

Vorlesungsteil: Interaktive Grafik (zu Kapitel 5)

Eingabe grafischer Daten für geometrische Objekte

1. Eingabeprozess:

Die Eingabe von Daten für geometrische Objekte ist ein aufwändiger Teilprozess, der nur gerechtfertigt ist, wenn nachfolgende Teilprozesse wie Speicherung, Verarbeitung, Ausgabe und mehrfache Wiederverwendung sich so gestalten lassen, dass eine hohe Gesamtarbeitsproduktivität erreicht werden kann.

Eingabemethodik:

Bezogen auf die Rolle des Konstrukteurs beim interaktiven Konstruieren am Computer gibt es hinsichtlich der Eingabe der Objekte zwei prinzipielle Arbeitsweisen:

1. **die Eingabesprache:** Der Konstrukteur gibt Kommandos zum Zeichnen ein.
2. **Dialog:** Der Konstrukteur erhält vom Computerprogramm ein Angebot möglicher grafischer Leistungen (ein Menü), aus dem er innerhalb festgelegter Grenzen auswählen kann.

Eingabesprache: Diese Arbeitsweise erfordert eine genaue Kenntnis der Eingabesprache (Syntax, Semantik) und hat den Vorteil der Flexibilität beim Konstruieren, wenn die Eingabesprache gut erlernbar und flexibel ist. Der Computer interpretiert die Kommandos mit Hilfe eines Kommandointerpreters, der schnell arbeiten muss.

Dialog: Diese Arbeitsweise erspart dem Konstrukteur, eine Eingabesprache beherrschen zu müssen. Er erhält eine Hilfestellung beim Konstruieren, muss aber das System handhaben können. Das ist bedienaufwendiger und auch starrer.

2. Wie wird das geometrische Objekt beschrieben ?

1. durch interaktive Erzeugung grafischer Primitive
2. durch Bezugnahme auf bereits gespeicherte Objekte (Makros)
3. durch Variantentechnik
4. durch Digitalisieren einer Vorlage eines geometrischen Objektes

2.1 Die interaktive Erzeugung grafischer Primitive

2.1.1. Kommandosprache (Eingabesprache)

Kommando

AN/N	nutzer //	Anmeldung des Konstrukteurs
AN/P	projekt1.BIB//	Name des Projektes
@Bild=B/ANF//		Definition einer globalen Zeichnung
Fundament=B/ANF//		Definition einer aktuellen Zeichnung

Erzeuge Linie waagrecht von P1 nach rechts mit Länge=20

L1= L/V P1 W R L=20 Diese Linie heißt L1

Masken und Fenstersätze

```
M1=M/L 1: ,1,2//   Stift 1 Linienbreite 1
//               Linienart 2 -----
M2=M/L 2: ,3,1//   Stift 2 Linienbreite 3
//               Linienart 1 _____
F1/=F/0.,0., 1:100,1:100 //   Maßstabsfestlegung
Punkte/Linien/Skalare
P1=P/0.,0.//
P2=P/0.,500.//
P3=P/300.,500.//
P4=P/300.,700.//
P6=P/2000.,700.//
P7=P/2000.,0.//
L1=L/V  P1 B P2 M=M2//
L2=L/V  P2 B P3 M=M2//
L3=L/V  P3 B P4 M=M2//
L4=L/V  P4 B P6 M=M2//
L5=L/V  P1 B P7 M=M2//
L6=L/V  P6 B P7 M=M1//
S1=SK/X P4+200.//
S2=P/Y P4//
P5=P/S1,S2//
```

Konturen bilden (Bauteilbildung)

```
KO1=KO/L1 L2 L3 L4 L5 L6//Kontur KO1 besteht aus den
Linien L1 bis L6
```

```
KO2=KO/KO1 T=S L6 //Kontur KO2 entsteht durch Spiegeln an
der Achse L6
```

```
:
:
:
```

```
A/Fundament F=F//   Ausgabe des Fundamentes im M 1:100
```

Vorteil:

Mittels dieser Eingabesprache ist die Herstellung einer parametrisierten Konstruktion möglich:

- Entwurf einer Kommandofolge wie oben dargestellt
- Parametrierung der Konstruktion durch Einführung von Variablen, Dialog zur Eingabe der Variablen
- Test der Kommandofolge mit Parametern
- Speicherung der Kommandofolge als Datei auf Datenträger
- Aufruf der Kommandofolge mit aktuellen Parametern ergibt eine spezielle, auftragsgebundene Konstruktion

2.1. 2 Menütechnik

Am weitesten verbreitet ist in interaktiven Software-Systemen die Menütechnik. Es wird ein hierarchisch aufgebautes Menü am Bildschirm in Form von Text, Bildsymbolen (Icons) und Funktionstasten (Hotkeys) angeboten. Auch bei Digitalisiergeräten findet man es in Form eines Tableaus (Menüschablone). Die einzelnen Funktionen können per Tastatur, Funktionstasten bzw. meist mit Maus, Lichtstift, Fadenkreuz ausgewählt werden, wobei dies möglichst frei beweglich erfolgen sollte. Das Menü sollte übersichtlich und leistungsfähig sein und den Nutzer zielgerichtet führen!

Eingabeprobleme: Die interaktive Eingabe von graphischen Primitiven ist in jedem Fall (Eingabesprache oder Dialog) ein aufwändiger Prozeß. Sie muß oft koordinatenorientiert erfolgen, was nicht unbedingt den Konstruktionsgewohnheiten entspricht. Der Konstrukteur muss die interaktive computergestützte Konstruktion erlernen!

Die genaueste Eingabe erfolgt durch Zahleneingabe über die Tastatur.

Fadenkreuz- bzw. Mauseingabe kann meist erst erfolgen, wenn auf bereits gezeichnete Punkte Bezug genommen werden kann (Punktfangmechanismus).

Es gibt Konstruktionsaufgaben, bei denen der Bildschirm wegen seiner geringen Größe nicht immer ideal ist (z. B. bei Aufgaben, bei denen man das Gesamtbild und das Detail zur gleichen Zeit sehen muss!). Die Arbeit ist in jedem Fall bedienaufwendig, die Einsparung gegenüber manueller Arbeitsweise liegt weniger bei der Erstzeichnung, als bei der vielfachen Verwendung und Manipulierung der Zeichnungen und der digitalen Datenbasis, die sofort für Berechnungen nutzbar sein sollte.

2.1.3 Erzeugung von Makros

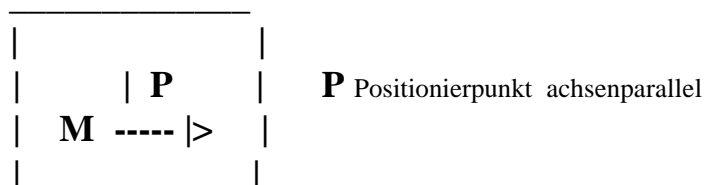
Der Entwurf neuer Bauteile und Erzeugnisse ist in vielen Branchen kein völlig individueller Vorgang, der beim „leeren“ Zeichenbrett beginnt. DIN-Vorschriften, Standards, betriebs-spezifische Festlegungen erlauben und erfordern die Benutzung bereits konstruierter Bauteile, die als Zeichnungen auch bei manueller Arbeitsweise vorlagen. Dem wird bei computer-gestützten Arbeitsweisen durch die Makro- und die Variantentechnik entsprochen.

MAKRO:

Ein Makro ist ein in der Gestalt und in der Dimension fest abgespeichertes Bauteil. Es wird in die aktuelle Zeichnung durch Aufruf von einem Direktzugriffs-Speicher (Platte, Diskette) geholt und nur noch in der Zeichnung positioniert (Transformationen: Drehen, Spiegeln, Duplizieren).

Beispiele: Sinnbilder, Symbole in der Elektrotechnik, Bauwesen, Gebäudeausrüstung

TGL 31101 Motorpumpe mit Grundplatte



Arbeitsgänge zur Herstellung eines Makros

1. Interaktive Erstellung des Makros als Zeichnung
2. Abspeicherung des Makros unter einem Namen `makro -n`
3. Entwurf eines Menueblattes als Bildschirmtableau oder Schablone
4. Abspeicherung des Menueblattes unter einem Namen `menue_m`

5. Einordnung des Makros `makro_n` in das Menueblatt
6. Herstellung der Funktionalität zur Auswahl eines Makros `makro_n` aus dem Menueblatt in die aktuelle Zeichnung

2.1.4 Variantentechnik

VARIANTEN: Es wird davon ausgegangen, dass Teilefamilien bestehen. Es gibt Maßvarianten und Gestaltsvarianten.

Maßvarianten: Gestalt stets konstant, Maße ändern sich.

Gestaltsvarianten: Gestalt und Dimension (Maße) ändern sich. Es ist ein Grundprogramm erforderlich, das in einem Eingabedialog die Parameter entgegennimmt, sie auswertet (Alternativen, Verzweigungen) und die entsprechenden Zeichenbefehle parameterabhängig generiert. Der Aufwand ist sehr hoch, die Programmiersprache muss die Elemente einer modernen problemorientierten Programmiersprache enthalten!

assoziative Bemaßung: Die Bemaßungs-Software ist **assoziativ** zu gestalten. Das heißt, **automatische** Änderung der Maßkette mit Änderung der Gestalt.

2.1.5 Digitalisierung geometrischer Objekte

Ingenieurgeodäsie ---- > Geländemodell ----- > Strassenbauprojektierung ----- > Trassierung
---- > Berechnungen

Geländemodell ----- >

Trassierungsentwurf --- > Digitalisiergerät + Computerprogramm ----- >
Datei----- > Berechnungen (individuelle Programme)

Arbeitsgänge im Computerprogramm:

1. Eröffnung des GKS
2. Vereinbarung einer Datei `name.typ`
3. Digitalisieren Funktionsauswahl (Punkt, Linie usw.)
4. Eingabe eines Punktes x, y als ASCII-Zeichenkette
5. Interne Umwandlung in Gleitkommazahlen
6. Runden der Zahlen
7. Abspeicherung der Zahlen in der Datei

Auswertung der Dateien in individuellen Programmen:

1. Kenntnis des Dateiaufbaus
2. Abschätzung des Digitalisierfehlers auf Brauchbarkeit der Zahlen für die spezifische Berechnung!

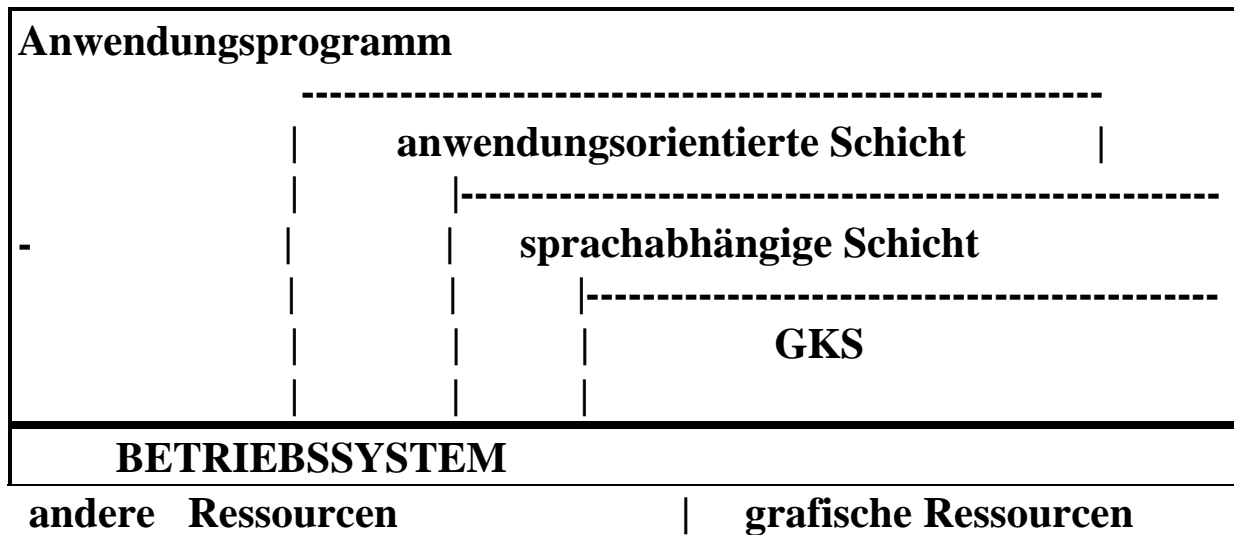
Vorlesungsteil: Grundlagen der interaktiven Grafik

Teil: Standardisierung in der 2D-Grafik

Das Grafische Kernsystem (GKS)

Durch die GKS-Norm (ISO 7942) wurde eine genormte Schnittstelle für die Vielfalt an grafischen Anwenderprogrammen und grafischen Geräten geschaffen.

In Form eines Schichtenmodells wird die Grafik in ein Anwenderprogramm eingebettet:



Das grafische Kernsystem besteht aus einer Reihe von Funktionen, die zunächst sprach- und geräteunabhängig im Standard definiert sind.

Dabei wird der Gesamtumfang an Funktionen in mehrere Leistungsstufen unterteilt:

Stufe :	Eingaben:		
	A	B	C
0	x 90 Funktionen		
1			
2			x 180 Funktionen

Je nach anliegenden Anwenderanforderungen und Geräteausstattung kann eine spezielle Leistungsstufe ausgewählt werden.

Die Funktionen werden in einem konkreten implementierten System des GKS sprachabhängig geliefert: Fortran, Pascal, C++.

Leistungsumfang der Funktionen

Die Funktionen behandeln: Darstellungselemente, Attribute, grafische Arbeitsstationen, Transformationen, Bildstruktur, grafische Eingabe, Bilddatei, Erfragefunktionen (Inquire-funktionen).

Zu einer Beschreibung der Funktionen gehört:

der Name der Funktion,

die Anzahl, die Datentypen und die Bedeutung der Parameter,

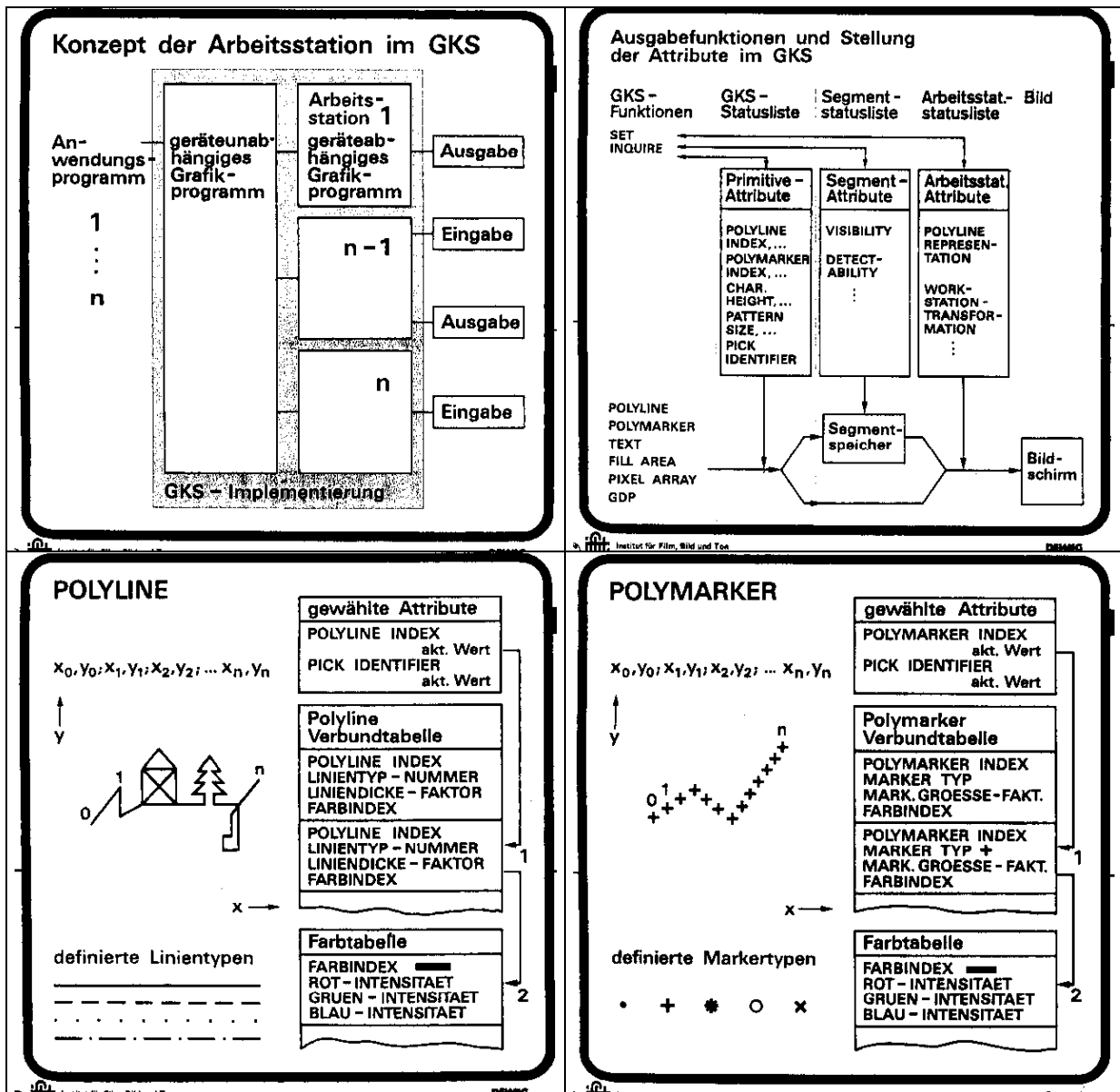
die Wirkung der Funktion (Rückgabewerte) ----- Änderung der grafischen Darstellung, Rückgabe von Werten an das Anwenderprogramm, Änderung des GKS-Zustandes, und die Fehlerbedingungen.

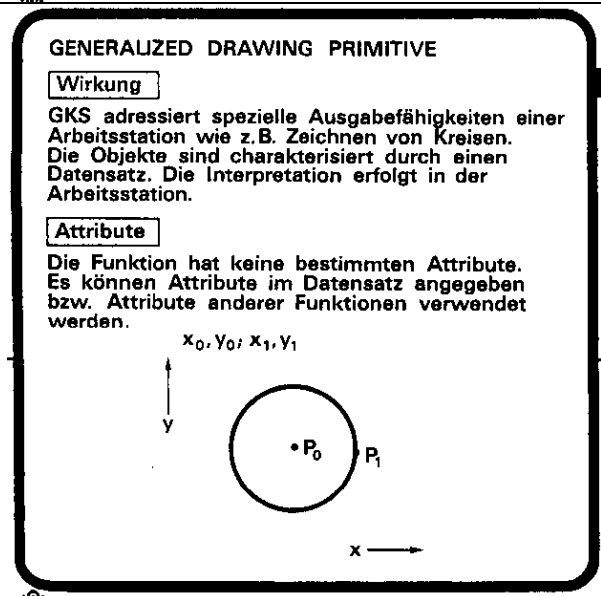
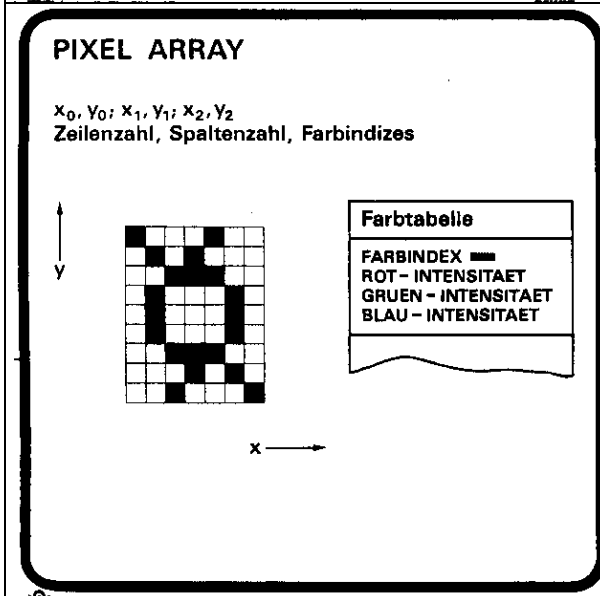
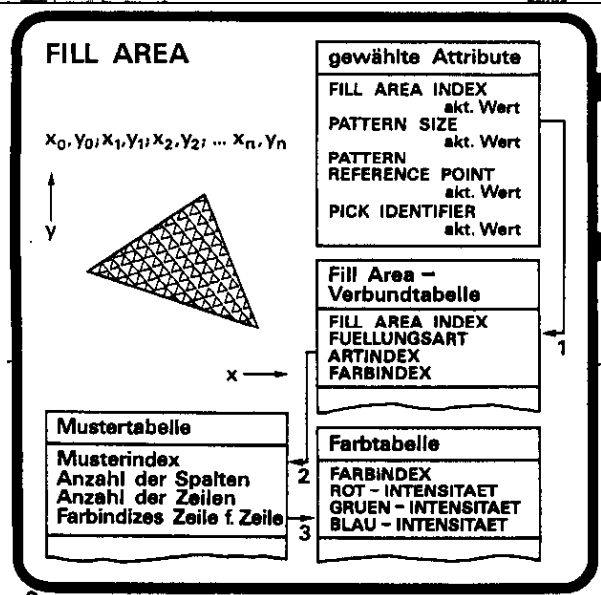
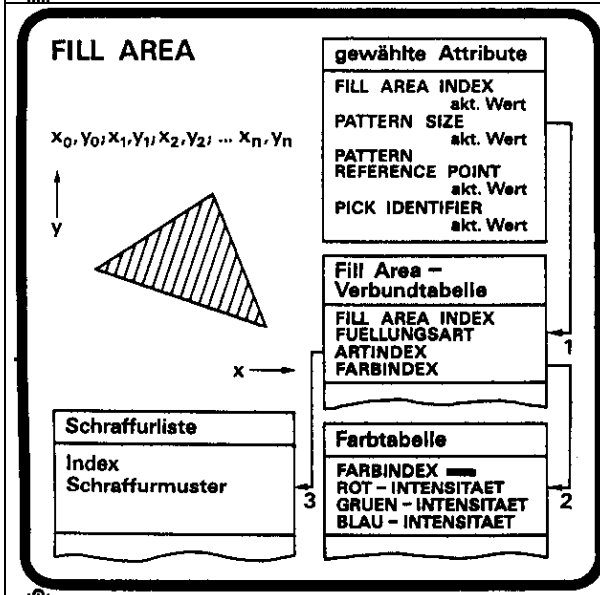
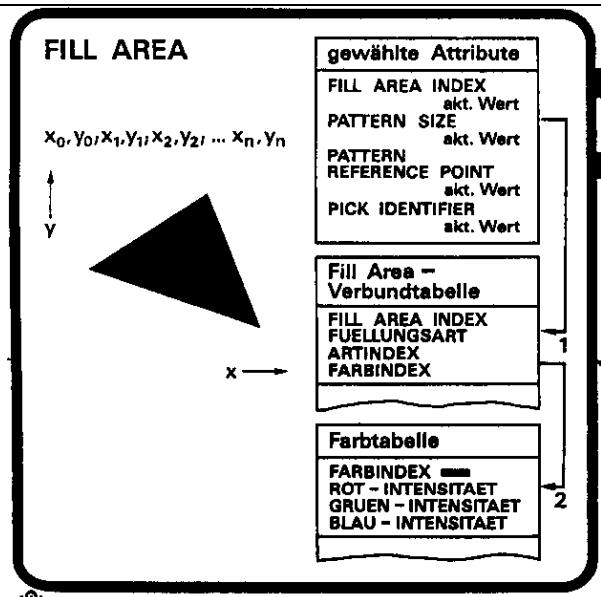
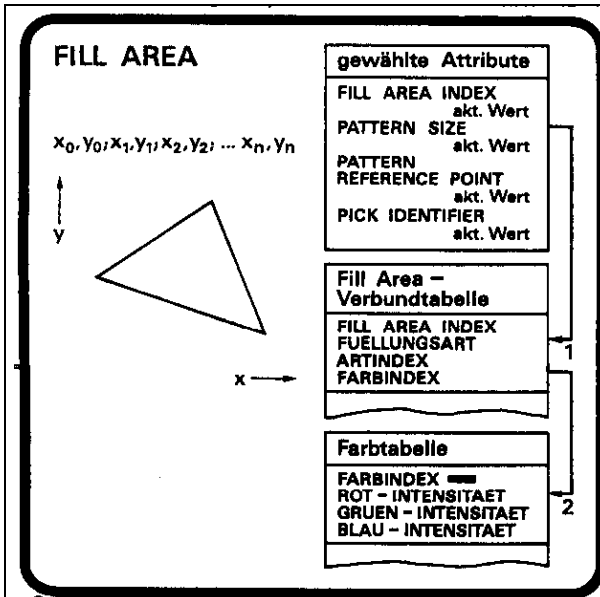
Darstellungselemente:

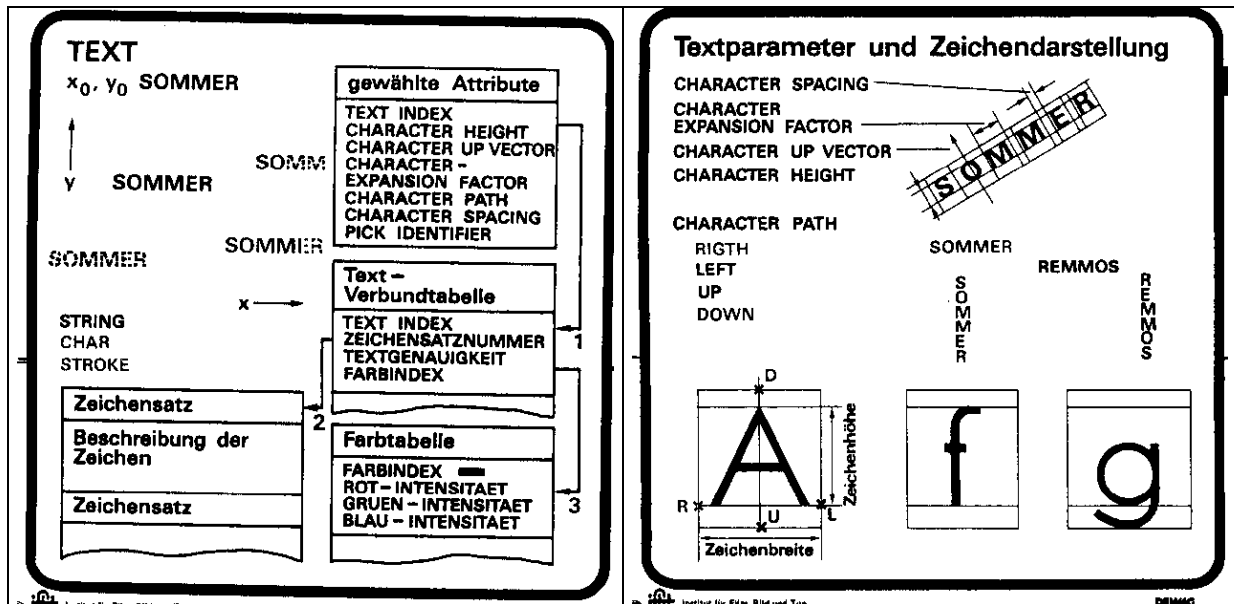
Das GKS definiert die folgenden Darstellungselemente :

- Polyline
- Polymarker
- Text
- Füllgebiet (*fill area*)
- Zellmatrix (*cell array*)
- Allgemeines Zeichenelement (*GDP General Drawing Primitive*)

Die entsprechenden Funktionen dazu zeichnen das Primitiv auf eine interne Darstellungsfläche.





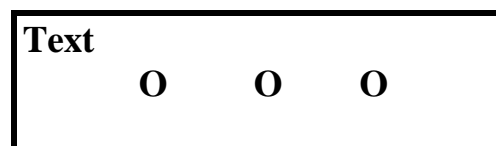


Attribute:

Jedes Darstellungsprimitive kann mit sogenannten Attributen artspezifisch näher beschrieben werden (Linienart, Linienbreite, Farbe usw.). Dabei können diese Attribute global oder arbeitsstationsabhängig vergeben werden. Sie werden in sogenannten Bündeltabellen zusammengefasst und gespeichert. Über eine Funktion, die einen Bündelindex setzt, kann zu den Einträgen einer Bündeltabelle zugegriffen werden. Damit werden gebündelte Attribute z.B. für ein Polygon als aktueller Satz von Attributen ausgewählt, der so lange aktiv bleibt, bis ein neuer Index ausgewählt wird.

Segmente:

Alle Darstellungselemente einer Zeichnung können beliebig zu einer größeren Einheit, dem *Segment*, zusammengefasst werden.



Ein Segment kann

- geschlossen transformiert werden (drehen, spiegeln usw.)
- wieder in Primitive zerlegt werden
- identifiziert werden (PICK-Identifikator)
- Prioritäten bei Überlagerung erhalten
- weitere Segmente in sich aufnehmen
- in einem Segmentspeicher abgelegt werden

Koordinatensysteme und Transformationen:

Das GKS definiert mehrere Koordinatensysteme und stellt Funktionen zu ihrer gegenseitigen Transformation bereit:

Weltkoordinaten (WKS):

Der Anwender entwirft eine Zeichnung zunächst in Weltkoordinaten. Das sind maßstabslose Zahlenpaare, je nach Fachgebiet und Problem (z.B. auf der x -Achse 100 Einheiten, die mm, cm, m oder Sekunden bedeuten können und auf der y -Achse 50 Einheiten, die auch mm, cm, m oder technische Einheiten (kp/cm^2) bedeuten können). Durch Definition eines Weltkoordinatenfensters (x_u, y_u, x_o, y_o) begrenzt der Anwender dieses Fenster auf die Minimal- bzw. Maximalwerte seines Problems.

Normalisierte Koordinaten (NKS):

Durch Festlegung eines Fensters, dessen Koordinaten 0,0 bis 1,1 sein können, werden alle Weltkoordinaten auf normalisierte Koordinaten umgerechnet und damit auf dieser virtuellen Zeichenfläche dargestellt.

Gerätekoordinaten (GKS):

Erst für die Ausgabe der Zeichnung auf ein konkretes Ausgabegerät wird ein gerätespezifisches Fenster definiert ($x_{gu}, y_{gu}, x_{go}, y_{go}$).

Die Transformationen **WKS** ---- \rightarrow **NKS** ----- \rightarrow **GKS** für die Ausgabe
GKS -- -- \rightarrow **NKS**----- \rightarrow **WKS** für die Eingabe

werden automatisch von Transformationsfunktionen übernommen.

Logische Fenster werden als **Window**, Gerätefenster als **Viewport** bezeichnet und erhalten bei Definition eine Fensternummer. Dabei ist durch die Fensterkoordinatendefinitionen nur auf eine verzerrungsfreie Bildwiedergabe zu achten, wenn das problemseitig erforderlich ist (gleiche Seitenverhältnisse der Fenster). Oft wird dadurch die Rasterpunktfläche eines Bildschirms nicht voll ausgenutzt.

Arbeitsstationen:

Das GKS definiert sogenannte Arbeitsstationen, die als Eingabe-, Ausgabe- oder E/A- Stationen arbeiten können.

Ihre spezifischen Fähigkeiten (Ausstattungen wie Stiftanzahl, Farben, ...) werden in Arbeitsstations-Beschreibungstabellen je Arbeitsstation gespeichert. Eine solche Arbeitsstation kann geschlossen, geöffnet oder aktiv sein. Wenn sie in aktivem Zustand ist, wird die Zeichnung auf diese Arbeitsstation gerätespezifisch ausgegeben.

Als Arbeitsstationen können folgende Geräte dienen:

Tastatur E

Platte (Metafile) E /A

Bildschirm A

Zeichentisch A

Digitalisierbrett E

Erfragefunktionen (*Inquire*-Funktionen):

Das GKS stellt eine Reihe von Funktionen zur Verfügung, die den Zustand des GKS-Systems selbst abfragen und dem Anwender bei Bedarf mitteilen (gesetzter Linientyp, aktive Arbeitsstationen, usw.)

Sprachschale des GKS:

Zur Demonstration des GKS wird ein Anwenderprogramm dargestellt, das in der Programmiersprache FORTRAN für eine konkrete Anlage programmiert wurde:

```

PROGRAM BEISPIEL1
C Erzeugung einer Beispielgrafik
C Vereinbarungen
REAL PF(10)
BYTE WSNR,WSTYP, LI
C Verarbeitung
CALL GKSOPN // Eröffnung des GKS
WRITE (5,1)
1 FORMAT ('Bitte grafische Arbeitsstationsnr,
Arbeitsstationstyp waehlen: ` /
* ` Grafisches Terminal .....1,3 ` /
* ` Zeichentisch DZT.....4,6 `/
* ` Digigraph .....2,4 `/
* ` Plotter .....4,8 `/ )
READ(5,2) WSNR,WSTYP
2 FORMAT (I1,I1)
CALL GKSOPW(WSNR,WSTYP) // Eröffnung einer grafischen
// Arbeitsstation
CALL GKSAPW(WSNR) // Aktivierung einer grafischen
// Arbeitsstation
// Festlegung von Werkzeugen und Linienarten,
// arbeitsstationsbezogen
CALL GKSCR(WSNR,1,1,1,1) Werkzeug 1
CALL GKSCR(WSNR,2,1,1,2) Werkzeug 2
CALL GKSLR(WSNR,6,1,3,1) Linienart 6, Volllinie, dreifach
// breit, Werkzeug1
CALL GKSLR (WSNR,7,2,5,2) Linienart 7, Strichlinie, 5fach
// breit, Werkzeug 2
CALL GKSWN(1,0,3000, 0,3000) // Weltkoordinatenfenster 1 mit
// Eckpunkten
CALL GKSVP (1,0,1,0,1 ) // Viewport im NDC-Raum für
// Fenster 1 mit 0,1,0,1
CALL GKSELNT(1) // Normalisierungstransformation
// für Fenster 1
CALL GKSWKWN(1,0,1,0,1) // Arbeitsstationsfenster für
// gesamten NDC-Raum definiert
CALL GKSWKVP(1,0,30,0,30) // Arbeitsstationsviewport
// innerhalb der max. Zeichenfläche des Ausgabegerätes z. B. 30 x
// 30 cm
// Bereitstellung von grafischen Daten (Polygonzugkoordinaten)
10 WRITE (5,3)
3 FORMAT (' Bitte zwei Eckpunkte für Rechteck xu,yu,xo,yo:')
READ(5,4 ,ERR=10) PF(1),PF(2),PF(5),PF(6)

```

```

4 FORMAT (4E12.5)
PF(9)=PF(1)
PF(10)=PF(2)
PF(3)=PF(5)
PF(4)=PF(2)
PF(7)=PF(1)
PF(8)=PF(6)
WRITE(5,5)
5 FORMAT ('Bitte Linienart waehlen: \' /
• \' Volllinie 1fach breit .....1' /
• \' Strichlinie 3fach breit.....2' / )
READ(5,6,ERR=11) LI
6 FORMAT (I2)
CALL GKSLIX(LI)           // Setze Linienindex
CALL GKSPLI(5,PF)        // Zeichne Polylinie
CALL GKSPMA(5,PF)        // Zeichne Polymarker

C ABSCHLUSS
CALL GKSDEW(WSNR)       // Deaktiviere Workstation
CALL GKSCLW(WSNR)       // Schliesse Workstation
CALL GKSCLS             // Schliesse GKS
STOP
END

```

* Beachte: die Namen der GKS-Routinen beziehen sich auf eine konkrete GKS-Implementation von 1990. Sie sind den Namen des Standards jedoch ähnlich.

Interaktives Zeichnen einer Linie mittels GKS

Menue: Auswahl:
Linie
Symbol
Text
Füllgebiet
Zellfeld
GDP

Submenue "Linie":

Anfangspunkt----- x,y----- > Tastatureingabefunktion des GKS

Fadenkreuz--- > Fadenkreuzeingabefunktion 1

Pick----- > Fadenkreuzeingabefunktion 2

Endpunkt-----x,y

Fadenkreuz

Pick

Menue: Linienarten:

...Volllinie----- > setzt Linienindex GKSLIX

Strichlinie ----- > (als Zugriff zu Bündeltabellen)

Punktlinie ----- >

Funktionen, die am interaktiven Zeichnen der Linie beteiligt sind:

- 1. Eingabefunktion für Anfangspunkt je nach Eingabegerät und Modus**
- 2. Eingabefunktion für Endpunkt je nach Eingabegerät und Modus**
Es wird x1,y1,x2,y2 in einem Datenfeld PF abgelegt!
- 3. Funktion zum Setzen des Linienindex GKSLIX (Nr)**
`CALL GKSLIX (1)`
- 4. Funktion zum Zeichnen der Linie GKSPLI durch 2 Punkte des Punktfeldes PF**
`CALL GKSPLI (2,PF)`
Die Linie wird auf der aktiven Arbeitsstation ausgegeben.