden V-Zeilen angegeben. Einer D-Zeile müssen immer eine oder mehrere V-Zeilen folgen. Wenn keine Daten verfügbar sind, müssen alle Felder (ab der Position 9) der gesamten V-Zeile mit "?" gefüllt werden. Die D-Zeile soll nach folgendem Schema aufgebaut sein: D<Status><Entity-Nr.>(<DDI-Nr.><Feldlänge> <Dezimalstellen>) <CR><LF>

V = "Value" - Zeile

Die V-Zeile enthält die Werte der Datenelemente, die in der letzten D-Zeile aufgeführt sind. Die V-Zeile soll nach folgendem Schema aufgebaut sein: V<Status><Entity-Nr.>(<Wert>)<CR><LF>

E = "End of logical file" - Zeile

Die E-Zeile markiert das logische Ende einer ADIS-Datei. In einer Datei können meh-

rere E-Zeilen vorkommen. Der E-Zeile folgt immer eine DH- und VH-Zeile oder eine Z-Zeile. Das physische Ende eines Files wird durch eine Z-Zeile angegeben. Die letzte E-Zeile vor der Z-Zeile ist nicht erforderlich. Die E-Zeile hat immer den gleichen Aufbau: EN<CR><LF>

C = "Comment" - Zeile

Die C-Zeile kann an beliebiger Stelle in den ADIS-File eingefügt werden, und sie kann beliebigen Text enthalten. Die C-Zeile soll nach folgendem Schema aufgebaut sein: C<Status><beliebiger Text><CR><LF>

S = "Search" - Zeile

In einer S-Zeile können eine oder mehrere Bedingungen spezifiziert sein. Siehe auch Abschnitt Abfragen.

R = "Request" - Zeile

Eine R-Zeile beinhaltet die Data Dictionary-Nummern von Attributen, die angefragt werden. Siehe auch Abschnitt Abfragen

F = "File" - Zeile

Eine F-Zeile gibt den vollständigen Namen einer Bibliographie-Datei an (Pfadname und Dateiname). Diese Zeile vermeidet die Datei selbst einzuschließen. Die F-Zeile hat folgenden Aufbau: F<Status><Pfad + Dateiname><CR><LF>

I = "Include" - Zeile

Eine I-Zeile beinhaltet den Pfadname und den Dateiname einer auf der I-Zeilen-Position einzufügenden Datei. Die Einfügezeile soll die ADIS-Anforderungen erfüllen, als ob sie Teil der übergeordneten Datei wären. Dies impliziert, daß die Kombination von übergeordneter Datei und der Einfügedatei den ADIS-Anforderungen entspricht. Die I-Zeile hat folgenden Aufbau: I<Status><Pfad + Dateiname><CR><LF>

ANMERKUNG – Die Option des Einfügens ist ein Weg, um Dateien physikalisch zu trennen, die logisch eine einzige sind. Die eingefügte Datei kann einen Header enthalten, wenn die Kombination mit der übergeordneten Datei die ADIS-Anforderungen erfüllt.

O = "Output" - Zeile

Eine O-Zeile beinhaltet den Pfadnamen und den Dateinamen einer Datei, zu der Daten geschrieben werden müssen. Die O-Zeile hat folgenden Aufbau: O<Status><Pfad + Dateiname><CR><LF>

T = "Terminate" - Zeile

Eine T-Zeile kennzeichnet das Ende einer Informationshaupteinheit in einer ADIS-Datei. Eine T-Zeile kann verwendet werden, um unterschiedliche Blöcke innerhalb einer Datei zu unterscheiden. Nach einer T-Zeile kann keine DH + VH-Zeile (siehe auch E-Zeile) folgen. Eine T-Zeile ist immer gleich: TN<CR><LF>

Z = "Physical end of file" - Zeile

2.4 Statuszeichen

Auf die 2. Position in jeder Zeile muß das **Statuszeichen** in Form von Großbuchstaben eingefügt werden. Das Statuszeichen beschreibt die Funktion der Daten, die in der Zeile

stehen. Folgende Statuszeichen stehen standardmäßig zur Verfügung:

Statuszeichen	Bezeichnung
H	Header - Daten
N	Normal - Daten
S	Synchronisation - Daten
F	Faulty - Daten
D	Deletion - Daten

H = "Header"

Daten Jede ADIS-Datei muß einen Header haben. Dies impliziert, daß eine ADIS-Datei mit einer DH-Zeile beginnt, gefolgt von einer VH-Zeile. Lediglich Kommentar-Zeilen (ungeachtet ihrer Statuszeichen) können dem Header vorangehen. Wenn der Header ausgelassen wird, ist die ganze Datei unzulässig. Es ist obligatorisch, die Header-Daten (direkt nach der Event-Nummer) mit dem Namen des Data Dictionaries, der beim Erstellen der Datei verwendet wurde, zu beginnen. Es können mehrere Data Dictionaries verwendet werden. Um die erhaltene ADIS-Datei zu lesen muß der Empfänger wissen, welches Data Dictionary zur Erkennung der DDI-Nummern verwendet werden muß.

Hersteller: Organisation/Firma, die die ADIS-Datei erstellt hat;

Empfänger: Name des Empfängers der ADIS-Datei;

DDI-Listenversion: Version der DDI-Liste, die beim Erstellen der Datei verwendet wurde;

Produktname: Name des Softwareprogrammes, das beim Erstellen der Datei verwendet wurde;

Softwareversion: Programmversion, die beim Erstellen der Datei verwendet wurde;

Erstellungsdatum: Datum der Erstellung der Datei;

Erstellungszeit: Zeit der Erstellung der Datei.

Es ist zulässig, mehr Daten in den Header zu setzen. Die DDI-Nummern dieser Datenfelder müssen Teil des Data Dictionaries sein.

N = "Normal" - Daten Normale Daten werden bereitgestellt, um den Datenbestand des Empfängers zu aktualisieren.

S = "Synchronisation" - Daten

die Datenbasis zu initialisieren (erste Eingabe/Füllung);

Daten in der Datenbasis der sendenden und empfangenden Partei zu vergleichen, um zu bestimmen, welche korrekt zwischen den Nachrichtengruppen ist.

F = "Faulty" - Daten Wenn eine ADIS-Datei eine Zeile enthält, die einen oder mehrere Fehler zu enthalten scheint, ist es für den Empfänger möglich, diese ganze Zeile in eine separate ADIS-Datei zu geben, mit dem in "F" geänderten Statuszeichen. Es ist ebenfalls möglich, daß eine F-Statuszeile mehr Felder als die entsprechenden N-Statuszeile hat. Die F-Statuszeile kann z.B. detaillierte Informationen über die Fehlerart, was falsch ist, etc. enthalten. Diese Art von Informationen können in einer F-Statuszeile enthalten sein

gemäß der vorangehenden DF-Zeile. Der Empfänger entdeckt die Fehler in der ADIS-Datei und sendet die F-Zeile zurück zum ursprünglichen Sender.

D = "Deletion" - Daten Wenn eine ADIS-Datei versandt wurde, die falsche Informationen enthält, kann der Übermittler die Zeile, die den Fehler enthält, in eine andere ADIS-Datei kopieren mit einem in "D" geänderten Statuszeichen und kann diese Datei ebenfalls versenden. Eine Zeile mit einem D-Status soll vollständig gelöscht werden. Die Verwendung des D-Status ist wählbar. Wenn der D-Status nicht unterstützt wird, soll er durch eine F-Zeile beantwortet werden.

2.5 Kombination von Zeilentypen und Statuszeichen

Die Matrix zeigt, welche Kombinationen von Zeilentypen und Statuszeichen erlaubt sind und welche nicht.

Zeilentyp		Statuszeichen				
		H	N	S	F	D
		Header	Normal	Synch	Fehler	Löschen
D	Definition	+]]	+]]	+]]	+]]	+]]
V	Wert	+]]	+]]	+]]	+]]	+]]
E	Ende	-	+	-	-	-
C	Kommentar	+	+	+	+	+
S	Suche	-	+]1)	-	+]2)	-
T	Abschluß	-	+	-	-	-
F	Datei	-	+	-	+	-
I	Einfügen	-	+	-	+	-
O	Ausgabe	-	+	-	+	-
Z	Physikalisches Ende	-	+	-	-	-

1) Eine S-Zeile erfordert eine R-Zeile, aber eine R-Zeile keine S-Zeile
2) Eine S- oder R-Zeile mit einem F-Status ist keine Abfrage, aber eine Antwort vom Empfänger zum ursprünglichen Sender, dass die Abfrage nicht bearbeitet werden kann.
Schlüssel:
+ Erlaubt
- Nicht erlaubt
] Geforderte Kombination

2.6 Ereignisnummer

Die Positionen 3 bis 8 jeder Zeile enthalten die Ereignisnummern dieser Zeile. Ereignisnummern können auf drei Arten verwendet werden, wie nachfolgend beschrieben.

Zentral registrierte Ereignisnummern. Diese sind vordefiniert und bereits vor dem Empfang allen Mitgliedern einer Nachrichtengruppe bekannt:

- jede registrierte Ereignisnummer ist eine Vorlage für eine bestimmte Gruppe von DDI-Nummern in einer spezifizierten Reihenfolge;
- jede registrierte Ereignisnummer ist ein Ereignis für eine bestimmte Minimal/Maximalgruppe von DDI-Nummern ohne eine spezifizierte Reihenfolge.

Die Ereignisnummern werden vom System beim Schreiben einer ADIS-Datei erstellt. Die Ereignisnummer hat keine spezifizierte Bedeutung. Sie kann zur Identifizierung unterschiedlicher Datenblöcke verwendet werden.

Die Ereignisnummer hat überhaupt keine Bedeutung: ihr Wert ist "000000". Ereignisnummern, die mit einer Null beginnen, können frei von Herstellern verwendet werden. Ereignisnummern, die mit einer Null beginnen, können genormt werden. Ereignisnummern können in der selben Weise wie DDI-Nummern genormt werden.

Länge einer Ereignisnummer Eine Ereignisnummer ist ein numerischer Wert und hat eine festgelegte Länge von 6 Stellen.

Verknüpfen von V-Zeilen mit D-Zeilen Die Ereignisnummer einer D-Zeile soll die selbe sein wie die Ereignisnummern der zugehörigen folgenden V-Zeilen. Wenn sich die Ereignisnummer einer V-Zeile von der Ereignisnummer der vorangehenden D-Zeile unterscheidet, dann ist die V-Zeile unzulässig.

Ereignisnummern in S- und R-Zeilen

Die Ereignisnummer einer R-Zeile soll die selbe sein wie die Ereignisnummern der zugehörigen vorangehenden S-Zeilen (falls es vorangehende S-Zeilen gibt). Wenn sich die Ereignisnummern einer R- und S-Zeile voneinander unterscheiden, ist die Anfrage nicht möglich. Die empfangende Partei kann die Ereignisnummer einer R-Zeile in ihrer Antwort verwenden.

2.7 Abfrage

Jede Anfrage besteht aus einer oder mehreren S-Zeilen (wahlweise) und einer R-Zeile (obligatorisch). Die S-Zeilen beinhalten die Suchbedingungen, die R-Zeile beinhaltet die DDI-Nummern der Datenelemente, die die angefragten Daten beinhalten. Eine S-Zeile kann nur Bedingungen enthalten, die sich auf Schlüsseldata beziehen, die sich auf den Ereignisschlüssel beziehen. DDI-Nummern der S-Zeile sollen sich auf Schlüsseldata beziehen.

ANMERKUNG – Die Umsetzung von Abfrage-Bearbeitungsprogrammen, die mit ADIS-Dateien bearbeitet werden, ist nicht obligatorisch. Da Suchbedingungen im höchsten Maße von Schlüsseldata abhängig sind (bekannt im Data Dictionary), kann die Antwort auf eine Abfrage umsetzungsabhängig sein.

```
[(S<Status><Ereignis-Nr.>(<DDI-Nr.><Feldlänge><Auflösung><Niedrigster  
Wert><Höchster Wert>)<Max. Anzahl von Aufzeichnungen><CR><LF>)]  
R<Status><Ereignis-Nr.>(<DDI-Nr.><Feldlänge><Auflösung>)<CR><LF>
```

Es ist nicht zulässig, S/R-Zeilen zwischen die dazugehörigen D- und V-Zeilen zu setzen. Die S-Zeilen (wenn vorhanden) folgen immer sofort der R-Zeile. Lediglich C-Zeilen können, ungeachtet ihres Status, in einer Gruppe von Abfrage-Zeilen plaziert werden. Es wird empfohlen, Abfragen von D- und V-Zeilen durch Verwendung einer T-Zeile oder durch Setzen der Abfragen in separate Dateien zu trennen. Wenn eine Abfrage eine unzulässige S- oder R-Zeile enthält, ist die ganze Abfrage unzulässig.

Suchbedingungen einer Abfrage:

[(S<Status><Ereignis-Nr.>(<DDI-Nr.><Feldlänge><Auflösung><Niedrigster Wert><Höchster Wert>)<Max. Anzahl von Aufzeichnungen><CR><LF>)]

Bedenken Sie, daß eine S-Zeile mehrere Suchbedingungen enthalten kann. Diese Bedingungen sind unbedingt mit dem logischen AND-Operator verbunden. Wenn eine Abfrage mehr als eine S-Zeile enthält, sind die Bedingungen in dieser S-Zeile ebenfalls unbedingt mit einem logischen AND-Operator verbunden. Jede Suchbedingung enthält einen unteren Wert und einen oberen Wert, d.h., daß eine Suche für einen Bereich getätigt werden kann:

Unterer_Wert<=Wert_im_Feld_mit_DDI-Nr.<=Oberer_Wert

Das Format von *<niedrigster Wert>* und *<höchster Wert>* soll das selbe sein wie das für *<Feldlänge>* und *<Auflösung>*. Die Feldlänge für *<Max. Anzahl von Aufzeichnungen>* ist festgelegt auf 6 Zeichen (ausgefüllt mit Zwischenräumen, wenn die Felder keinen Wert haben). *<Max. Anzahl von Aufzeichnungen>* spezifiziert die maximale Anzahl von Aufzeichnungen, die als Antwort auf die in einer S-Zeile plazierten Bedingungen abgegeben werden können.

Wenn *<Max. Anzahl von Aufzeichnungen>* keinen Wert hat, gibt es keine Einschränkung der maximalen Anzahl von Aufzeichnungen, die für die vorhergehende(n) Bedingung(en) abgegeben werden können.

Um die ersten n Aufzeichnungen zu suchen, werden die *<niedrigster Wert>* Felder vollständig mit Fragezeichen ("??") und die *<höchster Wert>* Felder haben keinen Wert: sie sind vollständig mit Zwischenräumen gefüllt.

S<Status><Ereignis-Nr.>(<DDI-Nr.><Feldlänge><Auflösung><(?)><(-)>)<Max. Anzahl von Aufzeichnungen><CR><LF>

in der "maximale Anzahl von Aufzeichnungen" den Wert n hat. Um die letzten m Aufzeichnungen zu suchen, haben die *<niedrigster Wert>* Felder keinen Wert, sie sind vollständig mit Zwischenräumen ausgefüllt und die *<höchster Wert>* Felder sind vollständig mit Fragezeichen ausgefüllt ("??").

S<Status><Ereignis-Nr.>(<DDI-Nr.><Feldlänge><Auflösung><(-)><(?)>)<Max. Anzahl von Aufzeichnungen><CR><LF>

in der "maximale Anzahl von Aufzeichnungen" den Wert m hat. Wenn eine S-Zeile gemischte Grenzen enthält (für eine DDI-Nummer ist das ? im "niedrigster Wert"-Feld gesetzt, für andere ist das ? im "höchster Wert"-Feld gesetzt), ist die S-Zeile unzulässig. Wenn die Abfrage keine S-Zeilen enthält, gibt es keine Suchbedingungen, was bedeutet, daß die Abfrage eine Anfrage für alle in der R-Zeile genannten Daten ist.

R-Zeile einer Abfrage:

R<Status><Ereignis-Nr.>(<DDI-Nr.><Feldlänge><Auflösung>)<CR><LF>

2.8 Feldlänge und Auflösung

Auf jede DDI-Nummer, die in einer ADIS-Datei auftaucht, folgt immer sofort die Feldlänge und die Auflösung (Anzahl von Dezimalstellen) dieser DDI-Nummer:

<DDI-Nr.><Feldlänge><Auflösung>

Der Empfänger soll mindestens das Format des entsprechenden Data Dictionaries unterstützen. Es ist nicht obligatorisch, daß der Empfänger das nicht genormte Format unterstützt (unterschiedliche Feldlänge und/oder Auflösung). Die Verwendung einer Feldlänge und Auflösung für ein Feld in der ADIS-Datei, die dem Format in dem Data Dictionary gleich sind, wird empfohlen.

Es ist zulässig, eine Feldlänge oder Auflösung für ein Feld in dieser ADIS-Datei zu verwenden, die sich vom Format in dem Data Dictionary unterscheidet (von der Nachrichtengruppe abgestimmt); es wird empfohlen, die Verwendung von Feldlängen und/oder Auflösungen in dieser Weise niedrig zu halten.

Die Feldlänge definiert die Länge eines Datenfeldes. Die Feldlänge kann verwendet werden, um eine DDI-Nummer zu überspringen, wenn sie im Data Dictionary nicht erkannt wird. Dies kann vorkommen, wenn der Sender und der Empfänger nicht die selbe Data Dictionary-Version verwenden. Sie hat eine festgelegte Länge von 2 Stellen. Die Feldlänge ist eine Nummer im Bereich zwischen 1 und 99.

Die Auflösung definiert die Anzahl von Dezimalstellen eines numerischen Wertes. Sie hat eine festgelegte Länge von einer Stelle. Die Auflösung ist eine Nummer im Bereich zwischen 0 und 9. Alphanumerische Werte haben eine Auflösung von 0.

2.9 Rückstellung von Nachrichteneinstellungen

Alle Einstellungen eines Programms, das eine ADIS-Datei liest, werden zurückgestellt, sobald die Header-Zeile (H) auftaucht.

2.10 DDI-Nummer

DDI-Nummern haben eine festgelegte Länge von 8 Zeichen.

Dies kann z.B. sein

- 2 vorangestellte Nullen plus sechsstellige numerische DD-Nummer;
- 8 Zeichen hexadezimale Identifier-Nummer;
- 8 Zeichen alphanumerischer Code.

2.11 Datenfelder

Werte mit Vorzeichen Negative Werte von Feldern sind mit einem Minuszeichen vor dem ersten gültigen Zeichen eines Wertes gekennzeichnet. Positive Werte haben kein

Pluszeichen vor dem ersten gültigen Zeichen. Die Feldlänge ist die selbe. Datenfelder ausfüllen Fehler von numerischen Feldern haben vorangestellte Leerzeichen, um die Felder auszufüllen, wenn das Feld weniger Zeichen enthält als die spezifizierte Feldlänge. In speziellen Situationen können nach Vereinbarung vorangestellte Nullen verwendet werden.

Nullwerte Felder, die keinerlei Wert enthalten (nicht initialisierte Felder), sollen vollständig mit Fragezeichen gefüllt werden ("?", dezimal ASCII 63).

Dezimalzeichen

Dezimalzeichen (Punkt, Komma) sind nicht in den Wertfeldern der ADIS-Datei dargestellt.

2.12 Daten und Zeitformat

Daten werden über 8 Positionen dargestellt:

CCYYMMDD

Zeitfelder haben voreingestellt ein sechsstelliges Format:

HHMMSS

Daten und Zeit haben immer vorangestellte Nullen (anstelle von vorangestellten Zwischenräumen).

000000<=Zeit<=235959
(Mitternacht: 000000, Mittag: 120000)

Siehe ISO 8601.

2.13 Kontrollsummen

ADIS-Dateien können Kontrollsummen enthalten, dies ist aber nicht obligatorisch. Kontrollsummen können nur in V-Zeilen gesetzt werden. Eine Kontrollsumme wird als normales Data Dictionary-Feld betrachtet. Dies bedeutet, daß die D-Zeile eine DDI-Nummer für die Kontrollsumme enthalten soll.

ANMERKUNG – Dieser Weg, die Kontrollsummen zu verwenden, hat eine andere OSI-Ebene als z.B. die Kontrollsummen, die vom Protokoll auf den niedrigeren OSI-Ebenen versendet werden. Die Position einer Kontrollsumme in einer V-Zeile hängt vollständig von der Position der DDI-Nummer für die Kontrollsumme in der entsprechenden D-Zeile ab.

Die Berechnungsmethode der Kontrollsumme ist auch in dem Data Dictionary beschrieben, in dem die DDI-Nummer der "Kontrollsumme" gespeichert ist.

BEISPIEL: Der Modulo-10-Check wird plziert. Jedes Zeichen vor der Kontrollsumme wird in die Berechnung der Kontrollsumme mit einbezogen. Wenn die Kontrollsumme auf der Position n+1 plziert ist, wird wie folgt berechnet:

$$\left\{ \sum_{x=1}^n [ASC(\text{Zeichen auf Position } x)] \right\} MOD 10$$

2.14 Unbekannte DDI-Nummern

Wenn ein System eine Datenanfrage mit einer unbekanntem DDI-Nummer erhalt, wird die Antwortdatei das Wertefeld, das zu dieser DDI-Nummer gehort, mit einem senkrechten Zeilensymbol ausfullen ("I").

2.15 Kommentar

Kommentar-Zeilen sind uberall in der Datei erlaubt. Kommentare konnen in verschiedenen Status-Zeilen plziert werden, um mehr Informationen zu geben, z.B. warum der Status nicht "N" ist.

2.16 Dateistruktur

ADIS erlaubt nicht die Platzierung der verschiedenen Zeilentypen in beliebiger Reihenfolge. Das Bild zeigt in welcher Reihenfolge Zeilen zu plazieren sind. Dieses Diagramm soll von links nach rechts gelesen werden, außer, wenn dies anderweitig spezifiziert ist. Es gibt keine Regel für die senkrechte Richtung, ausgenommen für die Fälle, bei denen ein "oder" verwendet wird. Im oberen Teil der Abbildung sind die möglichen D- und V-Zeilen gezeigt.

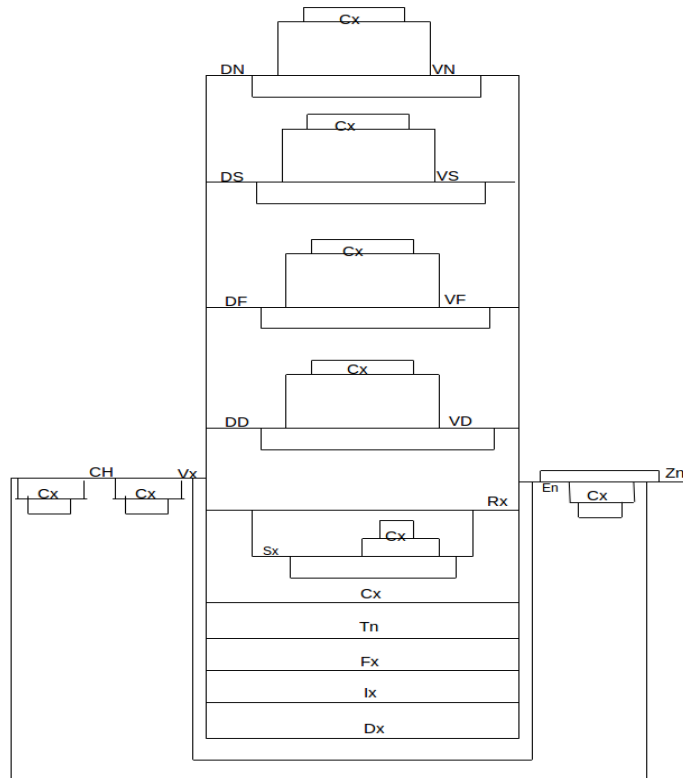


Abbildung 2.1: Anmerkung: Das "x" steht für das Statuszeichen im Falle, daß mehr als ein Status möglich ist.

3 Beschreibung ADED

3.1 Allgemein

Die landwirtschaftlichen Datenelemente, die in dem Data Dictionary ADED beschrieben werden, beziehen sich in erster Linie auf den Datenaustausch zwischen stationären Prozeßrechnern und Management-Computern. Dies bedeutet aber nicht, daß die Datenelemente nicht auch für andere Ebenen des Datenaustausches genutzt werden können, wie z.B. zwischen Management-Computer und einem externen Computer oder zwischen voneinander unabhängigen Software-Anwendungen auf dem Management-Computer. Grundvoraussetzung für den Datenaustausch ist, daß sowohl der Sender als auch der Empfänger der Daten mit der gleichen Version des ADED Data Dictionaries arbeiten.

- **Stufe 1:** Die Datenelemente sind zentral registriert auf einer weltweiten Basis (ISO). Die ISO-standardisierten Datenelemente haben eine "9" an der ersten Position der Datenelement-Nummer (DD-Nr.).
- **Stufe 2:** Die Datenelemente sind zentral registriert auf einer nationalen Basis. Die nationalen Datenelemente haben eine "1" bis "8" an der ersten Position der DD-Nr.
- **Stufe 3:** Die Datenelemente sind herstellerspezifisch. Die herstellerspezifischen Datenelemente sind durch eine "0" an der ersten Position der DD-Nr. gekennzeichnet. Dieser Bereich an "freien Nummern" kann für herstellerspezifische Tests benutzt werden. Herstellern wird geraten, eine registrierte Nummer zu beantragen, um die herstellerspezifische Nummer zu ersetzen, sobald das Objekt in einer offiziellen Softwareausgabe umgesetzt ist.

3.2 Datenelemente

Die folgenden Eigenschaften gelten für die internationalen Datenelemente:

der Identifier eines Datenelementes ist eine sechsstellige DD-Nummer, auch ADED-Nummer genannt;

65 Positionen stehen für den Namen des Datenelementes zur Verfügung;

der Datentyp eines Datenelementes kann "alphanumerisch" oder "numerisch" sein. Der Datentyp "signiert" wird nicht verwendet. Anstelle von "signiert" wird "numerisch" verwendet. Das bedeutet, daß eine der verfügbaren Positionen der Formatlänge eines Elementes für das Minuszeichen verwendet wird;

im DD werden ISO-Einheiten verwendet;

Datenelemente verwenden erweiterte ASCII-Zeichen (ISO 8-bit Code);
 Datenelemente sind Teil von mindestens einem Datenobjekt.

Die folgenden Eigenschaften sind für jedes Datenelement beschrieben:

k	ADED-Nummer: die Nummerierung der ADED-Datenobjekte beginnt mit 900001, ausgenommen des Datenobjektes "Data Dictionary Typ", das die ADED-Nummer 000 000 hat. Die Nummerierung wird als laufende Nummer ohne irgend eine andere logische Reihenfolge fortgesetzt.
m	Name
o	Synonyme
m	Definition
o	Kommentare
m	Format: a) Datentyp (alphanumerisch AN, numerisch N) b) Feldlänge oder Gesamtzahl der Stellen (falls numerisch) c) Auflösung (gleich der Anzahl der Dezimalstellen)
m	Einheit
o	Werte (Codesätze, Minimumwerte, Maximumwerte)
Schlüssel: k: key m: mandatory o: optional	

3.3 Datenobjekt(Entity)

Innerhalb des Kommunikationsbereiches ist ein Satz von Datenelementen, der eine logische Gruppe bildet, als ein Entity (im folgenden als Datenobjekt bezeichnet) für EDI definiert. Er besteht aus obligatorischen Schlüsselementen und anderen optionalen Datenelementen.

Wie bei Datenelementen gibt es unterschiedliche Normungsebenen der Datenobjekte:

Die ersten beiden Stellen von zentral registrierten Datenobjekten auf internationaler Ebene sind "99". Diese sind im voraus definiert und bereits vor dem Empfang allen Mitgliedern einer Kommunikationsgruppe bekannt. *ANMERKUNG* – Datenobjektnummern beginnen mit 99, um Verwechslung mit den DD-Nummern zu vermeiden.

Die erste Stelle von zentral registrierten Datenobjekten auf nationaler Ebene ist eine Nummer von "1" bis "8". Diese Nummern sind vordefiniert und bereits vor dem Empfang allen Mitgliedern einer Kommunikationsgruppe bekannt.

Die Datenobjektnummern werden vom System beim Schreiben einer ADIS-Datei erzeugt. Die Datenobjektnummer hat keine spezielle Bedeutung. Sie kann verwendet werden, um unterschiedliche Datenblöcke zu identifizieren. In diesem Fall beginnt die Datenobjektnummer mit "0". Deshalb können Hersteller mit "0" beginnende Datenobjektnummern frei verwenden.

Die Datenobjektnummer hat keine Bedeutung: ihr Wert ist "000000".

Jedes registrierte Datenobjekt ist als eine Gruppe von möglichen DD-Nummern beschrieben. Das Datenobjekt muß mit den Schlüsselementen beginnen. Die anderen Elemente sind optional und haben keine spezielle Reihenfolge. Nationale oder herstellerspezifische Datenelemente können einem registrierten Datenobjekt zugefügt werden.

Wenn ein Datenelement übertragen wird ist es notwendig, die Schlüsselemente des Datenobjektes, zu dem es gehört, zu übertragen. Der Satz von Schlüsselementen (Indexbaum) ist nicht in der Beschreibung der Datenelemente enthalten, kann aber in der Beschreibung der Datenobjekte gefunden werden.

Die folgenden Eigenschaften sind für jedes Datenobjekt beschrieben:

mk	Datenobjektnummer: die Nummerierung der ADED-Datenobjekte beginnt mit 990001. Die Nummerierung wird als laufende Nummer ohne irgend eine andere logische Reihenfolge fortgesetzt.
m	Definition
k	Schlüsselemente
o	Datenelemente: a) Datentyp (alphanumerisch AN, numerisch N) b) Feldlänge oder Gesamtzahl der Stellen (falls numerisch) c) Auflösung (gleich der Anzahl der Dezimalstellen)
m	Einheit
o	Werte (Codesätze, Minimumwerte, Maximumwerte)
Schlüssel: k: key m: mandatory o: optional	

3.4 Header

Der Header ist ein spezielles Datenobjekt, das keine Schlüsselemente hat und obligatorische (mandatory) Datenelemente enthält, die keine Schlüsselemente sind.

Jeder ADIS-File beginnt mit den Header-Zeilen, die folgende Datenelemente enthalten:

	DD Nr.	Inhalt	Format	Einheit	Anmerkung/Beispiel
m	000000	Data Dictionary Typ	8.0		"DD:" zeigt an, daß es sich um ein Data Dictionary mit 6-stelligen, numerischen DD-Nummern handelt. Im ADIS-File sind dann jeder DD-Nummer zwei Nullen "00" vorangestellt. "ID:" zeigt an, daß es sich um ein Data Dictionary mit 8-stelligen Identifiern handelt (verwendet beim LBS nach DIN 9684-5 und ISO 11783-11)
m	900002	ADED ISO Version	8.0		1996
m	900003	Datum, an dem der File erstellt wurde	8.0	CCJJMMTT	19970608
m	900004	Uhrzeit, zu der der File erstellt wurde	6.0	HHMMSS	152730
m	900006	Name des Senders	24.0		VIT Verden
m	900009	ADED Nationale Version	8.0		"AGRO2009" lautet der Name des nationalen Data Dictionaries in Deutschland, das für den Datenaustausch mit den Landeskontrollverbänden verwendet wird.
o	900005	System Status	1.0		
o	900007	Name des Empfängers	24.0		
o	900008	Version des Senders	8.0		
o	90001	Checksumme	1.0		
o	900011	Prozeß Computer Typ	10.0		
0	900012	ADED Hersteller Version	8.0		

Eine Beschreibung der einzelnen Datenelemente erhalten Sie, wenn Sie die Datenbank aufrufen.

4 Deutsches Data Dictionary

4.1 Anwendungsbereich

Zweck des Deutschen Data Dictionaries ist es, den Datenaustausch zwischen EDV-Systemen auf dem Gebiet der Rinder- bzw. Milchproduktion (für den Bereich der Schweineproduktion sind entsprechende Festlegungen in Vorbereitung) zu vereinheitlichen.

Diese Beschreibung gilt für den Austausch von ASCI-Dateien zwischen:

- Systemen zur Anlagensteuerung und zur automatischen oder manuellen Datenerfassung auf dem Betrieb (z.B. Melk- und Fütterungsanlagen);
- Management- und Berechnungssystemen auf dem landwirtschaftlichen Betrieb (z.B. Programme zur Herdenführung und Futterberechnung);
- Systemen der Anbieter von Serviceleistungen (z.B. Kontrollverbände, Zuchtorganisationen und tierärztliche Dienste).

Der Datenaustausch kann uni- oder bidirektional zwischen allen oben genannten Systemen erfolgen. Wie die einzelnen Datenelemente in den verschiedenen Systemen dargestellt und verarbeitet werden, ist nicht Gegenstand dieser Beschreibung.

4.2 Durchführen des Datenaustausches

4.2.1 Allgemeines

Die Details des Datenaustauschverfahrens werden zwischen den Anbietern der oben genannten Systeme vereinbart. Verschiedene Anbietergruppen sollten möglichst über eine Dachorganisation eine einheitliche Schnittstelle definieren.

Die anzuwendende Syntax zum Datenaustausch ist in ISO 11787:1995 beschrieben. Folgende Zeilentypen müssen mindestens unterstützt werden:

unidirektional C, D, E, V, Z

bidirektional C, D, E, R, S, V, Z.

4.2.2 Standarverzeichnis und Dateiname

Für den Datenaustausch wird das Verzeichnis ADED_XXX benutzt. XXX ist eine herstellereigene Kennung. In diesem Verzeichnis müssen sich alle für den Datentransfer benötigten Daten-, Programm- und Systemdateien befinden.

Bei der Datenbereitstellung gemäß den ADR-Richtlinien soll der Dateiname ADR.ADS verwendet werden.

4.2.3 Interfaceprogramm

Das zur Automatisierung des Datentransfers benötigte Programm sollte EDL_XXX genannt werden. XXX ist die gleiche herstellereigene Kennung wie die des Standardverzeichnisses. EDL_XXX muß folgende Parameter unterstützen:

EDL_XXX -i <infilename> -o <outfilename> -p <portnumber> -d <dumpfile> -b<baudrate> -e<errorfile>

Parameter	Benennung	Datentransferprogramm;Beschreibung
-i	Infilename	Importfile für das zu EDL_XXX gehörende System; Importdaten für das Programm EDL_XXX. Das können Daten und/oder/auch ein Request sein.
-o	Outfilename	Exportfile für das zu EDL_XXX gehörende System; Exportdaten aus dem Programm EDL_XXX. Daten, die aufgrund eines Requests erstellt wurden.
-p	Portnummer	Schnittstellenummer bei serieller Verbindung; Dieser Parameter wird nur bei seriellen Datentransfer unterstützt.
-d	Dumpfile	File mit allen unterstützten Datenelementen
-b	baudrate	Baudrate der Übertragung des EDL_XXX
-e	Errorfile	Datei mit den Fehlermeldungen des EDL_XXX

4.2.4 Dumbfile

Ein Dumpfile ist ein ADIS-File, in dem alle verfügbaren Entities und Datenelemente des Programmes EDL_XXX aufgeführt werden. Maximal sind dies alle Elemente des Data Dictionaries. Der Dumpfile enthält alle vom PRC unterstützten Entities mit einer Definition Line und einer Value Line.

Die Value Line wird dabei für jedes Datenelement mit ? dargestellt. Jeder Hersteller unterstützt das Data Dictionary entsprechend der im Dumpfile aufgeführten Elemente. Ein Dumpfile kann jederzeit erstellt werden, auch wenn das System keine Daten enthält.

4.2.5 Errorfile

Der Errorfile enthält Fehlermeldungen des EDL_XXX. Fehlermeldungen werden bei ADIS Formatverletzungen oder bei Datenfehlern generiert. Vor jedem Datentransfer muß ein bestehender Errorfile durch EDL_XXX gelöscht werden. Der Errorfile gibt somit immer Auskunft über Fehler des letzten Datentransfers.

4.3 Datenelemente

Für die sechsstelligen Datenelemente gelten die Regeln nach ISO 11788-1. Im Datenaustausch finden international und national genormte Datenelemente Verwendung. Die über die in ISO 11788-2 hinaus benötigten Datenelemente sind durch national festgelegte Datenelemente oder herstellerspezifische Datenelemente zu ergänzen.

Die nationalen Datenelemente beginnen mit der Ziffer 8. Die bisher festgelegten nationalen Datenelemente sind in DIN 11800 dargestellt. Neue Datenelemente werden nur dann in das nationale Data Dictionary aufgenommen, wenn sie nach ihrer Definition sich nicht durch andere Datenelemente darstellen lassen. Unterschiedliche Berechnungs- und Meßmethoden lassen sich durch zusätzliche Statusvariablen kennzeichnen.

Abweichungen hinsichtlich verwendeter Formate und Wertebereiche können Teil der Schnittstellenvereinbarung der Anbieter sein. Die in den Data Dictionaries aufgeführten Datenelemente müssen nur unterstützt werden, soweit sie im entsprechenden System tatsächlich vorhanden sind.

4.4 Entities (Datenobjekte)

Zum Austausch der Daten werden Entities definiert. In einer Entity können internationale, nationale und herstellerspezifische Datenelemente Verwendung finden.

Die bisher festgelegten nationalen Entities sind in DIN 11800 definiert.

4.5 Verwaltung des Data Dictionary

Die folgenden Organisationen sind in den genannten Ländern für die Verwaltung der nationalen Data Dictionaries zuständig.

Land	Organisation	Adresse
Kanada	Canadian Dairy Information Exchange (CDIE)	c/o Holstein Association of Canada P.O. Box 610, 170 Colborne Street Brantford Ontario N3T 5R4 Tel.: +1 519 756 8300 Fax: +1 519 756 5878
Frankreich	Ecole Nationale d'Ingénieurs des Travaux Agricoles (ENITA)	B.P. 201, 1, Cours du Général de Gaulle F-33175 GRADIGNAN CEDEX Tel.: +33 57 35 07 74 Fax: +33 57 35 07 89
Deutschland	Normengruppe Landtechnik (NLA)	Postfach 7108 64 D-60498 Frankfurt Tel.: +49 69 66 03 18 13 Fax: +49 69 66 03 14 64
Niederlande	Agricultural Telematics Centre	Agro Business Park, 58 NL - 8370 PW Wageningen Tel.: +31 7 47 96 58 Fax: +31 7 42 17 22

5 Beispiel

5.1 Datenblock

Der Datenblock für einen ADIS-Betrieb würde wie folgt beginnen:

```
DH99000100000000080009000020800090000308000900004060009000062400090000908000900012080
DN88000100800004150 VN880001 12345678
DN88000200800004150008001500800080015108000800019051008000200500080002103200800022030.....
VN880002 123456781996100119970215 352 6550412366.....
.
.
.
ZN
```

5.2 Abruf des Data Dictionarys mit einer Query vom ADISSERVER

5.2.1 Abfrage per Telnet

Beispiel: Abfrage des Data Dictionarys

Bei der Query zur Abfrage des DataDictionarys wird als Parameter das gewünschte DataDictionary angegeben.

Im Beispiel wird ADR2003 verwendet. (Die letzten sieben Zeichen in der Query-Zeile geben den Wert des Filters an, hier ADR2003. Andere Filterwerte sind hier möglich.)

Nur wenige Eingaben sind nötig, um Daten per Telnet abzurufen. (Sollte der abfragende Computer hinter einer Firewall stehen, so muß der PortZugriff auf den Port 1502 erlaubt sein.) Sie müssen nur die blauen Zeilen eingeben (per "Cut&Paste") um Daten beim AdisServer abzufragen.

-> Anmelden per Telnet:

```
Eingabe: telnet www.lkv-wl.de 1502
Antwort: Trying...
Antwort: Connected to www.lkv-wl.de
Antwort: Escape character is '^'.
-> Anmelden als Gast:
```


Eingabe: ?AN:BENUTZER=GAST;PASSWORT=GAST;PROJEKT=lkv;LESEDB=LKVDB.ADISINT;

ADISDB=LKVDB.ADISINT;

Antwort: Viel Text (Parameter, welche der Server übernommen hat)

Eingabe: DH990001000000000800090000208000900003080009000040600090000624000900009080

Antwort: CN ANMELDUNG AKZEPTIERT (und weitere Ausgaben)

Eingabe: VH990001DD 1997 20000418093453LKV Westfalen-Lippe e.V lkv

Antwort: VH990001DD: 1997 20030306073016LKW

tE

Eingabe: QN1900010088889150DDictionary 00190001080ADR2003

Antwort: Ergebnisdaten...

Eingabe: ZN

(unterer Textblock kann auch per Cut&Paste eingefügt werden.)

```
?AN:BENUTZER=GAST;PASSWORT=GAST;PROJEKT=lkv;LESEDB=LKVDB.ADISINT;ADISDB=LKVDB.ADISINT;
```

```
DH990001000000000800090000208000900003080009000040600090000624000900009080
```

```
VH990001DD 1997 20000418093453LKV Westfalen-Lippe e.V lkv
```

```
QN1900010088889150DDictionaryN 00190001080ADR2003
```

```
ZN
```

(Das Zeilenende hinter dem ZN und das Leerzeichen hinter dem Parameter bitte berücksichtigen.)

5.2.2 Benutzung eines Client Programms

Die Kommunikation mit dem Server kann auch mit einem ManagementProgramm oder dem vom LKV zur Verfügung gestellten Programm HofPC bzw. einem einfachen Client erfolgen.

Benutzung des Programms HofPC

Über dieses Programm können vom Server Daten aus den Bereichen Stammdaten, Kontrollen und Molkereigüte abgefragt und in diversen Formaten ausgegeben werden. Das Programm ist hier beschrieben.

Benutzung des ADIS Clients

Der Aufbau eines SimpleClients für Querys ist im Internet hier beschrieben. Dieser Client benutzt keine weiteren Klassen einer Standard-Javainstallation. Er wird aufgerufen mit

```
java micraAdisQClient
```

und verarbeitet obigen Anmeldestring und Queryanfrage, so dass als Ergebnis die DataDictionary Daten geschickt werden. Durch Austausch des Anmeldestrings bzw. des QueryStrings können individuelle Abfragen aufgerufen werden.

5.2.3 Generierung von XML aus ADS-Daten über den Server des LKV (Dienstleistung)

Eine Adis-Datei kann auch direkt dem Server zur Verarbeitung übergeben werden. Als Beispiel wird in dem Kapitel Verarbeitung von Adis-Dateien mit einem Client beschrieben, wie obige Ergebnisdatei in XML gewandelt wird.

6 ADIS/ADED Standards

Folgende Standards werden zur Anwendung der elektronischen Datenübertragung nach ADIS-ADED ganz oder teilweise benötigt:

ISO 11787	Machinery for agriculture and forestry – Data interchange between management computer and process computer – Data interchange syntax Land- und forstwirtschaftliche Maschinen - Datenaustausch zwischen Betriebs PC und Prozess Computer - Syntax für den Datenaustausch	Internationaler Standard seit 1995
ISO 11788-1	Electronic data interchange between information systems in agriculture – Agricultural data element dictionary – Part 1: General description Elektronischer Datenaustausch zwischen landwirtschaftlichen Informationssystemen – Data Dictionary für die Landwirtschaft – Teil 1: Allgemeine Beschreibung	Internationaler Standard seit 1997
ISO 11788-2	Electronic data interchange between information systems in agriculture – Agricultural data element dictionary – Part 2: Dairy farming Elektronischer Datenaustausch zwischen landwirtschaftlichen Informationssystemen – Data Dictionary für die Landwirtschaft – Teil 2: Milchviehhaltung	Internationaler Normentwurf seit 03/98
ISO 11788-3	Electronic data interchange between information systems in agriculture – Agricultural data element dictionary – Part 3: Pig Farming Elektronischer Datenaustausch zwischen landwirtschaftlichen Informationssystemen – Data Dictionary für die Landwirtschaft – Teil 3: Schweinehaltung	Internationaler Normentwurf erwartet für 07/98
DIN 11800	Landtechnik – Elektronischer Datenaustausch in der Landwirtschaft – Datenaustausch im Bereich der Rinder-Produktion Agricultural engineering – Electronic data interchange in agriculture – data interchange in the field of cattle production	Nationaler Normentwurf seit 01/98

Alle Normen und Normentwürfe sind zu beziehen über den Beuth Verlag GmbH; Burggrafstraße 6, D-10787 Berlin; Tel.: 030 26 01 22 60; Fax: 030 26 01 12 60; postmaster@beuth.de

Weitere Informationen über die weltweite Normenorganisation ISO (International Standards Organisation) gibt es unter www.iso.ch

Weitere Informationen über die nationale Normenorganisation DIN (Deutsches Institut für Normung e.V.) gibt es unter <http://www.din.de>

7 ADIS Header

7.1 Was gehört in einen ADIS-Header zum Datenaustausch nach DIN 11800?

M = Mandatory, O = Optional, C = Conditional

N = Numerisch, AN = Alphanumerisch

Die Feldlänge kann prinzipiell variieren, muß dann natürlich in der Definitionszeile mit der richtigen Länge angegeben werden.

7.1.1 Gemäß internationaler ISO-Norm:

DD-Nummer	Art	Format	Länge	Name	Erklärung
000 000	M	AN	8	DD(DataDictionary) Type	Bezeichner des DataDictionary Typs Für nationale ADIS-Daten immer "DD:"
900 002	M	AN	8	ADED ISO Version	Version der internationalen ISO-Norm des ADED Data Dictionary. Wird "von oben" (ISO) gegeben, d.h. internationale Festlegung. Zur Zeit: Die Jahressahl "1996"
900 003	M	N	8	File Creation / Update Date	Datum, wann die vorliegende Datei erzeugt oder das letzte geändert wurde. Format ccyymmdd, also zum Beispiel "19991224"
900 004	M	N	6	File Creation / Update Time	Uhrzeit, wann die vorliegende Datei erzeugt oder das als letzte geändert wurde. Format hhmmss, also zum Beispiel "230000"
900 005	O	AN	1	System Status	OK/Nicht OK gemäß CodeSet (welches zur Zeit nicht verfügbar ist)
900 006	M	AN	24	Sender Software Name	Name der Sender Software, nicht Adresse des Empfängers Beispiel: "Westfalia Landtechnik DP5"
900 007	O	AN	24	Receiver Software Name	Name der Empfänger Software, nicht Adresse des Empfängers. Beispiel: "LKV WESTFALEN-LIPPE E.V."
900 008	O	AN	9	Sender Software Version	Version der Software des Senders. Beispiel: "5.010"
900 009	C	AN	8	ADED National Version	Version der nationalen Norm des ADED Data Dictionary. Wird vom nationalen Gremium festgelegt Für Rinder vom ADR zur Zeit "ADR19991" Dieses Feld muss für ADIS Übertragung im Bereich des ADR unbedingt eingetragen sein.
900 010	O	N	1	Checksum	Modulo-n-Prüfsummenberechnung. Details siehe Internationales Data Dictionary
900 011	O	A	10	Process Computer Type	Bezeichner für Prozesscomputer. Könnte wichtig werden für die Auswahl Übertragungsart, Baudraten, etc.
900 012	C	AN	8	ADED Manufacturer Version	Der Sender kann eigene Entities und DD-Nummern (sollten dann jeweils Nummern mit weniger als 8 Stellen sein) verwenden. Dafür kann er ein eigenes ADED Data Dictionary in eigenen Versionen festlegen. Hier wird der Bezeichner dafür angegeben. Beispiel: "WLTDP50"

7.1.2 Gemäß nationaler Festlegung (DIN):

Folgende Zusätze:

DD-Nummer	Art	Format	Länge	Name	Erklärung
800 137	O	N	4	Serial Number	Mit der laufenden Nummer können einige Anwendungen die ausgetauschten DATen identifizieren und prüfen
800 622	O	AN	99	Sender Address	Name und ggf. Adresse des Senders, insbesondere ist E-mail-Adresse möglich
800 623	O	AN	99	Receiver Adress	Name und ggf. Adresse des Empfängers, insbesondere ist E-mail-Adresse möglich
830 077	O	N	2	Character Set	Code-Set für den verwendeten Zeichensatz 0=US_ASCII,1=ISO-8859-1,...9=ISO-8859-9
830 078	O	AN	3	Currency	ASCII-Bezeichnung der Währung gemäß ISO-4217, z.B. "DEM" für Deutsche Mark

7.2 Beispiel eines "Minimal-Headers"

(in den Kommentar-Zeilen stehen Bezeichner für die Felder bzw. machen Längen der Werte deutlich)

```
DH990001000000000800090000208000900003080009000040600090000624000900009080
CHEntityDD_____ADEDISO__Filedate__Filetime__Sender____ADEDNat____
CHEntityDD_____ADEDISO_FiledateFiletiSender_____ADEDNat_
VH990001DD: 1996 19991224230000WestfaliaDP5 ADR19911
```

8 Beschreibung des Clients

8.1 Mini Query-Client für Abfragen auf dem ADISSERVER des LKV Westfalen-Lippe.

Dieser Client verbindet sich über eine Socket-Verbindung mit dem AdisServer beim Landeskontrollverband Westfalen-Lippe und sendet einen Anmeldestring (beginnt mit ?AN) und dann den QueryString (beginnt mit QN).

Als Ergebnis sendet der Server die Daten. Es werden keine weiteren Klassen des LKVs benötigt. Dieses Programm ist mit einer Standard-JavaInstallation lauffähig.

Im Beispiel werden die Daten des DataDictionries abgefragt. Die Ausgaben erfolgen auf der Konsole und können mit folgendem Aufruf in eine Datei umgeleitet werden: java

microAdisQClient > DATEINAME.ads Der Quellcode:

```
/**
 * Title:      microAdisQClient.java
 * Description: micro Client showing how to cummunicate with AdisServer via SocketConnction
 * Copyright:   Copyright (c) 2003
 * Company:    Landeskontrollverband Westfalen-Lippe www.lkv-wl.de
 * @author     Andreas Glahn
 * @version    1.0
 */

import java.net.*;
import java.io.*;

public class microAdisQClient{
public static void main(String[] args){
Socket theSocket;
DataInputStream empfangsDatenStream;
PrintStream sendeDatenStream;
String Line;

try {
theSocket = new Socket("www.lkv-wl.de",1502);
empfangsDatenStream = new DataInputStream(theSocket.getInputStream());
sendeDatenStream = new PrintStream(theSocket.getOutputStream());
/* Anmeldung */
sendeDatenStream.println("?AN:BENUTZER=GAST;PASSWORT=GAST;PROJEKT=lkv;LESEDB=LKVDB.ADISINT;ADISDB=LKVDB.ADISINT;");
/* Aufruf der Query */
sendeDatenStream.println("QN1900010088889150DDictionaryN 00190001080ADR2003 ");

/* Ausgabe der Daten */
String eingeleseneZeile = null;
eingeleseneZeile = empfangsDatenStream.readLine();

while ((eingeleseneZeile != null)) {
System.out.println(eingeleseneZeile);
eingeleseneZeile=empfangsDatenStream.readLine();
}
```

```

if (eingelieseneZeile.startsWith("T")|| eingelieseneZeile.startsWith("tE" ) break;
}
/* Beenden der Verbindung */
sendeDatenStream.println("ZN");
System.out.println(empfangsDatenStream.readLine());
theSocket.close();
} catch (IOException e){
System.err.println(e);
}

}
}

```

8.2 ADIS-Dateien auf dem ADISSERVER des LKV Westfalen-Lippe verarbeiten (XML)

ADIS-Dateien auf dem ADISSERVER des LKV Westfalen-Lippe verarbeiten

Andreas Glahn Beschreibung eines Clients zur Umwandlung zu XML

Dieser Client verbindet sich über eine Socket-Verbindung mit dem AdisServer beim Landeskontrollverband Westfalen-Lippe und sendet einen Anmeldestring (beginnt mit ?AN) und dann die Zeilen einer AdisDatei (ADISDATEI.ads).

Im Anmeldestring wird als SCHREIBOBJEKT XML gewählt, wodurch der Server die Daten in XML zurückgibt. Weiterhin werden im Programm nur die Zeilen, welche mit XML beginnen ausgegeben (der Server schickt noch weitere Kommentarzeilen, die hier aber unterdrückt werden sollen).

Als Ergebnis sendet der Server die ErgebnisZeilen. Es werden keine weiteren Klassen des LKVs benötigt. Dieses Programm ist mit einer Standard-JavaInstalltion lauffähig. Im Beispiel werden die Daten aus dem Ergebnis der Query-Abfrage des DataDictionries verwendet.

Die Ausgaben erfolgen auf der Konsole und können mit folgendem Aufruf in eine Datei umgeleitet werden:

```
java microAdisClient > DATEINAME_XML.ads Der Quellcode:
```

```

/**
 * Title:      microAdisClient.java
 * Description: micro Client showing how to cummunicate with AdisServer via SocketConnction
 * Copyright:   Copyright (c) 2003
 * Company:    Landeskontrollverband Westfalen-Lippe   www.lkv-wl.de
 * @author     Andreas Glahn
 * @version    1.0
 */
import java.net.*;
import java.io.*;
public class microAdisClient{
    public static void main(String[] args){
        Socket theSocket;
        DataInputStream empfangsDatenStream;
        PrintStream sendeDatenStream;
        BufferedReader EingabeDatei = null;
        String Line;
        String EingabeZeile = "";
        String filename="ADISDATEI.ads";
        try {
            EingabeDatei = new BufferedReader(new FileReader(filename));
        } catch (IOException e){
            System.out.println("Fehler beim öffnen von " + filename+".");
        }
        try {
            theSocket = new Socket("www.lkv-wl.de",1502);
            empfangsDatenStream = new DataInputStream(theSocket.getInputStream());
            sendeDatenStream = new PrintStream(theSocket.getOutputStream());
            /* Anmeldung und Ausgabe nach XML definieren*/
            sendeDatenStream.println("?AN: BENUTZER=GAST;PASSWORT=GAST;PROJEKT=lkv;LESEDB=LKVDB.ADISINT;ADISDB=LKVDB.ADISINT;SCHREIBEN=XML;");
            /* ServerAntwort */
            String eingeleseneZeile = null;
            eingeleseneZeile = empfangsDatenStream.readLine();
            if (eingeleseneZeile != null){
                /* AnmeldeZeile soll nicht ausgegeben werden */
                // System.out.println(eingeleseneZeile);
            }
            /* XML Header schreiben */
            // hier evtl. Beginn und Ende des String mit spitzen Klammern versehen
            System.out.println("<?xml version='1.0' encoding='ISO-8859-1'?>");
            System.out.println("<LKVAdisXML>");
            /* Die einzelnen Zeilen der Eingabedatei an den Server übergeben und die ErgebnissZeilen abfragen */
            while (EingabeZeile != null){
                try{
                    EingabeZeile = EingabeDatei.readLine();
                    /* Zeile an den Server übregeben */
                    sendeDatenStream.println(EingabeZeile);
                    /* Daten empfangen */
                    boolean transakt_beendet = false;
                    while (!transakt_beendet){
                        eingeleseneZeile=empfangsDatenStream.readLine();
                        /* Filtern auf XML Rückgabe */
                        if
                        (eingeleseneZeile.startsWith("XML")) {
                            // XML: nicht mitausgeben, deshalb erst ab Position 4
                            System.out.println(eingeleseneZeile.substring(4));
                        }
                        if (eingeleseneZeile != null){
                            if (eingeleseneZeile.startsWith("T")|| eingeleseneZeile.startsWith("tE") ) transakt_beendet = true;
                            if (eingeleseneZeile.startsWith("ZN")|| eingeleseneZeile.startsWith("null") ) transakt_beendet = true;
                            if (eingeleseneZeile.equals("tN")) eingeleseneZeile = "TN";
                        } else {
                            transakt_beendet = true;
                        }
                    }
                }
                catch (IOException e){
                    System.out.println("Fehler beim lesen der Datei bzw. bei der Datenübertragung.");
                }
            }
            /* XML-Datei beenden */
            // hier evtl. Beginn und Ende des String mit spitzen Klammern versehen
            System.out.println("</LKVAdisXML>");
            /* Beenden der Verbindung */
            sendeDatenStream.println("ZN");
            theSocket.close();
        } catch (IOException e){
            System.err.println(e);
        }
        try{
            EingabeDatei.close();
        } catch (IOException e){
            System.out.println("Fehler beim Schließen der Datei.");
        }
    }
}

```