



Deutsch

FUJITSU Software BS2000

CIS V12.0 Manual 1

Einführung

Benutzerhandbuch

April 2020

Kritik... Anregungen... Korrekturen...

Die Redaktion ist interessiert an Ihren Kommentaren zu diesem Handbuch. Ihre Rückmeldungen helfen uns, die Dokumentation zu optimieren und auf Ihre Wünsche und Bedürfnisse abzustimmen.

Sie können uns Ihre Kommentare per E-Mail an manuals@ts.fujitsu.com senden.

Zertifizierte Dokumentation nach DIN EN ISO 9001:2015

Um eine gleichbleibend hohe Qualität und Anwenderfreundlichkeit zu gewährleisten, wurde diese Dokumentation nach den Vorgaben eines Qualitätsmanagementsystems erstellt, welches die Forderungen der DIN EN ISO 9001:2015 erfüllt.

Copyright und Handelsmarken

Copyright © 2020 Fujitsu Technology Solutions GmbH.

Alle Rechte vorbehalten. Liefermöglichkeiten und technische Änderungen vorbehalten.

Alle verwendeten Hard- und Softwarenamen sind Handelsnamen und/oder Warenzeichen der jeweiligen Hersteller.

Inhaltsverzeichnis

1 Einführung für den Benutzer von CIS	5
1.1 Allgemeines	5
1.2 Client/Server Architektur.....	6
1.2.1 Grafische Benutzeroberfläche unter MS-Windows	6
1.2.2 4-GL Anwendungsentwicklung.....	6
1.3 Die Grundstruktur des Datenbankkonzepts	7
1.4 Die Grundstruktur des CIS-Kommunikationssystems	9
1.5 Datenschutz.....	10
1.6 Datensicherung	11
1.7 Dienstprogramme	12
2 Dateien in der CIS-Datenbank	13
2.1 Hauptdateien	13
2.1.1 Abschnitt	14
2.1.2 Felder.....	14
2.1.3 Wiederholabschnitte	15
2.1.4 Wiederholfelder.....	15
2.1.5 Beispiel eines multivariablen (MV-) Satzaufbaus	16
2.2 Datenbeschreibungsdatei.....	17
2.2.1 Satzbeschreibungen.....	19
2.2.2 Transformations- und Bildbeschreibung.....	23
2.3 Verweisdatei	24
2.4 CIS-fremde Dateien.....	26
2.4.1 Zugriff auf SAM-und PAM-Dateien	26
2.4.2 Zugriff auf LEASY-Dateien	27
3 Datenbankzugriffe	29
3.1 Bereitstellung von Zielpunktlisten (ZPL).....	29
3.1.1 Zielpunktlistenerstellung mit Suchfragen.....	29
3.1.2 Aktivieren von Zielpunktlisten	30
3.1.3 Sortieren von Zielpunktlisten	30
3.2 Dateiverknüpfung	31
4 Einrichten und Verwalten von CIS-Datenbanken.....	33
4.1 Erstellen einer Datenbeschreibungsdatei	33
4.2 Verwalten der Datenbeschreibungsdatei	34
4.3 Erstellen einer Hauptdatei	35
4.4 Erstellen einer Verweisdatei.....	36
4.5 Einrichten der Datensicherungsdateien	37
4.6 Entscheidungskriterien	37
4.6.1 Entscheidungskriterien für Suchwege, Länge invertierter Felder und Datenbankschlüssel.....	37
4.6.2 Entscheidungskriterien für das Satzformat.....	39
5 Installation CIS	41
5.1 Varianteneinteilung in CIS V12.0	41
5.2 Versionsabhängigkeiten CIS-BS2000	42
5.3 Versionsabhängigkeiten CIS-UTM	42
5.4 Übernahme der Dateien	43
5.5 Installation der Datensicherung.....	46
6 Begriffsbestimmungen.....	51

1 Einführung für den Benutzer von CIS

1.1 Allgemeines

CIS ist ein dialogorientiertes Datenbank- und Informationssystem. Es ist für DV-Anwendungen entwickelt, die im Timesharing-Betrieb ablaufen. CIS ist unter dem Timesharing-Betriebssystem BS2000 anwendungsneutral einzusetzen. Es arbeitet sowohl im Teilnehmer- als auch im Teilhabermodus.

CIS-V12 setzt mindestens BS2000 V10.0 voraus.

CIS-V12 besitzt die volle XS-Fähigkeit, d. h. CIS unterstützt die volle Adreßbreite von 31-Bit Maschinen und kann damit auch im oberen Adreßraum ablaufen. Damit können insbesondere große TP-Anwendungen realisiert werden.

Die Dateien sind in CIS durch die indexsequentielle Zugriffsmethode ISAM organisiert. Hierarchische Datenstrukturen können abgebildet werden. Auf andere Dateiformate wie, LEASY, SAM oder PAM, kann lesend zugegriffen werden.

Das Datenbank- und Informationssystem CIS besteht aus zwei Hauptkomponenten, der

Datenbankkomponente und dem
CIS-Kommunikationssystem.

- Zur Datenbankkomponente gehören die in der Datenbank gespeicherten Daten, die Formatdefinitionen, unter denen die Daten gespeichert sind, sowie Steuerangaben, die für oft benötigte Daten einen beschleunigten Zugriff ermöglichen.
- Das CIS-Kommunikationssystem ist für die Einspeicherung und die Wiedergewinnung der Daten verantwortlich. Mit Hilfe von Kommandos können vielfältige Funktionen zur Speicherung und zum Abruf von Daten ausgeführt werden.

Der Datenbankszugriff im BS2000-Rechnernetz wird ebenso wie der synchronisierten CIS/UTM-Betrieb vom unabhängigen Date-Base-Handler CISDBH organisiert.

1.2 Client/Server Architektur

CIS kann wahlweise als Client/Server Version eingesetzt werden:

- Clients können zur Zeit die Systeme MXi oder RM unter SINIX oder PC's unter MS-Windows sein.
- CIS unter BS2000 dient als DB-Server.

Die Verbindung zu den Clients ist mit dem Verbindungsmodul CISDC für die oben genannten Rechner realisiert.

Auf dieser Schnittstelle können die Anwendungsprogramme mit diversen Programmiersprachen unter SINIX bzw. MS-Windows erstellt werden.

1.2.1 Grafische Benutzeroberfläche unter MS-Windows

Die grafische Benutzeroberfläche unter MS-Windows kann mit allen linearen Datenbeschreibungen arbeiten. Es sind sowohl Updatefunktionen in CIS als auch die Datenübergabe über die Zwischenablage an andere PC-Programme möglich. Somit können Daten z.B. in MS-Excel, MS-Word etc. weiterverarbeitet werden.

Die Oberfläche ist so gestaltet, daß auch Anwender ohne spezifisches CIS-Know-how mit CIS problemlos arbeiten können.

1.2.2 4-GL Anwendungsentwicklung

Für die unter 1.2 genannten Betriebssysteme kann das 4-GL Tool UnifAce von der Firma Uniface Deutschland GmbH für die Anwendungsentwicklung eingesetzt werden.

CIS ist ein vollwertiges Datenhaltungssystem für 4GL-Anwendungen unter UNIFACE. Auf vorhandene CIS-Daten kann ohne Einschränkungen zugegriffen werden.

UNIFACE ist ein Client/Server-Entwicklungswerkzeug der modernsten Generation, das alle grundlegenden Elemente für die Entwicklung unternehmensweiter Client/Server-Anwendungen beinhaltet. Hierzu gehören im einzelnen:

- Modellbasierte Entwicklung für die produktive Erstellung und Pflege von Anwendungen.
- Grafische, objektbasierte Umgebung für benutzerfreundliche Anwendungen.
- Technologieunabhängige Implementierung für offene, skalierbare Anwendungen.
- Teamentwicklung für mehr Produktivität in der Gruppe und bei Großprojekten.

Die Voraussetzung für die Anbindung von UNIFACE unter SINIX bzw. MS-Windows an CIS ist der Treiber CISDRX für SINIX bzw. CISDRW für MS-Windows.

1.3 Die Grundstruktur des Datenbankkonzepts

Die Datenbankkomponente von CIS besteht aus folgenden Dateitypen:

Hauptdatei	(HD)
Verweisdatei	(VD) und
Datenbeschreibungsdatei	(DABEL).

1. Hauptdatei

Die in der Datenbank enthaltenen Informationen sind in einer Hauptdatei gespeichert. Mehrere Hauptdateien können simultan bearbeitet werden. Eine Hauptdatei (HD) ist in verschiedene Informationseinheiten und -untereinheiten gegliedert. Die größte Informationseinheit ist der Satz, der aus Feldern besteht. Mehrere Felder können zu einem Abschnitt zusammengefaßt sein. Das Format eines in Abschnitte gegliederten Satzes wird MV (multivariables)-Format genannt, das Format eines Satzes, in dem die Felder nicht in Abschnitten zusammengefaßt sind, wird als V (variables)-Format bezeichnet.

Satz mit V-Format:

Feld 1	Feld 2		Feld n
--------	--------	--	--------

Satz mit MV-Format:

Abschnitt 1			Abschnitt 2		Abschnitt n	
Feld 1	Feld 2	Feld 3	Feld 4	Feld 5	Feld (m-1)	Feld m

Ein Satz muß in eindeutiger Weise gekennzeichnet sein, um ihn von anderen unterscheiden zu können. Die Identifikation erfolgt in CIS über eine Adresse.

2. Verweisdatei

Verweisdateien dienen zum raschen Auffinden von häufig benötigten Informationen. In Verweisdateien wird festgestellt, in welchem Satz einer Hauptdatei die gesuchte Information steht. Eine Verweisdatei (Sekundärindex) ist also mit dem Stichwortverzeichnis in einem Buch vergleichbar.

3. Datenbeschreibungsdatei

Jeder Benutzer benötigt für den Zugang zur CIS-Datenbank eine Datenbeschreibung. Diese stellt die Struktur der Sätze einer Hauptdatei dar. Datenbeschreibungen sind in ihrer Funktion vergleichbar mit der DATA DIVISION eines COBOL-Programms oder den Format-Anweisungen eines FORTRAN-Programms. Ähnlich wie Datenbereiche in COBOL mit Hilfe der REDEFINES-Klausel bzw. in FORTRAN mit der EQUIVALENCE-Anweisung mit verschiedenen Strukturen belegt werden können, so können auch die Sätze einer Hauptdatei mit verschiedenen Datenbeschreibungen unterschiedlich beschrieben werden.

Die zu einer CIS-Datenbank gehörigen Datenbeschreibungen sind in einer Datenbeschreibungsdatei (DABEL) enthalten. Vom Benutzer kann eine Datenbeschreibung über das ihr entsprechende Datenbank-Paßwort angesprochen werden.

Es wird zwischen vier verschiedenen Datenbeschreibungen unterschieden:

- Satzbeschreibung:

Beschreibung der Satzstruktur einer Hauptdatei.

- Transformationsbeschreibungen:

Beschreibung der Umwandlung einer vorgegebenen Satzstruktur in eine mit anderem Aufbau.

- Bildbeschreibungen:

Beschreibung der Ausgabeformate für Drucker und/oder Datenstationen.

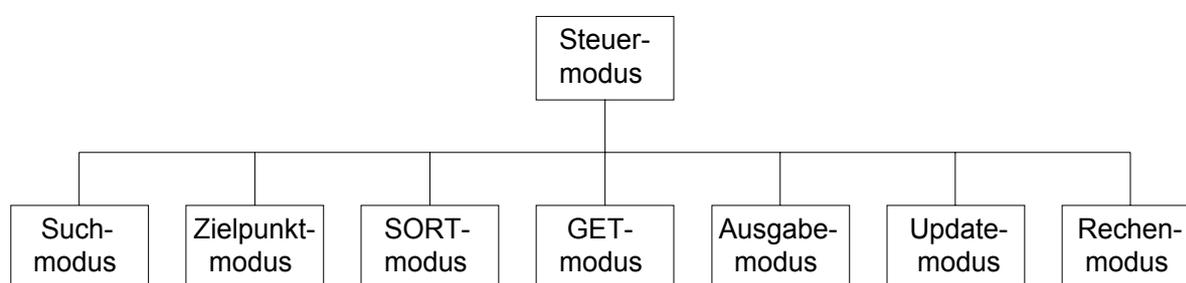
- Bildschirmmasken:

Eingabe der Daten über Bildschirm bzw. Ausgabe der Daten in einem gewünschten Format auf der Datensichtstation.

1.4 Die Grundstruktur des CIS-Kommunikationssystems

Das CIS-Kommunikationssystem ermöglicht dem Benutzer den Zugriff auf die CIS-Datenbanken. Es kann Informationen bzw. Daten abrufen, neue Daten eingeben sowie vorhandene Daten modifizieren oder löschen. Dazu dienen die im Programmsystem implementierten Funktionen, die über Kommandos ansprechbar sind. In ihnen sind gleichartige Funktionen zusammengefaßt. Einen Überblick über die Funktionsmodi gibt Abbildung 1. Ausführlicher wird auf die Funktionen auf Seite 29 eingegangen.

Abbildung 1: Funktionsübersicht des CIS-Kommunikationssystems



Das CIS-Kommunikationssystem kann aktiv oder passiv eingesetzt werden.

1. Aktiv-Betrieb:

Im aktiven Einsatz korrespondiert der Anwender direkt über eine Datenstation mit der Datenbank. Dieser Dialog kann auch simuliert werden, indem die Eingabe der Kommandos von einer Datei aus (Kurzkommando-Datei) oder über ein Programm erfolgt, in das die Module CISKURZ1+CISKURZ oder CISUTMA+CISKURZ oder CISAKT+CISKURZ eingebunden sind.

2. Passiv-Betrieb:

Im passiven Einsatz wird der Zugriff auf die Datenbank durch ein Benutzerprogramm gesteuert. Beim Aufruf von CIS als Unterprogramm werden Parameter (Antwort- /Auftragsparameter) übergeben, die die Kommunikation zwischen CIS und dem Hauptprogramm herstellen.

Die meisten CIS-Funktionen können in beiden Anwendungsarten benützt werden. Nähere Auskunft darüber gibt Manual-3: KOMMANDOS.

1.5 Datenschutz

Um eine mißbräuchliche Nutzung der Daten zu verhindern, darf der Zugriff auf Informationen der Datenbank nur berechtigten Personen möglich sein. CIS kennt drei Möglichkeiten des Zugriffsschutzes, die unabhängig voneinander angewandt werden können. Zwei dieser Schutzvorrichtungen sind durch das Betriebssystem realisiert und daher allgemein verwendbar (d.h. nicht nur bei einer CIS-Datenbank). Die dritte wurde speziell für CIS geschaffen.

1. Benutzerkennung:

Der Zugriff auf die Datenbank kann auf eine einzige Benutzerkennung beschränkt sein und zwar auf diejenige, unter der die Datenbank eingerichtet wurde. Es muß sichergestellt sein, daß die zugehörigen Dateien aufgrund des Katalogeintrages SHARE=NO nur unter dieser Benutzerkennung angesprochen werden können. Der Zugang zu den Daten kann zusätzlich noch durch die Vergabe eines LOGON-Paßwortes abgesichert werden.

2. Paßworte zum Dateischutz:

Eine weitere Möglichkeit des Datenschutzes, die das Betriebssystem bietet, besteht in der Vergabe von Paßworten für den Lese- und Schreibschutz der Dateien der Datenbank. Werden die entsprechenden Kennworte nicht angegeben, so können die auf diese Weise geschützten Dateien nicht geöffnet werden (Näheres vgl. DVS-Beschreibung).

3. Datenbeschreibung:

Der Zugriff auf eine CIS-Datenbank wird durch Datenbeschreibungen gesteuert. Die verwendete Datenbeschreibung wird über ihr Datenbank-Paßwort im CIS-Kommando angegeben. Die Angabe kann, wenn das System die in einem zuvor angewandten Kommando benützte Datenbeschreibung weiterverwenden soll, entfallen.

Ein Benutzer kann also nur mit einer Datenbeschreibung arbeiten, deren Datenbank-Paßwort ihm bekannt ist. Die Datenbeschreibung legt fest, welchen Aufbau ein Satz vom Standpunkt des Anwenders aus hat. Er kann nur die Felder ansprechen, die in dieser Datenbeschreibung vorkommen. Andere Felder, die im Satz existieren, treten für ihn nicht in Erscheinung, ihre Existenz bleibt für ihn unbekannt.

Zusätzlich ist in jeder durch das Paßwort geschützten Datenbeschreibung die Zugriffsart (Suchmodus, Update-Modus etc.) auf die Datenbank durch Angabe entsprechender Parameter festgelegt.

Der Datenbankverantwortliche kann für einen Benutzer nur das Lesen oder auch die Änderung der Datenbanksätze zulassen. Fehlt einer dieser Parameter, so ist der entsprechende Zugang zur Datenbank nicht möglich. Durch die implizite Eröffnung der Dateien kennt nur der Datenbankverantwortliche die Dateinamen, dem jeweiligen Benutzer bleiben sie verborgen.

1.6 Datensicherung

In CIS werden die Daten gegen Verlust durch Bedienfehler, Programmfehler, Systemfehler und physikalische Fehler gesichert. Bei jedem Update-Vorgang werden Duplikate der betreffenden Sätze erstellt.

CIS kennt zwei Sicherungsverfahren:

1. Before-Image Sicherung:

Der Benutzer faßt (mehrere) Änderungen, die für ihn eine logische Einheit bilden, in einer Transaktion zusammen.

Jeder zu ändernde Datensatz wird vor seiner Änderung in die Before-Image Datei kopiert.

Wird eine Transaktion korrekt zu Ende geführt, so werden alle seit Transaktionsanfang durchgeführten Änderungen permanent. Die Before-Image Sätze dieser Transaktion werden gelöscht.

Wird eine Transaktion abgebrochen, so werden alle seit Transaktionsanfang durchgeführten Änderungen zurückgesetzt, d.h. mit den Before-Image Sätzen dieser Transaktion wird der Stand der Datenbank(en) vor Transaktionsanfang wieder hergestellt.

2. After-Image Sicherung:

Die After-Image Sicherung ermöglicht die Wiederherstellung auch bei physikalischer Zerstörung der CIS-Datenbank.

Die After-Image Sicherung baut auf einer Sicherungskopie der Datenbank auf. Gemeinsam mit der Erstellung der Sicherungskopie(n) der CIS-Datenbank(en) muß mit dem Dienstprogramm CISINIT die After-Image Datei initialisiert werden.

Datensätze, die während des CIS-Laufs geändert werden, werden in der After-Image Datei gespeichert.

Bei Auftreten von Fehlern kann mit dem Dienstprogramm CISRA (vgl. Manual-2) aus der Sicherungskopie der Datenbank und den Daten der After-Image Datei der Stand der CIS-Datenbank(en), der beim Ende der letzten korrekten Änderung bzw. Transaktion vorlag, reproduziert werden.

1.7 Dienstprogramme

Zum Einrichten und zur Verwaltung einer CIS-Datenbank werden folgende Dienstprogramme benötigt:

- CISGEN erzeugt und verwaltet die Datenbeschreibungsdatei (DABEL), in der die Datenbeschreibungen und Bildschirmmasken in ladefähiger Form enthalten sind.
- CISLADF erstellt aus einem vorhandenen Datenbestand eine Haupt- und/oder Verweisdatei.
- CISDICT wertet die Datenbeschreibungsdatei aus, ähnlich wie es aus einem Data-Dictionary möglich ist.

Für den Einsatz von CIS mit Datensicherung stehen dem Benutzer folgende Dienstprogramme zur Verfügung:

- CISINIT initialisiert die After-und/oder Before-Image Datei, die für die Datensicherung verwendet werden.
- CISPROT protokolliert die After-Image Datei.
- CISRA führt den Recovery mit den After-Image Sätzen aus, erzeugt also einen konsistenten Zustand der Datenbank
- CISKOOR ist der Koordinationsprozeß der Datensicherung; er verwaltet die Sperrlisten, schreibt die After-/Before-Images und Protokoll-Sätze und setzt mit den Before-Images zurück.
- CISDBH ist der unabhängige CIS Data-Base-Handler
- CISSESV ist der CIS-Server für den Client-Server Betrieb.

Auf diese Programme wird im Manual-2: DIENSTPROGRAMME ausführlich eingegangen.

2.1.1 Abschnitt

Abschnitte werden durch einen Namen, die Abschnittsart, gekennzeichnet. Alle Felder einer Datenbeschreibung müssen mit der Abschnittsart einem bestimmten Abschnitt zugeordnet werden. Die CIS-Kommunikationskomponente erkennt nur an der Abschnittsart, daß der Satz MV-Format hat. Fehlt die Abschnittsart, so hat der Satz V-Format.

Die Abbildungen 2 bzw. 3 zeigen den Aufbau eines Satzes mit V-Format bzw. MV-Format. Die vier Bytes für die Satzlänge am Beginn jedes Satzes enthalten entsprechend den Konventionen des DVS in den ersten zwei Bytes die Satzlänge in binärer Form, die beiden weiteren Bytes sind für Organisationsaufgaben des DVS reserviert. Ab dem vierten Byte folgt bei Sätzen mit V-Format der Datenteil des Satzes. Bei Sätzen mit MV-Format folgt nach den vier Bytes für die Satzlänge die Abschnittslänge und Abschnittsart. Die ersten zwei Bytes jedes Abschnitts enthalten die Abschnittslänge, die nächsten zwei Bytes sind reserviert. In den darauffolgenden vier Bytes ist die Abschnittsart definiert. Ab dem 9. Byte schließt sich der Datenteil des Abschnitts an. Seine Länge kann unterschiedlich sein. Abschnitte sind somit selbständige Informationseinheiten, die wie variable Sätze aufgebaut sind.

Dieser multivariable Aufbau bewirkt, daß die Reihenfolge von Abschnitten innerhalb eines Satzes weitgehend beliebig ist. Jeder Abschnitt kann, ausgehend vom Beginn des Satzes, gefunden und über seinen Namen (Abschnittsart) identifiziert werden.

2.1.2 Felder

Felder sind die kleinsten Informationseinheiten in einem Satz. Sie bilden den Datenteil eines Satzes bzw. eines Abschnitts. Jedes Feld, das in einer Satzbeschreibung beschrieben ist, trägt einen `F e l d n a m e n` (bzw. Feldbezeichnung). Bei einem Satz mit V-Format wird ein Feld relativ zum Satzbeginn, bei einem Satz mit MV-Format relativ zum Beginn des Abschnitts adressiert. In der Satzbeschreibung sind alle für den Benutzer zugänglichen Felder des Satzes mit Feldnamen, Anfangsadresse und Länge in Bytes beschrieben. Bei Sätzen mit MV-Format muß auch die Abschnittsart für alle Felder angegeben sein.

Felder können sich entsprechend ihrer Anfangsadresse und ihrer Länge überlappen. Ist ein Feld ganz oder teilweise in einem anderen enthalten, so spricht man von einem Unterfeld.

Normalerweise sind Feldern eine feste Länge zugeordnet. Es können jedoch auch variabel lange Felder definiert werden. Sie müssen in der Satzbeschreibung besonders gekennzeichnet sein und können nur am Ende eines Satzes (bei V-Format) bzw. eines Abschnitts (bei MV-Format) auftreten.

Abbildung 4 zeigt als Beispiel aus der Personalabteilung einen Abschnitt in einem MV-Satz. Der Abschnitt ist 22 Bytes lang. Er hat den Namen STEU. Die Felder tragen die Feldnamen STEUERKLASSE, LOHNSTEUER, KONF (Konfession), BETRAG, KIRCHENSTEUER und STEUER. Das Feld mit der Bezeichnung STEUER wird durch die beiden Unterfelder, bezeichnet mit LOHNSTEUER und KIRCHENSTEUER, gebildet, wobei das Unterfeld, mit dem Feldnamen KIRCHENSTEUER noch zwei weitere Unterfelder mit der Bezeichnung KONF und BETRAG enthält.

Abbildung 4: Beispiel für einen Abschnitt in einem MV-Satz

		Felder																			
		STEUER										STEUERKL									
Abschnittslänge		Abschnittsart				LOHNSTEUER					KIRCHENSTEUER										
						KONF					BETRAG										
2	2	S	T	E	U	0	0	6	9	8	5	5	0	R	K	7	8	8	4		
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22

2.1.3 Wiederholabschnitte

Ein Satz kann mehrere gleiche Abschnitte enthalten. Alle werden durch die gleiche Abschnittsart gekennzeichnet. Solche Abschnitte werden als Wiederholabschnitte bezeichnet. Sie sind innerhalb eines Satzes nur durch ihre Reihenfolge zu unterscheiden. Jedem wiederholten Abschnitt einer Art wird eine Abschnittsmultiplizität zugeordnet. Der erste Abschnitt einer Art, vom Beginn des Satzes aus gerechnet, besitzt die Multiplizität 1, der zweite die Multiplizität 2 usw.. Abschnitte einer Art werden in einer Datenbeschreibung stets nur einmal definiert. Sie müssen in der Datenbeschreibung als wiederholbar gekennzeichnet sein.

Der erste Abschnitt eines Satzes darf nicht wiederholbar sein.

2.1.4 Wiederholfelder

Ebenso wie Abschnitte der gleichen Art innerhalb eines Satzes wiederholt werden können, können auch Felder gleicher Art mehrmals auftreten.

Nur das letzte Feld in einem Satz (bei V-Format) bzw. in einem Abschnitt (bei MV-Format) kann ein Wiederholfeld sein.

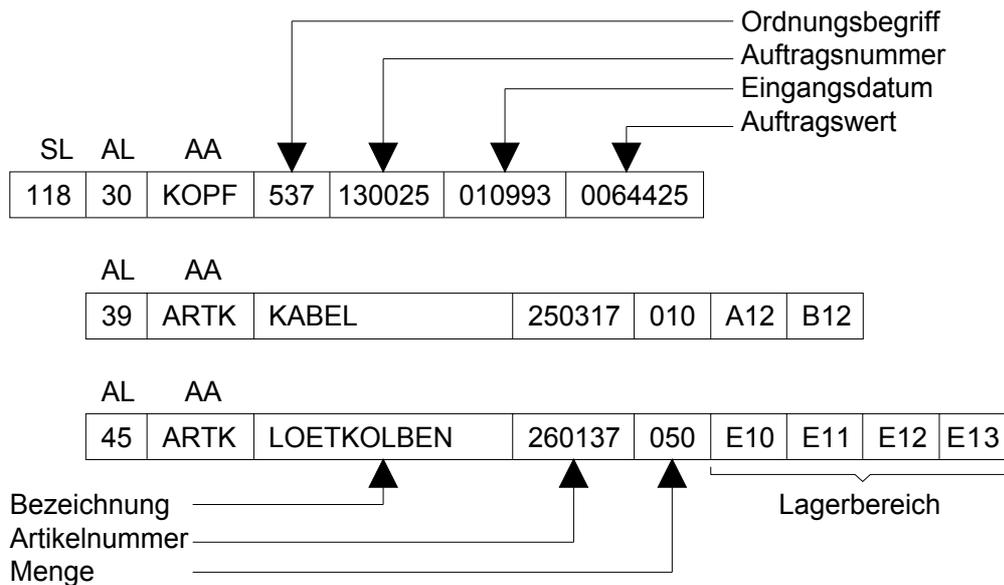
Wiederholfelder müssen als solche in der Satzbeschreibung gekennzeichnet sein. Sie werden aber nur einmal beschrieben. Wiederholten Feldern wird die Feldmultiplizität zugeordnet. Sie entspricht analog zur Abschnittsmultiplizität der Reihenfolge der wiederholten Felder innerhalb eines Satzes (bei V-Format) bzw. eines Abschnitts (bei MV-Format). Ein Feld kann auch Wiederholfeld in einem Wiederholabschnitt sein. Zu seinem Auffinden sind sowohl die Abschnitts- als auch die Feldmultiplizität nötig.

2.1.5 Beispiel eines multivariablen (MV-) Satzaufbaus

Eine Zulieferfirma für Installationsmaterial verwaltet die Bestellungen ihrer Kunden in einer Auftragsdatei. Die Auftragsätze stehen über die Auftragsnummer und das Eingangsdatum in eindeutiger Beziehung zum Kunden.

Zusätzlich zur Artikelbezeichnung, -nummer und -menge benötigt die Lagerverwaltung Angaben über die unterschiedlichen Lagerbereiche des Artikels.

Abbildung 5: Beispiel für Wiederholabschnitte und Wiederholfelder



SL = Satzlänge
 AL = Abschnittslänge
 AA = Abschnittsart

Ein Auftrag stellt in der Hauptdatei einen Satz dar. Der Satz enthält zwei Abschnittsarten: KOPF für den Auftragskopf und ARTK für die verschiedenen Positionen des Auftrages.

Der Abschnitt mit dem Namen KOPF enthält den Ordnungsbegriff, die Auftragsnummer, das Eingangsdatum und den Auftragswert.

Die Abschnitte mit dem Namen ARTK enthalten die Artikelspezifikationen und kommen in diesem dargestellten Auftragsatz zweimal vor. Einmal für den Artikel KABEL, zum Anderen für den Artikel LOETKOLBEN. Es handelt sich somit um einen Wiederholabschnitt.

Die Artikel werden in mehreren Lagerbereichen, einem Wiederholfeld, geführt. Das Wiederholfeld muß am Ende des jeweiligen Abschnitts stehen. Das Feld, das beispielsweise den Wert E12 enthält, hat die Abschnittsmultiplizität 2 und die Feldmultiplizität 3.

2.2 Datenbeschreibungsdatei

Die Datenbeschreibungsdatei (DABEL) ermöglicht den Zugriff auf die CIS-Datenbanken. Während eines Programmlaufs der Kommunikationskomponente kann stets nur eine DABEL verwendet werden. Alle CIS-Meldungen und gültigen Kommandos werden in der jeweiligen Sprache (Deutsch/Englisch) in der DABEL abgelegt. Eine DABEL (Abkürzung für Datenbeschreibungen ladefähig) enthält Datenbeschreibungen in ladefähiger Form, d.h. in einer Form, in der sie unmittelbar von der Kommunikationskomponente verwendet werden können. Jede ladefähige Datenbeschreibung ist ein ISAM-Satz variabler Länge. Der sechs Zeichen lange frei wählbare Teil xxxxxx des zugehörigen Datenbank-Paßwortes DB.xxxxxx ist der ISAM-Schlüssel dieses Satzes und der Namen der Datenbeschreibung.

Jede Datenbeschreibung bezieht sich auf eine bestimmte Hauptdatei und, falls notwendig, auf eine bestimmte Verweisdatei. Der Bezug wird durch die Angabe der Namen dieser Dateien in der Datenbeschreibung hergestellt. Eine DABEL kann Datenbeschreibungen enthalten, die sich auf verschiedene Haupt- und Verweisdateien beziehen.

Datenbeschreibungen regeln innerhalb von CIS den Zugang der Benutzer zur Datenbank. Dazu gehören sowohl Steuerfunktionen verschiedener Art als auch die Beschreibung der Satzstruktur, die den Sätzen der mit der Datenbeschreibung verbundenen Hauptdatei zugrunde liegt. Man unterscheidet vier Typen von Datenbeschreibungen:

- Satzbeschreibung
- Transformationsbeschreibung
- Bildbeschreibung
- Bildschirmmaske.

Die Verwaltung der Datenbeschreibungen und Masken (Erstellen, Ändern, Löschen) erfolgt mit dem Dienstprogramm CISGEN. Sie werden aus einer Primärform in die ladefähige Form übersetzt (generiert) und in der DABEL abgelegt.

Bildschirmmasken spielen eine gewisse Sonderrolle. Ihr Aufbau ist von den drei anderen Datenbeschreibungsarten wesentlich verschieden und ihre Anwendung ist eng begrenzt auf wenige Typen von Kommandos. Diese Kommandos übertragen die Daten vom Bildschirm ins Benutzerprogramm (bei passivem CIS) oder in die Datenbank (bei aktivem CIS).

DATEIEN

Aufbau der Datenbeschreibungen:

Datenbeschreibungen sind segmentiert aufgebaut (vgl. Abbildung 6 auf Seite 20 und Abbildung 7 auf Seite 22).

Satzbeschreibungen bestehen aus einem A-Segment der Kopfbeschreibung, sowie D- und E-Segmenten der Feldbeschreibung.

Transformations- und Bildbeschreibungen enthalten ein A-Segment in der Kopfbeschreibung, sowie D-Segmente in der Feldbeschreibung. Darüber hinaus können auch noch E- und F-Segmente in der Feldbeschreibung auftreten.

1. A-Segment

Im A-Segment wird das Datenbank-Paßwort festgelegt. Ferner können Parameter angegeben werden, die für das Laden und Verändern (Update) einer Hauptdatei von Bedeutung sind. Mit Parameterangaben kann der Zugriff auf Daten und auf einzelne Funktionen (Update, Suchen, Ausgeben) eingeschränkt werden. Weitere Parameter steuern die Druckerausgaben, sowie das implizite Öffnen der Haupt- und Verweisdatei, auf die sich die Datenbeschreibung bezieht.

2. D-Segment

Das D-Segment dient drei Zielen:

- Adressierung eines Feldes innerhalb des Satzes.
- Formatdefinition des Feldinhaltes (binär, gepackt, ...)
- Festlegung eines Kurznamens (Segmentname in der Verweisdatei) bei Invertierung (Spalte 32-34 der Feldbeschreibung)

Bei Transformations- und Bildbeschreibungen wird im D-Segment angegeben, welches Feld wohin abgebildet werden soll und auf welche Weise die Übertragung auszuführen ist.

3. E-Segment

E-Segmente in Satzbeschreibungen definieren den Feldnamen.

E-Segmente in Transformations- und Bildbeschreibungen ergänzen die Abbildungsvorschrift von Wiederholfeldern bzw. -abschnitten (Manual 2: E-Segment der Transformationsbeschreibung).

4. F-Segment

Ein F-Segment (Eintrag F unter SEGM der Feldbeschreibung) enthält eine Konstante (z.B. Text bei Bildbeschreibungen für Bezeichnungen), die bei der Transformation in den zu erstellenden Satz eingefügt werden.

2.2.1 Satzbeschreibungen

Felder können nur über ihre Feldnamen identifiziert und angesprochen werden. Die Zuordnung der Feldnamen zu den Feldern einschließlich deren Eigenschaften erfolgt in einer Satzbeschreibung.

Die Abbildungen 6 und 7 stellen den Aufbau und die Verwendung der Satzbeschreibungen dar. Beide Abbildungen beziehen sich auf das Beispiel auf Seite 16.

Die Darstellung der beiden Abbildungen entspricht dem Ausdruck der Datenbeschreibung mit CISGEN.

Die KOPFBESCHREIBUNG beinhaltet das A-Segment, d.h. den Namen (Paßwort) der Datenbeschreibung und die in ihm angegebenen Steuerparameter.

Die FELDBESCHREIBUNG der Satzbeschreibung kann folgende Bereiche enthalten:

SEGM	'D'-Kennzeichen für D-Segment	Spalte 3
QUELLDEFINITION		Spalte 4-14
AA	Abschnittsart: (muß bei MV-Format belegt werden)	
ADR	Feldanfangsadresse	
LNG	Feldlänge	
LOG-DARST		Spalte 15-17
LOG	Logische Feldbedeutung Vgl. Manual 2, 2.4.2 D-Segment der Satzbeschreibung.	
DARST	Feldformat Vgl. Manual 2, 2.4.2 D-Segment der Satzbeschreibung.	
SONDERFUNKTION		Spalte 32-35
SF1	Kurzname bei Invertierung	
SF2	Reserviert	
WA	Wiederholabschnitt	
SEGM	'E'-Kennzeichen für E-Segment	Spalte 45
FELDBEZEICHNUNG	Feldbezeichnung	Spalte 46

Die Feldanfangsadresse in der Rubrik ADR, die Länge des Feldes unter LNG und die Feldbedeutung unter LOG-DARST müssen stets angegeben sein.

DATEIEN

Abbildung 6: Beispiel für den Aufbau einer Satzbeschreibung

S E G M	KOPFBESCHREIBUNG												
	NAME	PARAMETER IM A-SEGMENT											
A A	AUFTR1	LOGADR=J , DSS=J , DSA=J , DSU=N , MAXDES=16 , HD=HD . AUFTRAG , VD=VD . AUFTRAG											
FELDBESCHREIBUNG													
S E G M	QUELL- DEFINITION			L D O A G R S T	T V	ZIEL- DEFINITION			SONDER FUNKTION		W A	S E G M	FELDBEZEICHNUNG TRANSFORMATIONS- ERGAENZUNG
	AA	ADR	LNG			AA	ADR	LNG	SF1	SF2			
3	4	8	12	15	18	21	25	29	32	36	0	5	46 (41-44 FREI)
D	KOPF	9	3	O	B							E	ORDNUNGSBEGRIFF
D	KOPF	12	6	T					001			E	AUFTRAGSNUMMER
D	KOPF	18	6	T								E	EINGANGSDATUM
D	KOPF	24	7	2R								E	AUFTRAGSWERT
D	ARTK	9	16	T					BEZ		W	E	BEZEICHNUNG
D	ARTK	25	6	T							W	E	ARTIKELNUMMER
D	ARTK	31	3	P							W	E	MENGE
D	ARTK	34	3	W	T						W	E	LAGERBEREICH
E N D E													

Die in AUFTR1 beschriebenen Sätzen sind Sätze mit MV-Format. Es können in ihnen Abschnitte der Art KOPF und ARTK vorkommen. Zwingend notwendig ist jedoch nur, daß in jedem Satz der Hauptdatei HD.AUFTRAG der Abschnitt KOPF vorhanden ist und an erster Stelle steht, da er den Ordnungsbegriff enthält.

Jeder Satz enthält den ISAM-Schlüssel (ORDNUNGSBEGRIFF), der im Abschnitt KOPF steht, sowie Felder mit den Bezeichnungen AUFTRAGSNUMMER, EINGANGSDATUM und AUFTRAGSWERT. Der Wiederholabschnitt ARTK kann beliebig oft auftreten (null bis n-mal), ebenso das mit LAGERBEREICH bezeichnete Wiederholfeld.

Die Satzbeschreibung in Abb. 6 erlaubt auf alle Informationen zuzugreifen, soweit sie vorhanden sind. Hingegen lassen sich mit der Satzbeschreibung AUFTR2 in Abb. 7 (vgl. Seite 22) nur die artikelspezifischen Daten ansprechen.

Die logische Feldbedeutung des Feldes, dem die Bezeichnung ORDNUNGSBEGRIFF zugeordnet ist, wird mit 'O' gekennzeichnet (erstes eingetragenes Zeichen unter LOG-DARST). Der Abschnitt ARTK ist als Wiederholabschnitt definiert, da unter WA (Spalte 40), bei allen zu diesem Abschnitt gehörenden Feldern, ein W eingetragen ist. Die Felder, mit den Feldbezeichnungen AUFTRAGSNUMMER und BEZEICHNUNG, sind invertierte Felder, da unter SF1 (Spalte 32-34) der Kurzname 001 bzw BEZ eingetragen ist. Das Feld mit der Feldbezeichnung LAGERBEREICH ist ein Wiederholfeld, da unter LOG-DARST (Spalte 15) die logische Feldbedeutung 'W' eingetragen ist.

Das erste Feld eines Abschnitts beginnt erst ab Byte 9, da beim MV-Format die ersten 8 Bytes von der Abschnittslänge (4 Byte) und der Abschnittsart (4 Byte) belegt werden.

Die Rubrik DEF enthält die einzelnen Formatdefinitionen:

nP gepackte Zahl mit n Dezimalstellen

B binäre Darstellung

nR rechtsbündige entpackte Zahl mit n Dezimalstellen

T linksbündige alphanumerische Zeichenfolge (Text), rechts ggf. mit Spaces aufgefüllt.

Weitere Formatdefinitionen sind im Manual 2, "D-Segment der Satzbeschreibung", beschrieben.

DATEIEN

Abbildung 7: Beispiel für den Aufbau einer Satzbeschreibung

SEGMENT	KOPFBESCHREIBUNG												
	NAME	PARAMETER IM A-SEGMENT											
AA	AUFTR2	LOGADR=J , DSS=J , DSA=J , DSU=N , MAXDES=16 , HD=HD . AUFTRAG , VD=VD . AUFTRAG											
FELDBESCHREIBUNG													
SEGMENT	QUELL-DEFINITION			L D O A G R S T	T V	ZIEL-DEFINITION			SONDERFUNKTION		W A	SEGMENT	FELDBEZEICHNUNG TRANSFORMATIONS- ERGAENZUNG
	AA	ADR	LNG			AA	ADR	LNG	SF1	SF2			
3	4	8	12	15	18	21	25	29	32	36	0	5	46 (41-44 FREI)
D	KOPF	9	3	O	B							E	KEY
D	ARTK	9	16	T					BEZ			W	E ART-BEZEICHNUNG
D	ARTK	25	6	T								W	E ART-NUMMER
D	ARTK	31	3	T								W	E ART-MENGE
D	ARTK	34	3	W	T							W	E ART-LAGERBEREICH
E N D E													

Die Satzbeschreibung AUFTR2 in Abb. 7 beschreibt wie die Satzbeschreibung AUFTR1 die Sätze der gleichen Hauptdatei HD.AUFTRAG. Auch der Zugriffsschutz ist mit den Parametern DSS, DSA und DSU in gleicher Weise geregelt.

Die Felder werden in AUFTR2, im Gegensatz zur Satzbeschreibung AUFTR1, mit anderen Namen, über die sie angesprochen werden, bezeichnet. Die Felder, mit den Namen AUFTRAGSNUMMER, EINGANGSDATUM und AUFTRAGSWERT, sind zwar in den Sätzen der Hauptdatei vorhanden, können aber nicht angesprochen werden.

2.2.2 Transformations- und Bildbeschreibung

Transformationsbeschreibungen werden fast ausschließlich zum Laden eines schon vorhandenen Datenbestandes mit CISLADF verwendet.

Transformationsbeschreibungen legen eine Abbildungsvorschrift fest. Sie ermöglicht es, aus Sätzen einer vorgegebenen Struktur Sätze mit einer anderen Struktur zu erstellen. Der Ausgangssatz wird bei einer Transformation nicht verändert.

CIS-Satzformate sind beliebig ineinander überführbar. Folgende Typen von Abbildungen sind möglich:



Die Definition der Abbildung erfolgt feldweise. Ein Feld muß aber nicht mit einem in der Satzbeschreibung beschriebenen Feld identisch sein sondern kann ein beliebiger Bereich innerhalb eines Satzes (bei V-Format) bzw. eines Abschnitts (bei MV-Format) sein. Für jedes Feld sind im D-Segment folgende Angaben notwendig:

Eingabedefinition (QUELLDEFINITION): Beschreibung der Felder des umzuformenden (bzw. auszugebenden bei Bildbeschreibung) Satzes mit Abschnittsart, Feldanfangsadresse und Feldlänge.

Ausgabedefinition (ZIELDEFINITION) Beschreibung der Felder des zu erzeugenden Satzes mit Abschnittsart, Feldanfangsadresse und Feldlänge.

Übertragungsvorschrift (TV) bestimmt wie die Information des umzuformenden in den zu erzeugenden Satz übertragen werden soll.

Der Aufbau der zu erzeugenden Sätze erfolgt in der Reihenfolge der Felder der Quelldefinition.

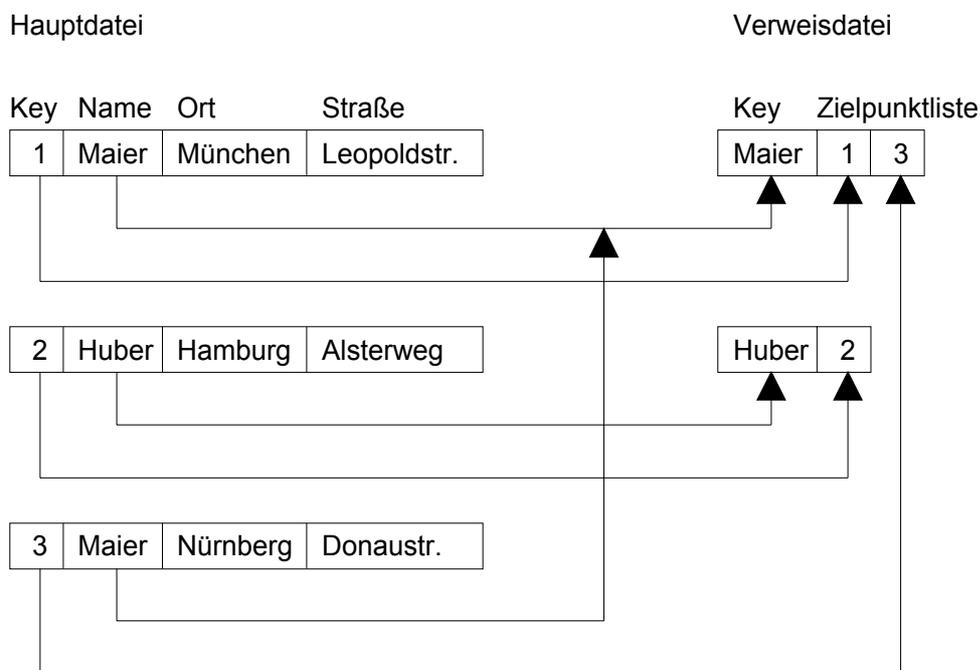
Bildbeschreibungen dienen zur Ausgabe der Daten auf Bildschirm oder Drucker. Sie stellen eine Sonderform der Transformationsbeschreibung dar. Der Aufbau der Bildbeschreibung entspricht der Transformationsbeschreibung. Mit der Bildbeschreibung wird ein Satz in der Kommunikationskomponente erzeugt und anschließend ausgegeben.

2.3 Verweisdatei

Ein schneller Zugriff (SUCHE) ist nur über den Ordnungsbegriff (Primärindex) bei ISAM oder die Blocknummer bei PAM möglich. Ist kein Ordnungsbegriff vorhanden (SAM) oder soll nach zusätzlichen Feldern performant gesucht werden, müssen Sekundärindizes in einer Verweisdatei aufgebaut werden (Invertierung).

Alle Felder der Satzbeschreibung, die invertiert werden, erhalten einen Kurznamen (Spalte 32-34). Mit dem Dienstprogramm CISLADF wird die Verweisdatei aufgebaut. Der Inhalt des gekennzeichneten Feldes des Hauptdateisatzes wird zum Ordnungsbegriff (Primärindex) des Verweisdateisatzes. Der Ordnungsbegriff des Hauptdateisatzes bei ISAM, die Wiedergewinnungsadresse bei SAM oder die Block- und Satznummer bei PAM wird zum Satzinhalt des Verweisdateisatzes.

Abbildung 8: Schematische Darstellung der Invertierung (Invertierung des Feldes mit der Bezeichnung: Name)



Da gleiche Werte eines invertierten Feldes in verschiedenen Sätzen der Hauptdatei nur *einen* Primärindex bilden können, müssen diesem Primärindex alle Ordnungsbegriffe der Hauptdatei zugeordnet sein.

Abbildung 9: Schematischer Aufbau eines Satzes in einer Verweisdatei

Inhalt	LL	Seg	Feldinhalt	US	OB bzw. Adresse	EK	
Länge	2	2	3	MAXDES	4	n x Länge OB	4

ISAM-Schlüssel der Verweisdatei

LL	Satzlängengebiet Die folgenden 2 Bytes sind vom DVS reserviert.
Seg	Segmentname (Kurzname, Spalte 32-34 der Satzbeschreibung)
MAXDES	Längenangabe in Stellen, mit der der Feldinhalt der invertierten Felder in den ISAM-Schlüssel der Verweisdatei eingehen soll. Feldinhalte, deren Länge kleiner als die angegebene MAXDES ist, werden auf die volle Länge (MAXDES) mit Spaces aufgefüllt. Feldinhalte, deren Länge größer als MAXDES ist, werden nur bis zu der Stelle übertragen, die der angegebenen MAXDES (A-Segment der Satzbeschreibung) entspricht.
US	Wird beim Aufbau eines Verweisdateisatzes infolge der Vielzahl von Verweisen die max. Satzlänge von 2048 Bytes überschritten, werden weitere Verweisdateisätze gebildet, deren ISAM-Schlüssel den gleichen Segmentnamen und Feldinhalt des invertierten Feldes und einen für jeden weiteren Satz unterschiedlichen Unterschlüssel enthalten. Der Unterschlüssel des ersten Satzes ist immer 1, die folgenden sind aufsteigend numeriert. Für Einfügungen sind Nummernlücken freigelassen.
OB/Adresse	Ordnungsbegriff: ISAM-Schlüssel der betreffenden Hauptdateisätze in aufsteigender Reihenfolge. Adresse: Wiedergewinnungsadresse bzw. Block- und Satznummer der Sätze bei SAM bzw. PAM-Dateien in aufsteigender Reihenfolge.
n	Anzahl der Verweise (OB bzw. Adresse) auf die Hauptdatei.
EK	Endekennung 0 Letzter Verweisdateisatz zu einem bestimmten Inhalt des Invertierten Feldes. ungleich 0 Es folgt ein weiterer Satz zu einem bestimmten Inhalt des invertierten Feldes.

Bei Änderung des Inhalts eines invertierten Feldes werden die betroffenen Verweisdateisätze ebenfalls geändert. Bei Invertierung numerischer Felder wird der jeweilige Zahlenwert über einen Umwandlungsalgorithmus so umgeformt, daß in der Verweisdatei Gleichlauf zwischen Sortierreihenfolge und arithmetischer Wertigkeit erreicht wird. Dadurch wird beim SORTIEREN gewährleistet, daß negative Zahlenwerte kleiner als positive Zahlen erkannt werden.

2.4 CIS-fremde Dateien

CIS kann auf Dateien zugreifen, die nicht von CIS oder speziell für CIS aufgebaut sind.

CIS läßt für ISAM, PAM und LEASY-Dateien keine Update-Funktionen zu. Sekundärindizes von LEASY können direkt von CIS verarbeitet werden. Der Aufbau von Sekundärindizes für SAM- und PAM-Dateien ist möglich.

2.4.1 Zugriff auf SAM- und PAM-Dateien

SAM- und PAM-Dateien können nur im INPUT-Modus verarbeitet werden. Sätze mit fester (RECFORM=F) oder variabler Satzlänge (RECFORM=V) können gelesen werden. Der Aufbau einer Verweisdatei erfolgt mit dem Dienstprogramm CISLADF (Funktion V).

Außer AENDERN, EINGEBEN, KORRIGIERE, LOESCHEN und PUT sind alle Kommandos erlaubt. Das RECHNE-Kommando kann nicht zum Ändern in der Hauptdatei verwendet werden. Das OPEN-Kommando darf die Hauptdatei nur INPUT eröffnen.

CIS benötigt einen Ordnungsbegriff bzw. eine eindeutige Adresse jedes Satzes. In den Benutzerdaten kann ein solcher Schlüssel nicht vorausgesetzt werden. CIS verwendet deshalb bei PAM die Blocknummer und bei SAM die Wiedergewinnungsadresse als Schlüssel. In der Satzbeschreibung wird daher kein Ordnungsbegriff definiert. Ein direkter Zugriff (SUCHE bzw. GET,KD) ist nicht möglich.

Für PAM-Dateien können im A-Segment der Satzbeschreibung der Satztyp (RECFORM (Default-Wert V)) und die Satzlänge (RECSIZE (Default-Wert BLKSIZE)) angegeben werden. Bei RECFORM=V werden die ersten zwei Bytes des Satzes als Satzlängenfeld interpretiert. Die Satzlänge muß bei sequentiellem Lesen der Regel

$$(BLKSIZE-2047) \leq n \leq BLKSIZE$$

entsprechen.

Bei PAM-Dateien mit RECFORM=F können mehrere Sätze in einem Block vorhanden sein. Die PAM-Seiten müssen nicht lückenlos belegt sein. Nicht belegte PAM-Seiten werden von CIS nicht berücksichtigt.

Bandverarbeitung: SAM- und PAM-Dateien können auch als Banddateien von CIS verarbeitet werden. Insbesondere ist bei SAM-Banddateien folgendes zu beachten:

- STD-Blockangabe
- Keine Mehrbanddatei

2.4.2 Zugriff auf LEASY-Dateien

LEASY-Dateien werden nur im INPUT-Modus verarbeitet. Die Sekundärindizes von LEASY können verwendet werden. SAM-Dateien, für die LEASY keine Sekundärindizes führen kann, werden wie andere CIS-fremde Dateien verarbeitet (vgl. auf Seite 26: Zugriff auf SAM- und PAM-Dateien).

Außer AENDERN, EINGEBEN, KORRIGIERE, LOESCHEN und PUT sind alle Kommandos erlaubt. Das RECHNE-Kommando kann nicht zum Ändern der Hauptdatei verwendet werden. Das OPEN-Kommando darf Dateien nur INPUT eröffnen. SICHERN- und FREIGEBEN-Kommandos dürfen nur mit Typ E, also temporäres Sichern, verwendet werden.

Die Satzbeschreibung für den CIS-Zugriff auf LEASY-Dateien kann mit CISGEN aus evtl. vorhandenen COBOL-COPY Elementen beschrieben werden (vgl. Manual-2: CISGEN, LDB-Anweisung).

Compound Keys (Zusammengesetzte Schlüssel-Feldbedeutung Z) existieren nur in der LEASY-SI Datei und dürfen deshalb nur zur indirekten Suche verwendet werden.

LEASY bewertet die Feldinhalte von numerischen Feldern bei der Sekundärindizierung nicht, d.h. LEASY bildet die Feldinhalte in der SI-Datei 1:1 ab. Damit sind die Werte nach ihrem binären Äquivalent sortiert und bei uneinheitlicher Vorzeichenverwendung können für den gleichen numerischen Wert mehrere SI-Sätze entstehen (vgl. Manual-3: KOMMANDOS, Behandlung numerischer Feldinhalte).

3 Datenbankzugriffe

3.1 Bereitstellung von Zielpunktlisten (ZPL)

3.1.1 Zielpunktlistenerstellung mit Suchfragen

Das Ergebnis nach Ausführung eines Suchkommandos ist eine Zielpunktliste. Sie enthält die Ordnungsbegriffe bzw. Adressen aller Sätze in aufsteigender Reihenfolge, für die die logische(n) Bedingung(en) zutreffen. Außerdem erhält der Benutzer die Anzahl der Treffer:

Suchfragen werden in Initialfragen und Folgefragen eingeteilt.

1. Initialfrage

Eine Initialfrage besteht aus einem oder mehreren Suchkommandos, die mit der Operation SUCHE beginnen. Sie erstellt eine neue Zielpunktliste.

2. Folgefrage

Eine Folgefrage beginnt mit einer logischen Verknüpfungsoperation, (UND, NICHT, ODER). Sie verknüpft eine bereits vorhandene aktuelle Zielpunktliste mit einer weiteren logischen Suchbedingung, so daß eine aktuelle Zielpunktliste entsteht.

Für das Suchergebnis ist es gleichgültig, ob die einzelnen Suchkommandos in einer einzigen Initialfrage zusammengefaßt sind, oder nacheinander in einer Initialfrage und mehreren Folgefragen abgesetzt werden. Die Abarbeitung der Suchteile erfolgt in jedem Fall sequentiell in der angegebenen Reihenfolge.

Abhängig davon, ob die im Suchkommando auftretenden Felder invertiert sind oder nicht, wird zwischen verschiedenen Typen von Suchkommandos unterschieden. Sie unterscheiden sich nicht in ihrer syntaktischen Form.

Direktes Suchen
Indirektes Suchen
Sequentielles Suchen

1. Direktes Suchen (Hauptdatei ist eine ISAM-Datei)

SUCHE Ordnungsbegriff=...

Enthält die Suchfrage den Ordnungsbegriff, so ist bei der Erstellung der Zielpunktliste nur ein Zugriff auf die Hauptdatei notwendig.

2. Indirektes Suchen

Enthält eine Suchfrage invertierte Felder, wird die Hauptdatei erst nach Zugriff auf die Verweisdatei in deren Zielpunktlisten die Ordnungsbegriffe/Adressen der Hauptdatei enthalten sind, gelesen.

3. Sequentielles Suchen

Bei Suchfragen mit nicht invertierten Feldern müssen alle Sätze der Hauptdatei sequentiell gelesen werden, um überprüfen zu können ob die in der Suchfrage angegebene logische Bedingung erfüllt ist.

3.1.2 Aktivieren von Zielpunktlisten

Die Bereitstellung einer aktuellen Zielpunktliste muß nicht immer über ein Suchkommando erfolgen. In vielen Fällen wird es sinnvoll sein, eine Zielpunktliste, die zu einem früheren Zeitpunkt erstellt und gesichert wurde, zu aktivieren und wieder zu verwenden. Zur Sicherstellung von Zielpunktlisten dient die Operation SICHERN (vgl. Manual-3:KOMMANDOS). Die Typergänzung E im Kommando bewirkt eine vorübergehende (temporäre) Sicherung. In diesem Fall ist eine Zielpunktliste jedoch nur während eines Programmlaufs aktivierbar. Soll eine Zielpunktliste längere Zeit verfügbar sein, muß sie in die Verweisdatei gesichert werden (Typergänzung V).

Die temporär gesicherte Zielpunktliste wird mit dem AKTIVIEREN-Kommando, die Verweisdatei-Zielpunktliste mit dem SUCHE-Kommando aktiviert.

3.1.3 Sortieren von Zielpunktlisten

Die gefundenen Sätze in der ZPL können nach bestimmten Kriterien mit dem SORT-Kommando sortiert werden.

3.2 Dateiverknüpfung

Mehrere Datenbanken (bis zu 11 Hauptdateien) lassen sich mit Retrieval-CIS-Funktionen wie eine Datenbank behandeln. Der Zugriff auf die Dateien erfolgt über Verweisdateien.

Die Datensätze aus den zu verbindenden Datenbanken werden über gemeinsame Feldinhalte verknüpft, d.h. die betreffenden Felder müssen invertiert sein. Diese Verknüpfung zu Gesamtsätzen geschieht mit Hilfe des VERBINDE-Kommandos (vgl. Manual-3: KOMMANDOS). Diese Menge von Gesamtsätzen (Relationszeilen) stellt für den Anwender eine virtuelle Datenbank dar, die er wie eine einzelne CIS-Datenbank bearbeiten kann. Diese virtuelle Datenbank heißt in der CIS-Terminologie RELATION oder FENSTER.

Die bei der Verknüpfung beteiligten Dateien können von gleichem oder unterschiedlichem Typ sein (ISAM, SAM, PAM).

4 Einrichten und Verwalten von CIS-Datenbanken

4.1 Erstellen einer Datenbeschreibungsdatei

Die Voraussetzung für jede Arbeit mit einer CIS-Datenbank ist eine Datenbeschreibungsdatei (DABEL). Sie enthält Datenbeschreibungen und Bildschirmmasken in generierter Form sowie Texte für Fehlermeldungen und Informationen. Die Verwaltung der Datenbeschreibungsdatei erfolgt mit dem Dienstprogramm CISGEN.

Ist keine DABEL beim Aufruf von CISGEN vorhanden, so richtet CISGEN eine Datenbeschreibungsdatei mit dem Dateinamen DABEL ein:

Mit der Kommandofolge

```
/START-PROGRAM FROM-FILE=CISGEN
END
```

werden nur die Texte (CIS-Fehlertexte, -Infotexte) in die DABEL geladen.

Um Datenbeschreibungen und Bildschirmmasken in die DABEL aufzunehmen, kann in CISGEN mit dem CISGEN-Kommando 'EDTF' in den EDT verzweigt werden. In ihm werden die Datenbeschreibungen und Masken in ihrer Primärform definiert. Die Primärform ist im Manual-2: DIENSTPROGRAMME im Kapitel CISGEN ausführlich beschrieben.

Mit dem EDT-Kommando 'HALT' wird nach CISGEN zurückgesprungen.

Mit dem CISGEN-Kommando 'GEND' (generieren Datenbeschreibungen) werden alle Datenbeschreibungen aus dem EDT-Speicher in die DABEL geschrieben (in ladefähiger Form). Analog dazu werden mit dem CISGEN-Kommando 'GENM' (generieren Masken) die Bildschirmmasken generiert.

```
/START-PROGRAM FROM-FILE=CISGEN
EDTF      Erstellung der Datenbeschreibungen und Bildschirmmasken im EDT.
HALT      Beendigung des EDT-Modus.

GEND      Generierung der Datenbeschreibungen.
GENM      Generierung der Bildschirmmasken.
END
```

Satzbeschreibungen können auch mit dem CISGEN-Kommando DEFD maskengeführt erfaßt und ergänzt werden.

4.2 Verwalten der Datenbeschreibungsdatei

Die DABEL wird mit dem Dienstprogramm CISGEN verwaltet. Zu den Verwaltungsfunktionen gehören:

1. Automatische Aufnahme der Fehlertexte z.B. bei Versionswechsel.
2. Ändern, Löschen und Neuaufnahme von Datenbeschreibungen und Bildschirmmasken.

Beispielsweise werden Datenbeschreibungen wie folgt geändert:

```
/START-PROGRAM FROM-FILE=CISGEN
RGEND ASP*      Alle Datenbeschreibungen, die mit dem Teilnamen ASP beginnen, werden
                 aus der Dabel in den EDT-Speicher geladen (regeneriert).

EDTF           Änderung der Datenbeschreibungen im EDT.

HALT          Beendigung des EDT-Modus.

PUTD          Alle im EDT-Speicher stehenden Datenbeschreibungen werden in die DABEL
                 geschrieben (generiert).

END
```

4.3 Erstellen einer Hauptdatei

Wird der erste Satz einer Hauptdatei mit dem CIS-Kommando PUT,K (K steht für Kapitel = Satz) oder mit EINGEBEN,K geschrieben, so wird die neue Hauptdatei (HD) von CIS eingerichtet. Einige Parameter für das SET-FILE-LINK Kommando können dabei im A-Segment der Satzbeschreibung (vgl. Manual-2, CISGEN) angegeben werden.

```
/START-PROGRAM FROM-FILE=CIS
EINGEBEN,K,DB.MASK01      Feldinhalte, erfaßt mit Hilfe der Bildschirmmaske MASK01,
                           werden in einen Satz geschrieben. HD (Hauptdatei) und VD
                           (Verweisdatei, falls Felder invertiert sind) werden von CIS
                           eingerichtet.
```

HALT

Hauptdateien können ebenso mit dem Dienstprogramm CISLADF (vgl. Manual 2) unter Verwendung eines bereits vorhandenen Datenbestandes erstellt werden.

Ein Datenbestand, der in eine Hauptdatei umgesetzt werden soll, kann als ISAM-oder als SAM-Datei vorliegen. Vor dem Aufruf des Dienstprogrammes CISLADF muß dieser Datei mit einem SET-FILE-LINK Kommando der LINK-Name EIN zugewiesen werden. Die Daten, aus denen ein Satz erzeugt werden soll, müssen innerhalb eines Satzes der Eingabedatei stehen.

Die zu erzeugende Hauptdatei wird vor dem Aufruf von CISLADF mit einem SET-FILE-LINK Kommando, das den LINK-Namen HD enthält, zugewiesen. Ferner sind in diesem SET-FILE-LINK Kommando mindestens

KEYPOS und KEYLEN

anzugeben, die die Position und die Länge des ISAM-Schlüssels der HD festlegen.

```
/SET-FILE-LINK FILE-NAME=EINGABE, LINK-NAME=EIN
/SET-FILE-LINK FILE-NAME=HD.AUFTRAG, LINK-NAME=HD,
ACCESS-METHOD=ISAM(KEY-LENGTH=3, KEY-POSITION=5)
/START-PROGRAM FROM-FILE=CISLADF
L,H,DB.AUFTRG              Laden Hauptdatei
```

Die Datenbeschreibung mit dem Paßwort AUFTRG kann eine Satzbeschreibung oder eine Transformationsbeschreibung sein. Ist es eine Satzbeschreibung, so werden die Satzinhalte der Eingabedatei 1:1 übernommen. Ist es eine Transformationsbeschreibung so können die Satzinhalte aus dem vorliegenden Format in einen Satz mit anderem Format umgewandelt werden.

Das Dienstprogramm CISLADF kann aus einem Datenbestand entweder nur eine Hauptdatei oder nur eine Verweisdatei oder beides erzeugen. Das Laden aus einem Datenbestand wird in zwei Fällen erforderlich:

- Daten, die in einem weitgehend beliebigen aber fest definierten Format vorliegen, sollen mit CIS weiterbearbeitet und verwaltet werden,
- Reorganisation einer vorhandenen Hauptdatei wenn die Sätze z.B. in ihrem Format geändert werden sollen.

4.4 Erstellen einer Verweisdatei

Beim erstmaligen Schreiben eines Satzes mit invertierten Feldern wird von CIS eine Verweisdatei, falls sie noch nicht existiert, angelegt. Dazu wird der VD-Parameter im A-Segment der Satzbeschreibung (vgl. Manual-2, CISGEN Satzbeschreibung) verwendet, falls kein expliziter OPEN in CIS - mit dem Kommando OPEN VD=Dateiname - durchgeführt wurde.

Die Generierung einer Verweisdatei mit CISLADF wird normalerweise nur in folgenden Fällen notwendig sein (vgl. Manual-2: CISLADF):

1. Eine Hauptdatei wurde aus einem bestehenden Datenbestand erstellt.
2. Reorganisation einer bestehenden Verweisdatei, ohne daß die Hauptdatei reorganisiert werden muß.
3. In einer bestehenden Hauptdatei sollen bestimmte Felder neu invertiert werden.

Vor dem Aufruf von CISLADF wird die bestehende Hauptdatei mit dem SET-FILE-LINK Kommando, das den LINK-Namen EIN enthält, zugewiesen. Die Verweisdatei wird von CISLADF angelegt bzw. zugewiesen (falls sie bereits katalogisiert ist).

```
/SET-FILE-LINK FILE-NAME=HD.AUFTRAG, LINK-NAME=EIN  
/START-PROGRAM FROM-FILE=CISLADF  
L, V, DB.AUFTRG          Laden Verweisdatei
```

Das Paßwort AUFTRG bezeichnet eine Satzbeschreibung, die im A-Segment den VD-Parameter enthält.

4.5 Einrichten der Datensicherungsdateien

Die Datensicherung bearbeitet folgende Dateien:

- After-Image Datei
- Before-Image Datei
- Protokoll-Datei

Wird mit independent CISKOOR gearbeitet, gibt es diese Dateien jeweils nur ein Mal pro CISKOOR. Sie werden zentral von der CIS-Datensicherung bearbeitet.

Die Dateien werden mit dem SET-FILE-LINK Kommando angelegt und mit dem Dienstprogramm CISINIT (vgl. Manual-2) initialisiert.

Ebenso können mit der DO-Prozedur D.CIS.DASI.INSTALLIEREN die Datensicherungsdateien angelegt werden (vgl. Installation der Datensicherung auf Seite 46 und Beispiel von D.CIS.DASI.INSTALLIEREN auf Seite 48).

4.6 Entscheidungskriterien

4.6.1 Entscheidungskriterien für Suchwege, Länge invertierter Felder und Datenbankschlüssel

CIS kennt drei Suchwege:

1. Direkt über den Datenbankschlüssel (nur bei ISAM-HD)

Es wird in der Hauptdatei mit ISAM direkt gelesen, d.h. in der Regel mit dem GETKY-Makro. In jedem Satz muß ein Feld als Ordnungsbegriff erklärt sein, der als ISAM-Schlüssel verwendet wird.

2. Indirekt über die Verweisdatei

In der Verweisdatei wird mit einem ISAM-Zugriff ein Satz gelesen, der die ISAM-Schlüssel bzw. Adressen der Hauptdatei enthält, mit denen in der Hauptdatei direkt zugegriffen werden kann.

3. Sequentiell

Wenn für ein Feld keine der beiden Möglichkeiten zutrifft (d.h. Das Feld ist nicht Ordnungsbegriff und ist nicht invertiert), wird "sequentiell" in der Hauptdatei gelesen.

Dabei werden zwei Fälle unterschieden:

- Bei einer Initialfrage (vgl. Zielpunktlistenenerstellung mit Suchfragen auf Seite 29) wird die Hauptdatei sequentiell von Anfang bis Ende durchsucht, d.h. es wird der gesamte Datenbestand gelesen.

- Bei einer Folgefrage (vgl. Zielpunktlistenenerstellung mit Suchfragen auf Seite 29) werden alle Sätze, die über die vorhandene Zielpunktliste adressiert sind, direkt gelesen. In der Regel wird dadurch nur ein kleiner Teil des Datenbestands gelesen.

Das Zeitverhalten ist stark abhängig vom peripherem Speicherbedarf, den Suchwegen, der Anwendung und vom Inhalt der Datenbank. Die Zusammenhänge sind sehr komplex und nicht formelmäßig darstellbar. Ohne genaue Kenntnis der Anwendung und des Datenbestandes können deshalb keine Empfehlungen für die Struktur der Datenbank gegeben werden.

Im Folgenden sind die wichtigsten Kriterien aufgeführt, die in die Überlegungen eingehen sollten. Aus vorstehenden Gründen muß eine Wertung an dieser Stelle unterbleiben.

1. Direkte Suchfragen nach einem bestimmten Ordnungsbegriff sind sehr schnell. Alle anderen Fragestellungen erfordern das Lesen mindestens so vieler Sätze wie Treffer erzielt werden.
2. Nur ein Feld kann als Ordnungsbegriff (Datenbankschlüssel) erklärt werden.
3. Der Ordnungsbegriff kann nicht direkt in der Datenbank geändert werden, sondern nur über Löschen und Neuaufnahme eines Satzes.
4. Der Ordnungsbegriff muß innerhalb der Datenbank eindeutig sein.
5. Der sequentielle Suchweg ist schneller als der invertierte, wenn aus einem Datenbestand bereits eine relativ kleine Menge ausgewählt ist und diese Menge mit einer Bedingung, die auf wesentlich mehr als die zuvor ausgewählte Menge zutrifft, weiter eingeschränkt werden soll.

Beispiel: aus einer Personaldatenbank

Es gibt 5 Ingenieure mit Englischkenntnissen. Es soll nun eingeschränkt werden, wer davon ledig ist. Es gibt 800 ledige Mitarbeiter.

6. Im Änderungsmodus muß zusätzlich zur Hauptdatei je invertiertem Feld die Verweisdatei fortgeschrieben werden.
7. Wie oft wird nach einzelnen Feldinhalten gesucht?
8. Wie oft werden einzelne Feldinhalte geändert?
9. Die Verweisdatei enthält je invertierten Feldinhalt mindestens einen Satz. Je Satz wird der Feldinhalt und 15 Bytes Verwaltungsinformation benötigt. Der Feldinhalt wird in der Länge von MAXDES dargestellt. Ein hoher MAXDES-Wert erfordert eine große Verweisdatei.
10. In der Verweisdatei werden je invertierten Feldinhalt die zugehörigen Ordnungsbegriffe bzw. Adressen gespeichert. Lange Ordnungsbegriffe brauchen mehr Platz und evtl. mehr Sätze.
11. Der ISAM-Schlüssel der Verweisdatei setzt sich aus 7 Bytes Verwaltungsinformation und dem Feldinhalt in der Länge von MAXDES zusammen. ISAM-Zugriffe mit langen ISAM-Schlüsseln sind ungünstig.
12. Der ISAM-Schlüssel der Hauptdatei ist der Ordnungsbegriff. ISAM-Zugriffe mit langen ISAM-Schlüsseln sind ungünstig.

4.6.2 Entscheidungskriterien für das Satzformat

Das V-Format ohne Wiederholfeld oder variabel langem Feld ist das einfachste Satzformat. Sätze mit V-Format können jedoch nicht komprimiert werden, da nicht belegte Felder nicht wegfallen dürfen

Im Gegensatz dazu läßt sich mit dem MV-Format eine hierarchische Satzstruktur bilden. Dies ermöglicht die Datenkompression, d.h. nicht belegte Abschnitte, außer dem ersten der den Ordnungsbegriff enthält, werden nicht in den Satz geschrieben. Die Adressierung der Abschnitte erfolgt über die Anfangsadresse und die Abschnittsart.

Suchfragen lokalisieren Sätze und keine Abschnitte oder Felder. Dies kann eventuell beim MV-Format mit Wiederholabschnitten zu Problemen führen.

Die Abschnitte der Sätze mit MV-Format sind logische Informationseinheiten, bestehend aus beliebig vielen Feldern (z.B. Kundenanschrift mit den Feldern Name, Straße, PLZ, Ort oder Lager mit dem Wiederholfeld Lagerort etc.). Sie werden in der Satzbeschreibung nur einmal definiert und können beliebig oft im Satz vorkommen. Der Vorteil ist die höhere Performance, da alle Informationen im Satz stehen und nicht verschiedene Datenbanken verbunden werden müssen.

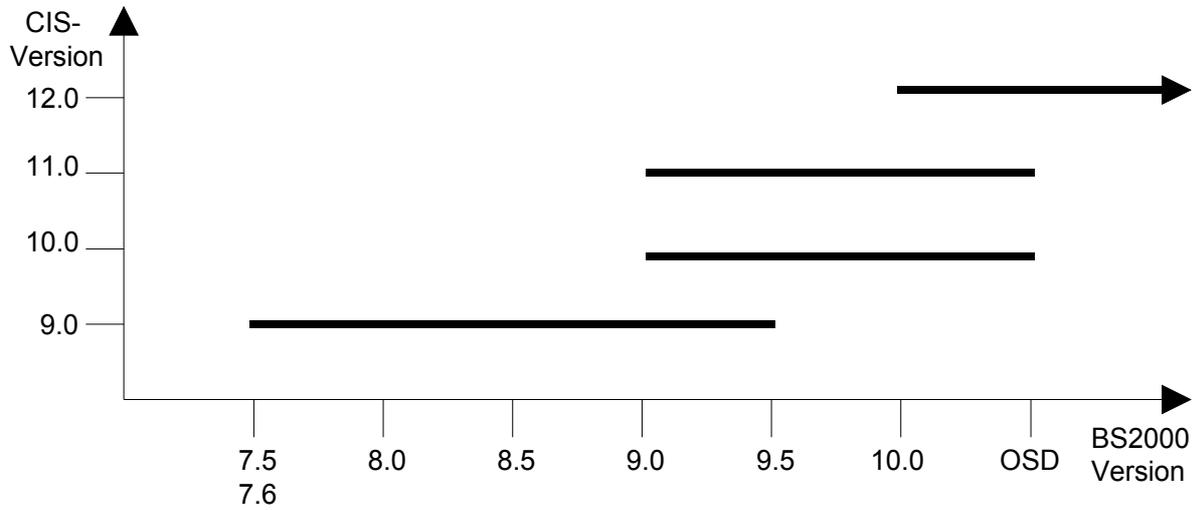
5 Installation CIS

5.1 Varianteneinteilung in CIS V12.0

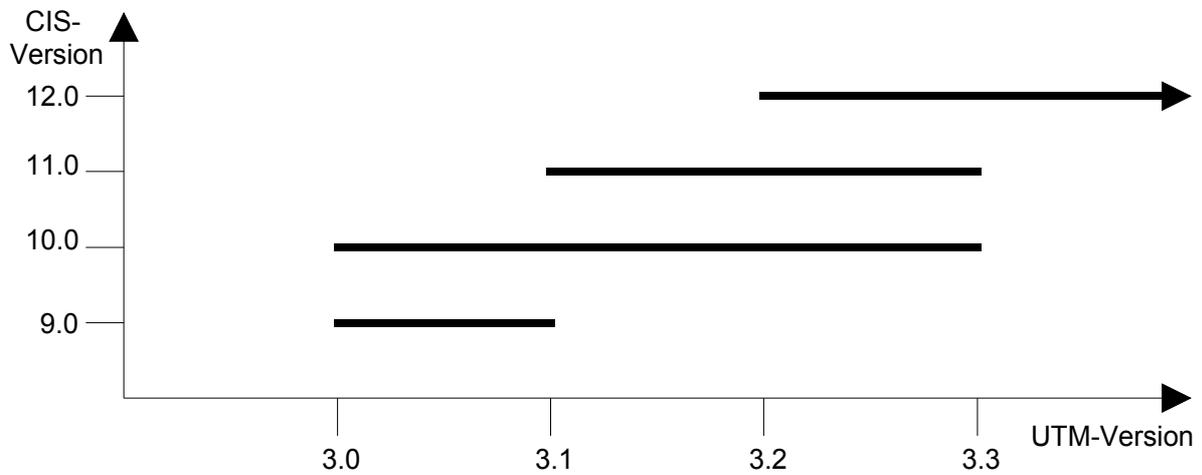
	CIS-STD	CIS-IDP	CIS-SCS	CIS-ICS
Inlinked Data-Base-Handler	x	x	x	x
Independent Data-Base-Handler		x		x
Teilnehmerbetrieb mit TIAM	x	x	x	x
Teilhhaberbetrieb mit UTM im unsynchronisierten Mehr-Task-Modus	x	x	x	x
Teilhhaberbetrieb mit UTM im synchronisierten Mehr-Task-Modus		x		x
Teilhhaberbetrieb mit beliebigen DC-Monitoren bzw. eigenen DCAM-Programmen	x	x	x	x
Mehrrechnerbetrieb über DCAM		x		x
Dialogsprache (Aktivbetrieb)	x	x	x	x
Programmaufruf von CIS (Passivbetrieb)	x	x	x	x
Verbindungsmodul zu Clients, CISDC			x	x
Treiber für 4-GL UnifAce unter MS-Windows (auf Wunsch)			x	x
Treiber für 4-GL UnifAce unter SINIX (auf Wunsch)			x	x

CIS-STD CIS-Standard
 CIS-IDP CIS-Independent
 CIS-SCS CIS-Standard-Client/Server
 CIS-ICS CIS-Independent-Client/Server

5.2 Versionsabhängigkeiten CIS-BS2000



5.3 Versionsabhängigkeiten CIS-UTM



Zusätzliche Versionsabhängigkeiten der CIS-Version 12

- LMS \geq 1.4
- EDT \geq 16.3
- UTM \geq 3.2
- SORT \geq 7.3

5.4 Übernahme der Dateien

Die Dateien werden über das Verfahren SOLIS ausgeliefert, außer CISDC für MS-Windows und der Treiber für UnifAce unter SINIX bzw. MS-Windows. Diese sind über das CIS-Team zu beziehen.

Zur Übernahme der Dateien sind die SOLIS-Richtlinien zu beachten.

5.4.1 Inhaltsverzeichnis der Version 12

Programme und Bibliotheken

Variante				Dateiname	Inhalt
STD	IDP	SCS	ICS		
x	x	x	x	CIS	CIS mit Datensicherung
x	x	x	x	CIS.ODASI	CIS ohne Datensicherung
		x	x	CIS.SINIX	LMS-Bibliothek mit CISDC-Dateien für SINIX sowie FT-Prozeduren zum Übertragen in SINIX.
	x		x	CISDBH	Independent Data-Base-Handler
x	x	x	x	CISDICT	Auswertung Data-Dictionary
x	x	x	x	CISGEN	Datenbeschreibungsgenerator-deutsch
x	x	x	x	CISGENE	Datenbeschreibungsgenerator-englisch
	x		x	CISIND	TIAM-CIS-aktiv für CISDBH
x	x	x	x	CISINIT	Initialisierung der Datensicherungsdateien
x	x	x	x	CISKOOR	Koordinationsprozeß
x	x	x	x	CISLADF	Ladeprogramm für Haupt- und Verweisdatei
x	x	x	x	CISPROT	Protokollierung
x	x	x	x	CISRA	Recovery mit After-Image-Datei
		x	x	CISSERV	CIS-Server
x	x	x	x	MODLIB.CIS.12	Modulbibliothek
	x		x	MACLIB.CIS.KDCB	Makrobibliothek mit den Makros KDCDB und KDCDBC

INSTALLATION

Prozeduren

Variante				Dateiname	Inhalt
STD	IDP	SCS	ICS		
x	x	x	x	D.CIS.DASI.INSTALLIEREN	Prozedur zur Installation der Datensicherung
x	x	x	x	D.CIS.GEN.CISDRV	Prozedur zum Generieren eigener Nachladetreiber
x	x	x	x	D.KORR	REP's und DPAGE-Korrekturen
x	x	x	x	D.KORR.OPTIONAL.MUSTER	Musterdatei für eigene Korrekturen
x	x	x	x	D.LNK.CIS	Bindeprozedur für CIS
x	x	x	x	D.LNK.CIS.ODASI	Bindeprozedur für CIS.ODASI
	x		x	D.LNK.CISDBH	Bindeprozedur für CISDBH
x	x	x	x	D.LNK.CISGEN	Bindeprozedur für CISGEN
x	x	x	x	D.LNK.CISGENE	Bindeprozedur für CISGENE
	x		x	D.LNK.CISIND	Bindeprozedur für CISIND
x	x	x	x	D.LNK.CISLADF	Bindeprozedur für CISLADF
		x	x	D.LNK.CISSERV	Bindeprozedur für CISSERV
		x	x	D.LNK.CISSERV.ODASI	Bindeprozedur für CISSERV.ODASI
x	x	x	x	SYS.SSD.CIS.120	Datei für Subsystem
x	x	x	x	SYSFGM.CIS.120.D	Freigabetext

Makros in der Makrobibliothek MACLIB.CIS.KDCDB

Makro	Bedeutung
KDCDB	Generierung des Anschlusses von UTM an CIS.
KDCDBC	Generierung des Anschlusses von UTM an CIS falls UTM mit mehreren Datenhaltungssystemen koordiniert zusammenarbeitet.

Module in der Modulbibliothek MODLIB.CIS.12

Variante				Dateiname	Inhalt
STD	IDP	SCS	ICS		
x	x	x	x	CIS	CIS-Anschlußmodul
x	x	x	x	CISAKT	Anschlußprogramm für CIS-Aktiv als Unterprogramm
x	x	x	x	CISCOBMV	Satzaufbereitung
	x		x	CISCON	Connection-Modul zu CISDBH
	x		x	CISDBH	Steuerung Data-Base-Handler
x	x	x	x	CISDRV	Modul zum Generieren eigener Nachladetreiber
x	x	x	x	CISGEN	Datenbeschreibungsgenerator (passiv)-deutsch
x	x	x	x	CISGENE	Datenbeschreibungsgenerator (passiv)-englisch
x	x	x	x	CISGENIO	Ein-/ Ausgabe zu CISGEN / CISGENE
x	x	x	x	CISI	CIS-Code (invariant)
x	x	x	x	CISKURZ	Kurzkommando-Prozessor
x	x	x	x	CISKURZ1	TIAM Ein-/ Ausgabe für CISKURZ
x	x	x	x	CISLADF	Ladeprogramm für Haupt- und Verweisdateien
x	x	x	x	CISLADIO	Ein-/ Ausgabe zu CISLADF
		x	x	CISSERV	Hauptmodul zu CISSERV
	x		x	CISTIAM	CIS-Anschlußmodul für TIAM-CISDBH
x	x	x	x	CISTSIM	Terminalsimulation für CISAKT
x	x	x	x	CISU	Dummy
x	x	x	x	CISUTM	CIS-Anschlußmodul für UTM / DBH
x	x	x	x	CISUTMA	UTM-Teilprogramm für CIS-Aktiv
x	x	x	x	CISV	CIS-Bereiche (variant)
x	x	x	x	CISVARI	CIS-Variable mit Datensicherung
x	x	x	x	CISVARI1	CIS-Variable ohne Datensicherung
x	x	x	x	CISVERS	Ausgabe der aktuellen Version

5.5 Installation der Datensicherung

Folgende Schritte müssen für eine voll funktionsfähige Datensicherung ausgeführt werden.

1. CISVARI an die Belange des Anwenders anpassen.
2. CIS neu binden.
3. CISDBH neu binden (nur bei Variante IDP) oder über Startparameter modifizieren.
4. Anwenderprogramm neu binden.
5. ENTER-Datei für CISKOOR erstellen.
6. Anlegen und Initialisieren der Datensicherungsdateien.
7. Eventuell DO-Prozedur zum Starten von CISRA erstellen.

Die Punkte 1 und 2 sowie 5 bis 7 können mit der Prozedur:

```
/CALL-PROCEDURE NAME=D.CIS.DASI.INSTALLIEREN
```

ausgeführt werden. Mit

```
/CALL-PROCEDURE NAME=D.CIS.DASI.INSTALLIEREN, LANG=E
```

wird die englische Version aufgerufen.

Die Prozedur sollte in jedem Fall unter der Benutzerkennung ablaufen, unter der die CIS-Programme und CIS-Bibliotheken enthalten sind.

Die Prozedur schreibt folgende Dateien:

CIS.DASI.WORK.WORK	Arbeitsdatei. Sie wird vor Ende der Prozedur gelöscht.
CIS.DASI.WORK.LMS.LNK	ENTER-Datei zum Anpassen des Moduls CISVARI mit LMS und zum Binden von CIS.

CISVARI in CIS mit Datensicherung:

Parameter	in CISVARI		mögliche Werte	Wert bei Auslieferung
	ab Byte	Länge		
CISKOOR-Typ	X'09'	1	I/J *)	J
After-Image-Datei ?	X'0A'	1	J/N	J
Before-Image-Datei ?	X'0B'	1	J/N	J
Protokolldatei ?	X'0C'	1	J/N	N
Arbeiten mit Transaktionen	X'0D'	1	J/N	J
VD-After-Images ?	X'B6'	1	J/N	N
Memory-Pool-Größe **)	X'B4'	2	C'cc' X'xxxx'	C'12'
CISKOOR "VON"-Zeit	C'C7'	4	0000-2400	0700
CISKOOR "BIS"-Zeit	C'CB'	4	0000-2400	1800
Beendigungsbedingung von CISKOOR	X'157'	1	1-3	1
Parameter für automatischen ENTER-JOB	x'300'	200	Operand für ENTER-JOB	Spaces

*) I = IN-Linked-CISKOOR
J = CISKOOR als eigener Task

**) Der Memory-Pool dient als Kommunikationsbereich zwischen CIS und CISKOOR: Seine Größe ist als Mehrfaches von 64 KB angegeben (vgl. Manual 2: CISKOOR, Memory-Pool).

INSTALLATION

5.5.1 Beispiel von D.CIS.DASI.INSTALLIEREN

Im Folgenden wird ein möglicher Ablauf der Prozedur D.CIS.DASI.INSTALLIEREN dargestellt, die im Dialog die Installation der Datensicherung unterstützt.

```
(IN) DO D.CIS.DASI.INSTALLIEREN
(OUT) CIS-V12.0 : INSTALLATION DER DATENSICHERUNG
(OUT) NAMEN DER AFTER-IMAGE-DATEI ODER "N" EINGEBEN
(IN) CIS.AIM
(OUT) NAMEN DER BEFORE-IMAGE-DATEI ODER "N" EINGEBEN
(IN) CIS.BIM
(OUT) NAMEN DER LOCK-DATEI EINGEBEN
(IN) CIS.LCK
(OUT) NAMEN DER PROTOKOLL-DATEI ODER "N" EINGEBEN
(IN) N
(OUT) NAMEN DER DABEL EINGEBEN
(IN) DABEL
(OUT) WERDEN VD-AFTER-IMAGES GESCHRIEBEN? (J/N)
(IN) N
(OUT) CISKOOR-TYP? (I=IN-LINKED/J=INDEPENDENT)
(IN) J
(OUT) WAS IST DAS CISKOOR-BEENDIGUNGSNIVEAU? (1 BIS 3)
(IN) 2
(OUT) CISKOOR-"VON"-ZEIT EINGEBEN (0000 - 2400)
(IN) 0700
(OUT) CISKOOR-"BIS"-ZEIT EINGEBEN (0700 - 2400)
(IN) 2130
(OUT) SOLL MIT AUTOMATISCHEM ENTER GEARBEITET WERDEN? (J/N)
(IN) J
(OUT) NAME DER ENTER-DATEI FUER CISKOOR ?
(IN) $CIS.ENTER.CISKOOR
(OUT) USER-ID FUER CISKOOR-LAUF ?
(IN) CIS
(OUT) ACCOUNTING NUMBER ? (ODER '*NONE')
(IN) 23400
(OUT) PASSWORT ? (ODER 'NONE')
(IN) 'OX'
(OUT) PRIORITAET ? (ODER 'STD')
(IN) STD
(OUT) MAXIMALE LAUFZEIT ? (ODER 'STD' ODER 'NO')
(IN) STD
(OUT) FROM-FILE=$CIS.ENTER.CISKOOR,PROC-ADMISS=PAR(USER-ID=CIS,
ACCOUNT=23400,PASS='OX'),RESOURCES=PAR(RUN-PRIOR=STD,CPU-L ( LIMIT=STD)
(OUT) PARAMETER O.K.? (J/N)
(IN) J
(OUT) MEMORY-POOL-GROESSE?
(IN) 16
(OUT) DATEI CIS.DASI.WORK.LMS.LNK ERSTELLT
(OUT) /LOGON
(NL) /SET-FILE-LINK FILE-NAME=DABEL,LINK-NAME=DB
(NL) /SET-FILE-LINK FILE-NAME=CIS.AIM,LINK-NAME=AIM
(NL) /SET-FILE-LINK FILE-NAME=CIS.BIM,LINK-NAME=BIM
(NL) /SET-FILE-LINK FILE-NAME=CIS.LCK,LINK-NAME=LCK
(NL) /START-PROGRAM FROM-FILE=CISKOOR
(NL) KAIM=J
(NL) KBIM=J
(NL) KLOG=N
(NL) KTRANS=J
(NL) KEND=2
(NL) KFROM=0700
(NL) KTO=2130
(NL) KPOOL=16
(NL) END
```

```

(NL) /LOGOFF
(OUT) DATEI $CIS.ENTER.CISKOOR ERSTELLT
(OUT) SOLL ENTER ZUM ANPASSEN CIS GESTARTET WERDEN? (J/N/P=DRUCKEN)
(IN) J
(OUT) SOLL EINE DO-PROZEDUR FUER DIE INITIALISIERUNGEN GENERIERT WERDEN?
(J/N)
(IN) J
(OUT) PROZEDUR-DATEI-NAMEN EINGEBEN
(IN) D.CISINIT
(OUT) PRIMAERZUWEISUNG FUER AIM-DATEI EINGEBEN (3-16384)
(IN) 300
(OUT) SEKUNDAERZUWEISUNG FUER AIM-DATEI EINGEBEN (3-16384)
(IN) 30
(OUT) PRIMAERZUWEISUNG FUER BIM-DATEI EINGEBEN (3-16384)
(IN) 300
(OUT) SEKUNDAERZUWEISUNG FUER BIM-DATEI EINGEBEN (3-16384)
(IN) 30
(OUT) PRIMAERZUWEISUNG FUER LOCK-DATEI EINGEBEN (3-16384)
(IN) 300
(OUT) SEKUNDAERZUWEISUNG FUER LOCK-DATEI EINGEBEN (3-16384)
(IN) 30
(OUT) /BEGIN-PROCEDURE LOGGING=NO
(NL) /SET-FILE-LINK FILE-NAME=DABEL, LINK-NAME=DB
(NL) /DELETE-FILE FILE-NAME=CIS.AIM
(NL) /SET-JOB-STEP
(NL) /CREATE-FILE FILE-NAME=CIS.AIM, SUPPORT=PUBLIC-DISK( -
(NL) /SPACE=RELATIVE(PRIMARY-ALLOCATION=300, SECONDARY-ALLOCATION=30))
(NL) /SET-FILE-LINK FILE-NAME=CIS.AIM, LINK-NAME=AIM
(NL) /DELETE-FILE FILE-NAME=CIS.BIM
(NL) /SET-JOB-STEP
(NL) /CREATE-FILE FILE-NAME=CIS.BIM, SUPPORT=PUBLIC-DISK( -
(NL) /SPACE=RELATIVE(PRIMARY-ALLOCATION=300, SECONDARY-ALLOCATION=30))
(NL) /SET-FILE-LINK FILE-NAME=CIS.BIM, LINK-NAME=BIM
(NL) /DELETE-FILE FILE-NAME=CIS.LCK
(NL) /SET-JOB-STEP
(NL) /CREATE-FILE FILE-NAME=CIS.LCK, SUPPORT=PUBLIC-DISK( -
(NL) /SPACE=RELATIVE(PRIMARY-ALLOCATION=300, SECONDARY-ALLOCATION=30))
(NL) /SET-FILE-LINK FILE-NAME=CIS.LCK, LINK-NAME=LCK
(NL) /ASSIGN-SYSDTA TO-FILE=*SYSCMD
(NL) /START-PROGRAM FROM-FILE=CISINIT
(NL) AIM
(NL) BIM
(NL) END
(NL) /ASSIGN-SYSDTA TO-FILE=*PRIMARY
(NL) /END-PROCEDURE
(OUT) SOLL EINE DO-PROZEDUR FUER RECOVERY GENERIERT WERDEN? (J/N)
(IN) J
(OUT) PROZEDUR-DATEI-NAMEN EINGEBEN
(IN) D.CISRA
(OUT) /BEGIN-PROCEDURE
(NL) /SET-FILE-LINK FILE-NAME=DABEL, LINK-NAME=DB
(NL) /SET-FILE-LINK FILE-NAME=CIS.AIM, LINK-NAME=AIM
(NL) /ASSIGN-SYSDTA TO-FILE=*SYSCMD
(NL) /START-PROGRAM FROM-FILE=CISRA
(NL) DAT=1000
(NL) TID=2000
(NL) END
(NL) ALL
(NL) /ASSIGN-SYSDTA TO-FILE=*PRIMARY
(NL) /END-PROCEDURE
(OUT) SOLL INITIALISIERUNGSPROZEDUR GESTARTET WERDEN? (J/N)
(IN) J
(OUT) % BLS0500 PROGRAM 'CISINIT', ...
(OUT) % BLS0551 COPYRIGHT (C) SNI 1994. ALL RIGHTS RESERVED
(OUT) CI50CISINIT V12.0 - ...
(OUT) CI52PARAMS (AIM/BIM/PRO/END) ?
(OUT) CI52PARAMS (AIM/BIM/PRO/END) ?
(OUT) CI52PARAMS (AIM/BIM/PRO/END) ?
(OUT) CI51ENDE CISINIT

```

INSTALLATION

Inhalt von CIS.DASI.WORK.LMS.BIND :

```
/LOGON
/MODIFY-JOB-SWITCHES ON=(1,4,5)
/START-PROGRAM FROM-FILE=$LMS
LIB MODLIB.CIS.12,USAGE=BOTH
PAR OVERWRITE=YES,LOG=MAX
PRT CISLDI
UPDR CISVARI
*COR CISVARI,009,C'J' * CISKOR-TYP (I/J/N) *
*COR CISVARI,00A,C'J' * AFTER-IMAGE-DATEI (J/N) *
*COR CISVARI,00B,C'J' * BEFORE-IMAGE-DATEI (J/N) *
*COR CISVARI,00C,C'N' * PROTOKOLL-DATEI (J/N) *
*COR CISVARI,00D,C'J' * TRANSAKTIONEN (J/N) *
*COR CISVARI,0B6,C'N' * VD-AFTER-IMAGES (J/N) *
* PARAMETER FUER AUTOMATISCHEN ENTER (200 BYTE):*
*COR CISVARI,300,C'FROM-FILE=$CIS.ENTER.CISKOR,PRO'
*COR CISVARI,320,C'C-ADMISS=PAR(USER-ID=CIS,ACCOUNT'
*COR CISVARI,340,C'=23400,PASS='OX'),RESOURCES=PAR('
*COR CISVARI,360,C'RUN-PRIOR=STD,CPU-LIMIT=STD) '
*COR CISVARI,380,C' '
*COR CISVARI,3A0,C' '
*COR CISVARI,3C0,C' '
*END
PRT (LST)
LSTP CISLDI
END
/MODIFY-JOB-SWITCHES OFF=(1,4,5)
/CALL-PROCEDURE NAME=D.LNK.CIS
/LOGOFF
```

Anmerkung: Die bei der Installation durchgeführten Änderungen werden in der Modulbibliothek im Listenelement CISLDI protokolliert. CISLDI (Elementtyp P) kann damit jederzeit per LMS zu Prüfzwecken ausgedruckt werden.

6 Begriffsbestimmungen

Abschnitt

Informationseinheit innerhalb eines **multivariablen Satzes**. Ein Abschnitt ist in **Felder** unterteilt, **Wiederholabschnitte** sind möglich.

Abschnittsart

Kennzeichnung bei **multivariablen** Format für alle **Felder**, die zu einem **Abschnitt** gehören.

Abschnittsmultiplizität

Erste bis n-te Wiederholung eines nur einmal in einer **Datenbeschreibung** definierten **Abschnittes** in einem **Satz**.

Bild

Ausgabeform, die mit einer **Bildbeschreibung** generiert wird.

Bildbeschreibung

Transformationsbeschreibung, die die Ausgabe von Feldinhalten in einer festen Form, dem **Bild**, ermöglicht. In diese **Datenbeschreibung** können Konstanten (Texte) zur Gestaltung des Bildes eingefügt werden.

Bildschirmmaske

Eingabe und Update im Direkt-Dialog.

Datenbeschreibung

Beschreibung des Formats und der Struktur der Daten.

Direkter Suchweg

Der Suchbegriff ist Ordnungsbegriff (Schlüssel) der **Hauptdatei**. Der Zugriff erfolgt direkt auf den Hauptdateisatz.

Feld

Kleinste **Informationseinheit** innerhalb eines **Abschnitts**. **Wiederholfelder** sind möglich. Ein Feld kann **Freitext** enthalten.

Feldmultiplizität

Erste bis n-te Wiederholung eines nur einmal in einer **Datenbeschreibung** definierten **Feldes** in einem variablen **Satz** bzw. **Abschnitt**.

Fenster

vgl. **Relation**

Freitext

Datenfelder, in denen freier Text enthalten ist, verhalten sich im Satz wie jedes beliebige Datenfeld. Die einzelnen Worte (**Stichworte**) können invertiert werden.

Hauptdatei (HD)

Menge von **Sätzen (Zielinformation)**. Bei ISAM-Datei ist der Schlüssel der **Ordnungsbegriff**, bei SAM-oder PAM-Dateien die **Adresse**.

Indirekter Suchweg

Der Suchbegriff ist invertierter Begriff der **Hauptdatei**. Der **Ordnungsbegriff/Adresse** der Hauptdatei wird durch Zugriff auf die **Verweisdatei** gewonnen

BEGRIFFE

Invertierung

Eintrag in der **Verweisdatei**.

Kurzname

Angabe in der **Satzbeschreibung** zu einem **Feld**. Diese Angabe bewirkt eine **Invertierung** des betreffenden **Feldes**. Der Kurzname wird in der **Verweisdatei** als Segmentname verwendet.

Multiplizität

vgl. **Abschnittsmultiplizität** und **Feldmultiplizität**.

Multivariabel (MV-Format)

Formatbezeichnung eines **Satzes**. Die **Abschnitte** eines MV-Satzes tragen eine eindeutige Identifikation (**Abschnittsart**) und können variable Länge haben. Die Formatdefinition erfolgt in der **Datenbeschreibung**.

Nullwertunterdrückung

Numerische **Felder** mit Feldinhalt Null (X'00...00') und alphanumerische Felder mit Feldinhalt Space (X'40...40') werden trotz **Invertierung** nicht in die **Verweisdatei** aufgenommen.

Ordnungsbegriff

Zeichenfolge zur Identifikation eines **Satzes** in einer ISAM-Hauptdatei.

Paßwort

Schlüsselwort für den Zugang zum CIS-System und zu den Daten. Das Paßwort ist gleichzeitig Schlüssel der **Datenbeschreibung**. Mit dem Paßwort können Daten (**Abschnitte** oder/und **Felder**) in Abhängigkeit von der Funktion (Suchen, Ausgabe, Update) geschützt werden.

Relation

oder auch Fenster genannt ist eine virtuelle Datenbank, die aus mehreren Datenbanken aufgebaut wird.

Relationszeile

Satz in einer **Relation**. Der Satzinhalt besteht aus den **Satzinhalten** der verknüpften Datenbanksätze.

Satz

größte **Informationseinheit** einer **Hauptdatei**: gliedert sich entweder in **Felder (V-Format)** oder in **Abschnitte** und Felder (**MV-Format**)

Satzbeschreibung

Datenbeschreibung, die die **Felder** mit ihren Bezeichnungen benennt. Die Satzbeschreibung enthält keine **Transformation**.

Sequentielle Suchfrage

Der Suchbegriff ist weder **Ordnungsbegriff** (OB)/Adresse der **Hauptdatei** (HD) noch invertierter Suchbegriff. Zur Beantwortung der Suchfrage werden alle **Sätze** der HD sequentiell gelesen.

Stichwort

ist ein Teilstring eines alphanumerischen Feldes das **Freitext** enthält.

Suchweg

Zur Beantwortung einer Suchfrage muß das Suchprogramm in Abhängigkeit des Suchbegriffs bestimmte Zugriffe auf die **Hauptdatei** und/oder **Verweisdatei** ausführen. Die zur Beantwortung einer bestimmten Suchfrage notwendigen Zugriffe sind der Suchweg. Es wird zwischen **direktem Suchweg**, dem **indirekten Suchweg** und der **sequentiellen Suchfrage**, wobei Mischformen bei Verknüpfungen auftreten können, unterschieden.

Transformation

Umformung der Struktur und des Formats bereits vorhandener Daten eines **Satzes**.

Transformationsbeschreibung

Datenbeschreibung, die eine **Transformation** der **Sätze** ermöglicht.

Variabel (V-Format)

Formatbezeichnung eines **Satzes**. Die **Felder** eines V-Satzes tragen eine eindeutige Identifikation (Feldbezeichnung) und können variabel lang sein. Das letzte Feld eines V-Satzes kann ein **Wiederholfeld** sein. Die Formatdefinition erfolgt in der **Datenbeschreibung**.

Verweisdatei (VD)

Die Verweisdatei ist eine ISAM-Datei, deren Schlüssel sich aus Daten der **Hauptdatei** (Feldinhalt) und dem **Kurznamen** in der **Datenbeschreibung** zusammensetzen.

Verweisdateisatz

enthält im Datenteil die **Ordnungsbegriffe/Adressen** der **Hauptdatei**. Gegebenenfalls wird der VD-Satz in Untersätze zerlegt, falls der **Satz** infolge der Vielzahl Ordnungsbegriffe/Adressen die maximale Länge von 2048 Bytes überschreitet.

Wiederholabschnitt

Abschnitt, der einmal in der **Datenbeschreibung** definiert wird und beliebig oft auftreten kann, (vgl. **Abschnittsmultiplizität**).

Wiederholfeld

Feld, das einmal in der Datenbeschreibung definiert wird und beliebig oft auftreten kann.

Zielinformation

Daten, die über eine Suchfrage selektiert werden und deren Anzahl sich auf **Sätze** bezieht.

Zielpunktliste (ZPL)

Die Zielpunktliste wird aufgrund von Suchfragen gebildet. Sie enthält **Ordnungsbegriffe** in aufsteigend sortierter Reihenfolge. Über die Ordnungsbegriffe bzw. die **Adressen** werden die **Sätze** der **Hauptdatei** adressiert.