



## 2 Medientypen

### 2.1 Text

# Schrift und Alphabet



- Schrift:  
System zur Lesbarmachung der gesprochenen Sprache
  - d.h. Transfer von kontinuierlicher Sprache in diskrete 2-dimensionale räumliche Darstellung
- Alphabet:  
Festlegung der Codierungselemente, der Symbole



# Schrift und Alphabet



## ■ Historie

### ■ Entwicklung der Schrift

- 4000 v. Chr.                   sumerische Bilderschrift
- 3000 v. Chr.                   Hieroglyphen, Keilschrift
- 2000 v. Chr.                   Buchstabenschrift (semitisches Uralphabet)

### ■ Codierungsvorgang

- Handschrift                   (mit Feder, Pinsel, Meißel)
- Druckverfahren               (J. Gutenberg, 1455)
- Schreibmaschine              (P. Mitterhofer, 1864)
- Computertastatur als Eingabegerät zur digitalen Speicherung der Texte
- Spracherkennungs-Software



# Schrift und Alphabet



- Phonetisches Alphabet
  - Buchstabenalphabete der westlichen Welt
- Silbenalphabet
  - Koreanisches Hanguk
- Ideographisches Alphabet
  - Symbole repräsentieren ganze Worte oder Konzepte
  - z.B. Kanji
- Ziffern, Punktierungszeichen und ähnliches gehören ebenfalls zum Alphabet



# Schrift und Alphabet



- Digitale Repräsentation
  - Abbildung der (abstrakten) Symbole des Alphabets auf digitale Werte
    - Das Bild heißt Zeichensatz – Character Set
    - Der Wertebereich (das Alphabet) heißt Symbolrepertoire – Character Repertoire
  - Für jedes Symbol im Repertoire definiert der Zeichensatz einen Codewert bzw. Codepunkt



# Codierungsstandards



- ASCII
  - American Standard Code for Information Interchange
  - Seit den 70ern der dominante Zeichensatz
    - Außer IBM: EBCDIC (Extended Binary Coded Decimal Interchange Code)
  - 7 Bit Code → 128 Codepunkte
  - 0 .. 31 Steuerzeichen,  
32 .. 127 druckbare Zeichen
  - Geeignet für US-English



# Codierungsstandards



- ASCII
  - 1972 Standardisierung als ISO 646
  - Unterschiedliche länderspezifische Varianten mit unterschiedlichen spezielleren Symbolen
    - z.B. ISO 646-US, ISO 646-UK
  - 8. Bit diente ursprünglich als Paritätsbit zur Fehlerkorrektur
    - Mit verbesserten Übertragungsmöglichkeiten unnötig
    - → dieses Bit kann den Code erweitern
  - Herstellerspezifische Belegungen der Codepunkte 128 .. 255 entstanden



# Codierungsstandards



- ISO 8859-X

- Standardisierung in den 80ern als Multipart-Standard
- z.Zt. 10 standardisierte 8-Bit-Erweiterungen von 7-Bit ASCII um länderspezifische Zeichen
  - ISO 8859-1 (ISO Latin1) westliche Alphabete
  - ISO 8859-2 (ISO Latin2) osteuropäische (tschech., slowakisch, kroatisch)
  - ISO 8859-5 kyrillisch
  - ISO 8859-6 arabisch
  - ISO 8859-7 griechisch
  - ISO 8859-8 hebräisch
  - ISO 8859-0 (ISO Latin0) Latin 1 + Eurosymbol



# Codierungsstandards



- ISO 8859-X
  - Nachteile:
    - Es gibt immer noch die herstellerspezifischen Abweichungen
    - Multilinguale Anwendungen sind nicht möglich
    - Ideographische Alphabete werden nicht unterstützt
- Ein 8 Bit Code hat einfach zu wenige Codepunkte



# Codierungsstandards



- ISO 10646
  - 32-Bit Code (1991)
  - Gruppierung von Zeichensätzen als Hyperkubus
    - g = Gruppe
    - p = Ebene (Plane)
    - r = Zeile (Row)
    - c = Spalte (Column)
    - Mit Wertebereich jeweils 0..255
  - Symbole werden als Tupel (g,p,r,c) angegeben
  - (0,0,0,\*) ist ISO Latin1
  - (0,0,\*,\*) ist die Basic Multilingual Plane
    - identisch mit Unicode



# Codierungsstandards



- ISO 10646
  - Für Chinesisch, Japanisch und Koreanisch werden die Zeichensätze unifiziert (CJK consolidation)
    - D.h. hier ist keine Multilingualität möglich
  - 39000 Codepunkte sind definiert
  - 6400 Codepunkte sind für private Nutzung



# Codierungsstandards



- Unicode
  - 16-Bit Code
  - Verwendet u.a. in HTML, XML, Java
  - Beinhaltet die gängigsten Alphabete für Sprache
    - Lateinisch, Griechisch, Kyrillisch, Armenisch, Hebräisch, Arabisch, Devanagarisch, Bengalisch, Gurmukhisch, Gujaratisch, Oriya, Tamilisch, Telugisch, Kannada, Malayalam, Thai, Lao, Georgisch, Tibetanisch
    - Chinesische, japanische und koreanische Ideogramme
    - japanische und koreanische phonetische und Silbenskripte
    - Zeichensetzungen, Akzente und Tilden, mathematische Symbole, Dingbats-Symbole



# Codierungsstandards



- Zeichenkodierung – Character Encoding
  - Abbildung der Codepunkte auf eine Bytesequenz zur Speicherung bzw. Übertragung
  - Quoted-Printable (QP)
    - 8-Bit Code in 7-Bit darstellen
    - Zeichen zwischen 128 und 255 werden als 3 Byte dargestellt
    - Z.B. é : 233 ( $E9_{16}$ ) in ISO Latin 1 → é : = E9 (= : = 3D)



# Codierungsstandards



- Zeichenkodierung – Character Encoding
  - ISO 10646 Kodierung
    - UCS-4      1:1 Abbildung auf 4 Byte
    - UCS-2      nur die niedrigen 2 Byte werden kodiert (entspricht Unicode)
  - Unicode Transformation Formats, UTF
    - UTF-7      alles kodiert in ASCII, ähnliches Prinzip wie Q-P
    - UTF-8      7-Bit ASCII werden als 1 Byte kodiert, alles andere in bis zu 6 Bytes mit 8.Bit = 1
    - UTF-16     Paare von 16-Bit Werten können als einzelner 32-Bit Wert zusammengefasst werden;  
dies erschließt 15 weitere Ebenen aus ISO 10646 (ca. 1 Mio. Symbole)



# Schriftart



- Glyphen dienen als visuelle Repräsentation von Symbolen
- Schriftart - Font
  - Zusammenstellung von zueinander passenden Glyphen zu einem Alphabet
  - früher Bleiletern
  - heute Beschreibung der grafischen Darstellung
- Schriftart-Familie
  - Passende Glyphen mit unterschiedlichen Größen und Schnitt





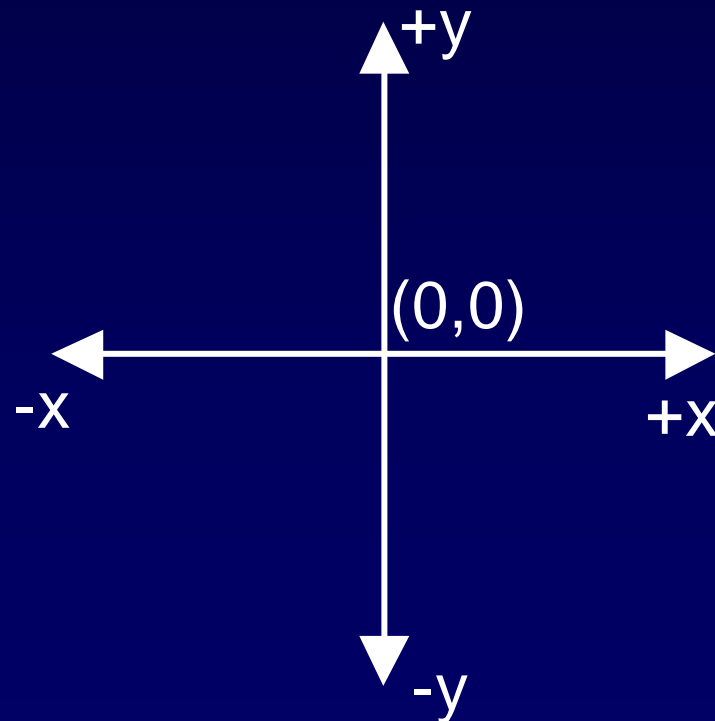
Einschub

Koordinaten und Fenster

# Koordinatensysteme



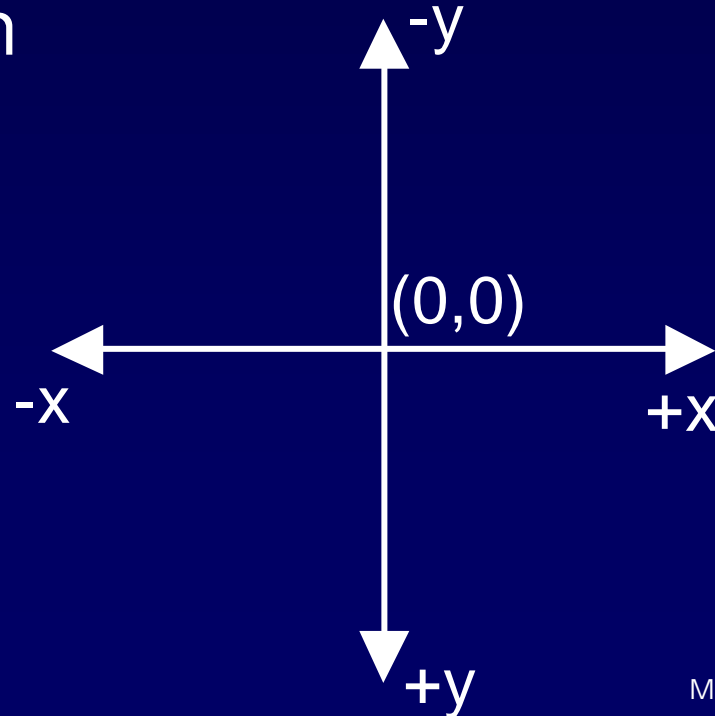
- Kartesisches Koordinatensystem
- spannt 2-dimensionalen Vektorraum auf



# Bildschirm-Koordinatensystem



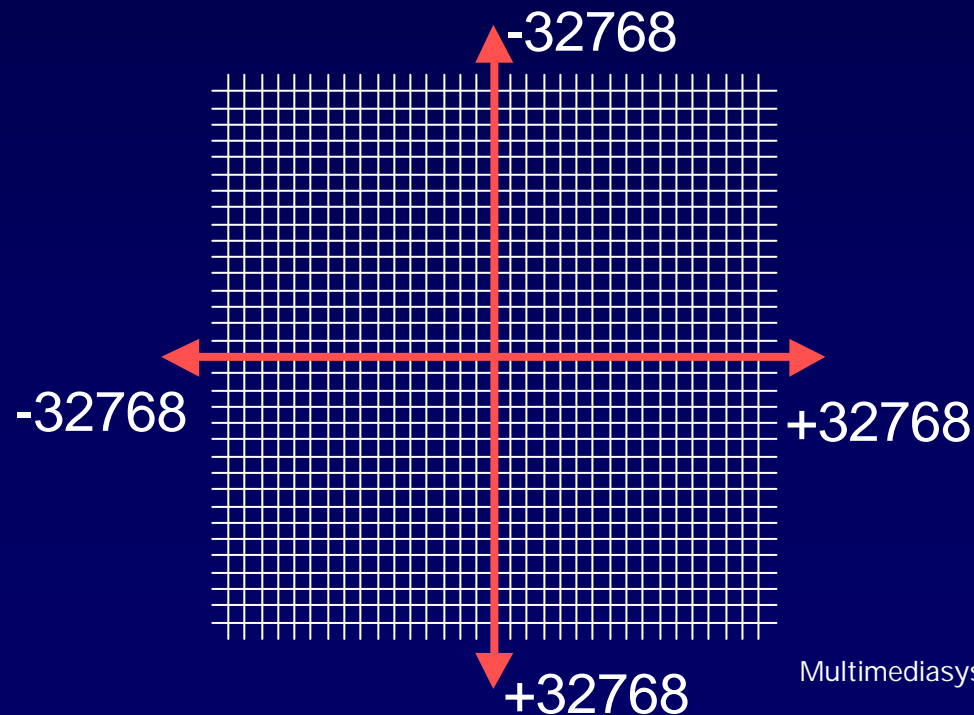
- meist gespiegeltes Kartesisches Koordinatensystem
  - der Elektronenstrahl des Bildschirms beschreibt von links oben nach rechts unten



# Bildschirm-Koordinatensystem



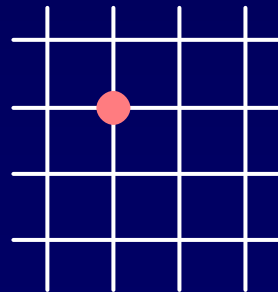
- Logisches Koordinatensystem
  - unendlich dünne Koordinatenlinien
  - endlicher Koordinatenbereich
  - keine feste Zuordnung zum Bildschirm



# Koordinatenpunkt und Bildpunkt



- Ein Koordinatenpunkt ist ein Kreuzungspunkt von zwei Koordinatenlinien
  - als geometrische Abstraktion
  - typischerweise  $2^{32}$  Punkte in einer Koordinatenebene
  - Punkt selbst ist unendlich klein

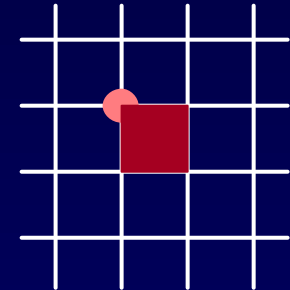


# Koordinatenpunkt und Bildpunkt



- Bildpunkt

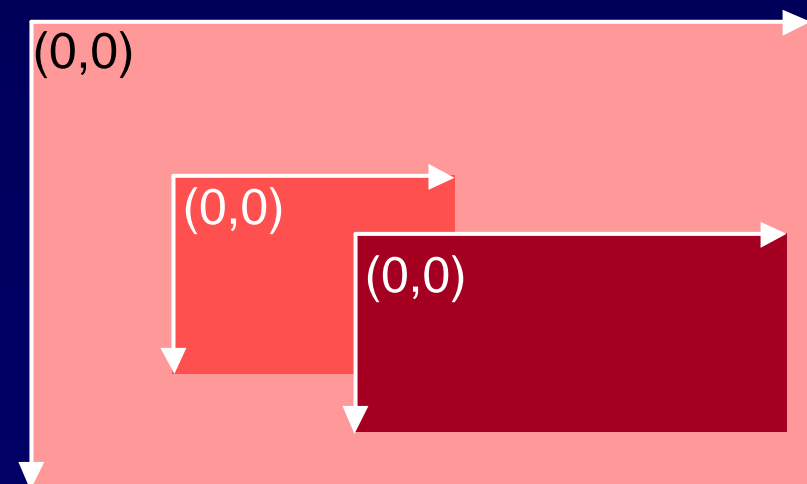
- als Pixel in einer BitMap bzw. PixelMap
- begrenzt durch vier Koordinatenpunkte
- Position meist rechts unter dem Koordinatenpunkt
- gemessen in Dots per Inch (dpi)
- unterschiedliche Abbildungen zwischen logischem Koordinatensystem und physikalischen Bildpunkten



# Fenster und Koordinaten



- globale und lokale Koordinatensysteme
  - Bereich von Bildpunkten (BitMap, PixelMap, der Bildschirm) besitzt ein globales Koordinatensystem
  - die auf einer Pixelmap arbeitenden Zeichenbereiche haben eigene lokale Koordinatensysteme



# Fenster und Koordinaten



- Koordinatenoperationen
  - Verschieben des Betrachtungsfensters
  - Verschieben des Ursprungs
- Koordination zwischen Fenstern übernimmt der Window-Manager





Ende des Einschubs  
Koordinaten und Fenster

# Schriftart



- Schriftgröße
  - Ursprung im Bleisatz
  - Größe festgelegt durch Kegelhöhe des Bleiletters
    - d.h. etwas größer als das Druckbild
    - beinhaltet das Fleisch um das Druckbild
    - Einheit Didot-Punkt (0,376 mm)  
[Didot, 1784]
  - Computerschriften verwenden Pica-Punkte (0,351mm)
    - d.h. 12 pt Pica entspricht ca. 11 pt Didot

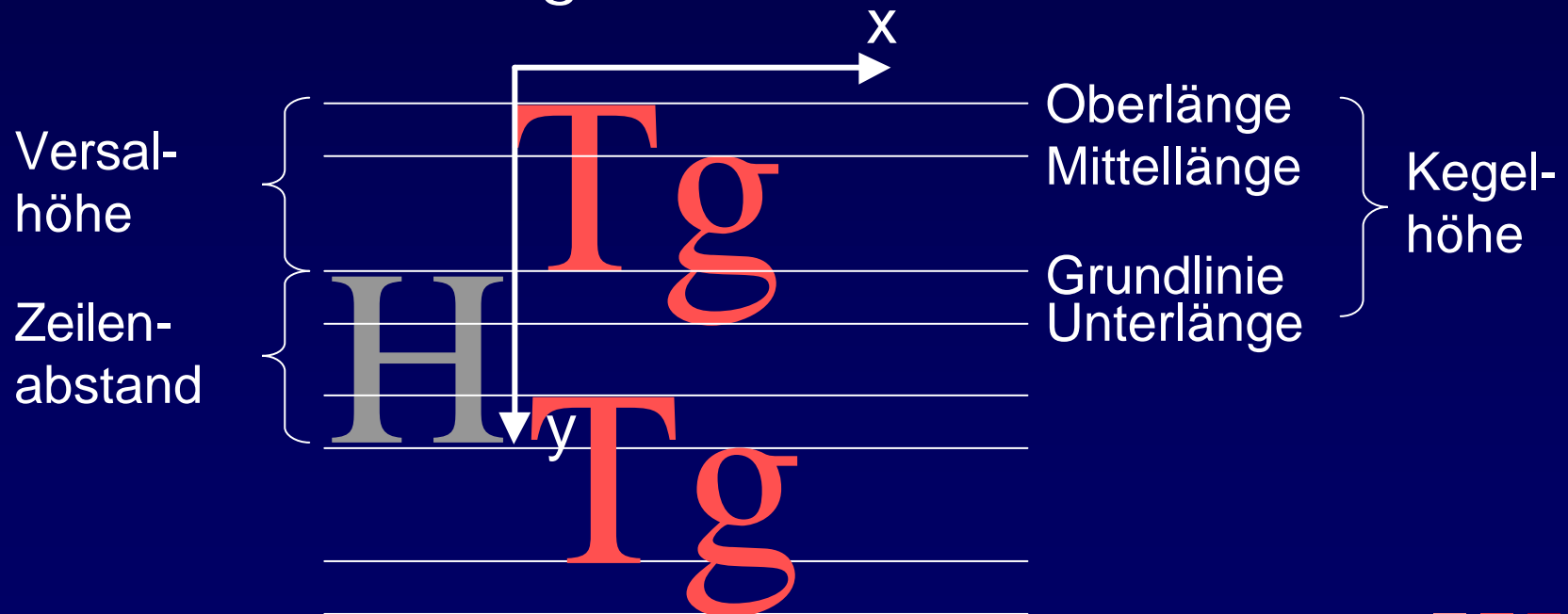


# Schriftart



## ■ Schriftmaße

- Referenzkoordinate oben links
- Zeilenabstand wird entsprechend berücksichtigt



# Schriftart



- Alternative Schriftmaße
  - ex      Höhe des kleinen x  
Länge zwischen Grund- und  
Mittellänge
  - em      Breite des großen M und damit  
des Kegels  
Oft identisch mit der  
Fontgröße in pt
  - en      Breite des großen N =  $\frac{1}{2}$  em



# Schriftart



- Wegen unterschiedlichen Fleisches sind Schriftarten gleicher Größe unterschiedlich groß im Druckbild

Dies ist 48 Punkt



# Schriftart



- Schriftattribute verändern die Präsentation der Zeichen
  - **Fett**
  - *Kursiv*
  - Schattiert
  - **Relief**
  - ...



# Schriftart



- es gibt tausende Schriftarten, z.B.
  - Times New Roman
  - Arial
  - Symbol                      Σψμβολ
  - Lucida Handwriting
  - Dingbats                      †)(■γ∞Ω)(■γ∞◆
  - ...
  
- → Typografie



# Schriftart



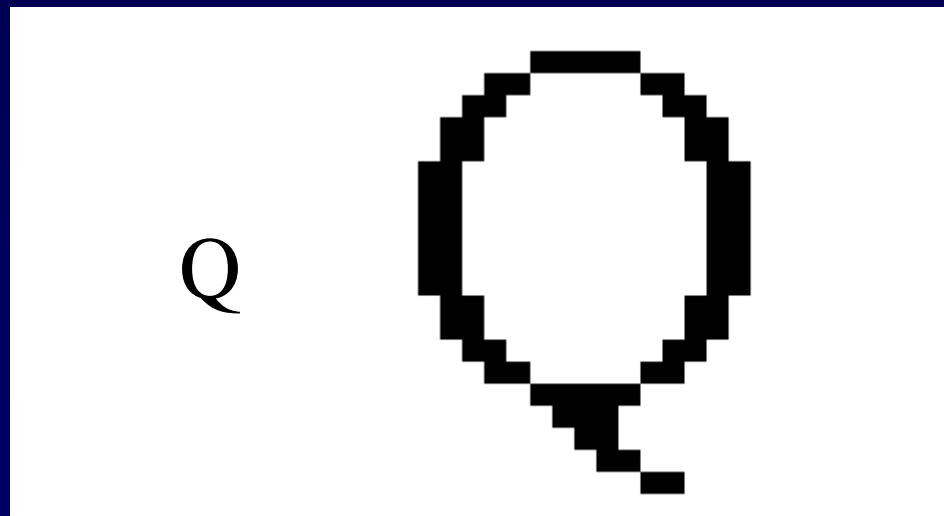
- Grafische Repräsentation der Glyphen
  - Bitmap-Font
    - Repräsentation als Bitmap
  - Outline-Font
    - Repräsentation als Vektorgrafik



# Schriftart



- Bitmap Font
  - Glyphen werden in Rasterform gespeichert
    - bei Bedarf werden sie in den Speicher geladen
    - Beispiel mit 8-facher Vergrößerung (Font= Times)



# Schriftart



- Eigenschaften von Bitmap Fonts
  - Auflösungsabhängig
  - schlecht skalierbar, oder
  - pro Fontgröße müssen die Bitmaps gespeichert sein
  - Bitmap Fonts brauchen bei zunehmender Größe viel Speicher
  - Schriftattribute müssen separat gespeichert werden
  - Betriebssysteme haben meist ihre speziellen zugeschnittenen Bitmap-Fonts



# Schriftart



- Outline-Font
  - Plattformübergreifende Formate
  - standardisierte Fontbeschreibung garantiert gleiches Aussehen auf unterschiedlichen Plattformen
  - Beispiele
    - Adobe Type 1 (Postscript Fonts)
      - Vorwiegend auf Macintosh
    - Apple TrueType
      - Vorwiegend auf Windows, aber auch Mac



# Schriftart



- Adobe Type 1
  - Glyphen werden in Untermenge von Postscript beschrieben
  - Grundlage sind kubische Bézierkurven
  - Der Adobe Type Manager (ATM) führt quasi als Treiber das Rendering durch
- TrueType
  - Grundlage sind quadratische Kurven
    - Ein Punktezug wird durch Kurven verbunden
  - Das Rendering ist ins Betriebssystem eingebaut

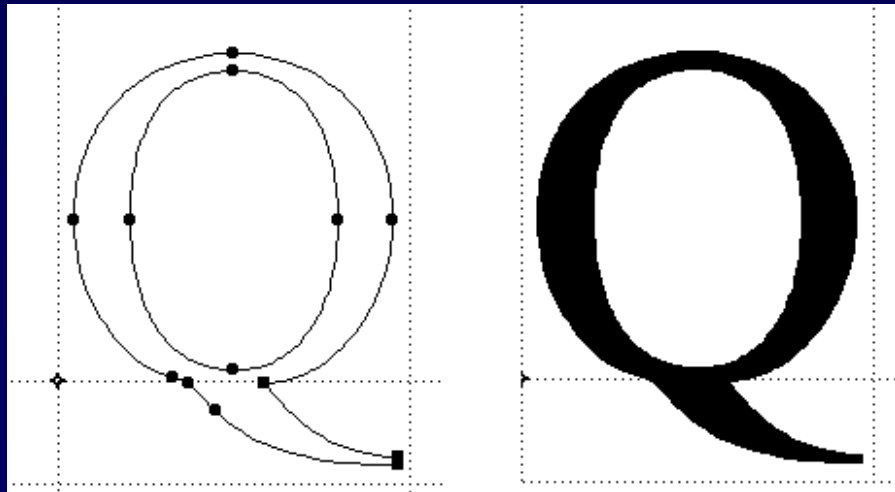


# Schriftart



- TrueType

- Umriss der Zeichen werden als quadratischer Kurvenzug angegeben
- zur Darstellung wird der Kurvenzug ausgefüllt
- „Procedural Instructions“ zur Verbesserung bei niedriger Auflösung



# Schriftart



- Interpolation & Approximation mit Splines

- stückweise linear:

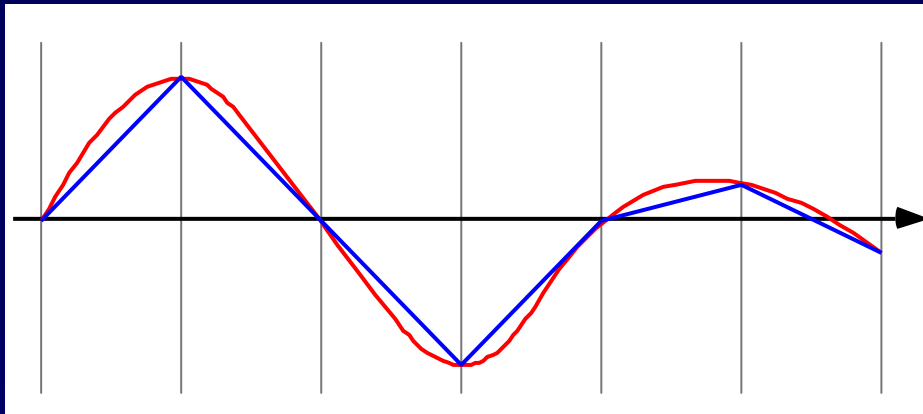
$$f_i(x) = a_i x + b_i$$

- an den Stützpunkten stetig:

$$f_i(x) = f_{i+1}(x)$$

- stückweise kubisch:

$$f_i(x) = a_i x^3 + b_i x^2 + c_i x + d_i$$



# Schriftart



- Interpolation & Approximation mit Splines
  - Berechnung:
    - stetig:
$$f_i(x_k) = S_k,$$
$$f_i(x_{k+1}) = S_{k+1}$$

→ 2n Gleichungen
    - 'glatt', Ableitungen gleich:
$$f'_i(x) = f'_{i+1}(x)$$

→ 2(n-1) Gleichungen
  - Gleichungssystem mit 4n Unbekannten:

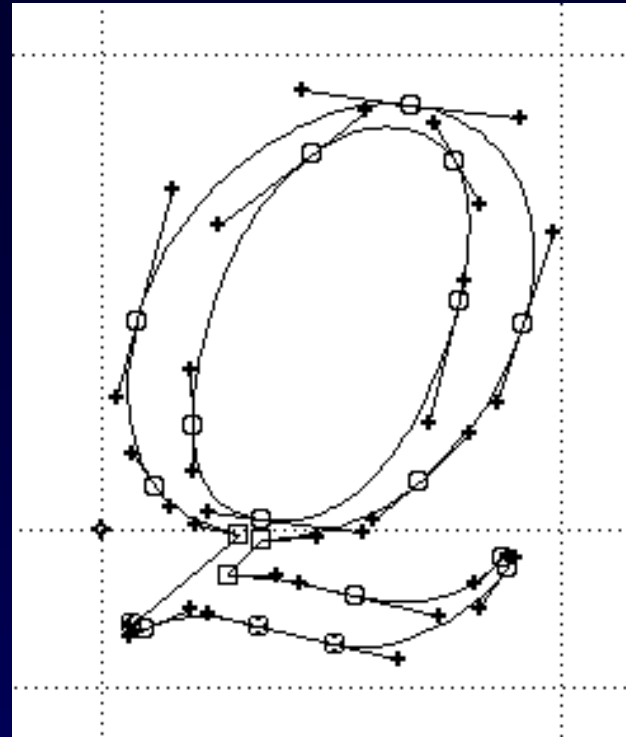
→ 2n + 2n - 2 Gleichungen
  - je nach Randbedingungen verschiedene Approximationseigenschaften



# Schriftart



- Adobe Type 1
  - Fontparameter
  - Zeichenparameter



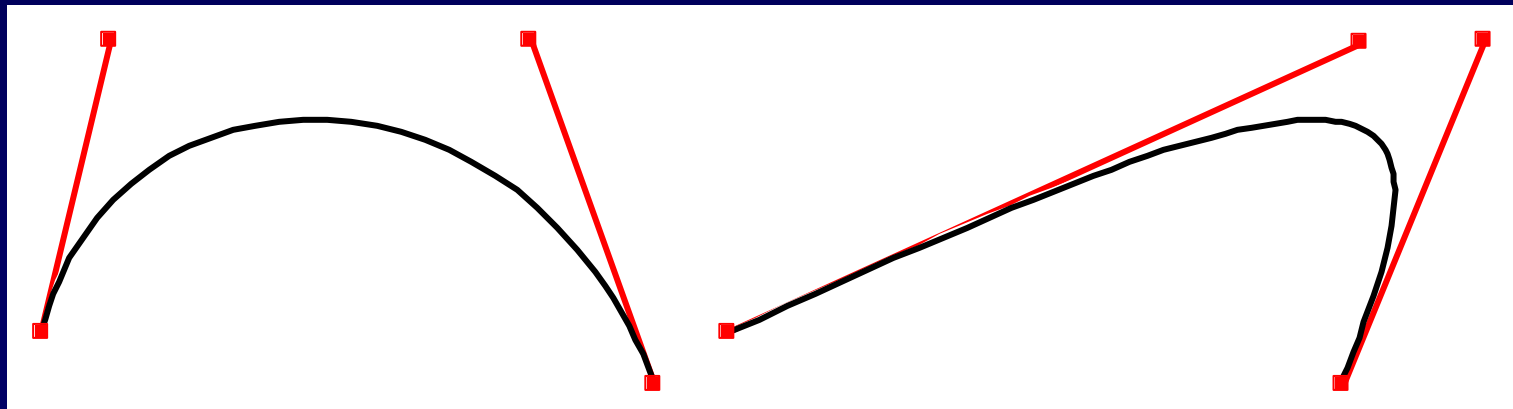
- Bézier-Kurven zur Beschreibung des Umrisses
- Deklarative Hints zur Verbesserung bei niedriger Auflösung



# Schriftart



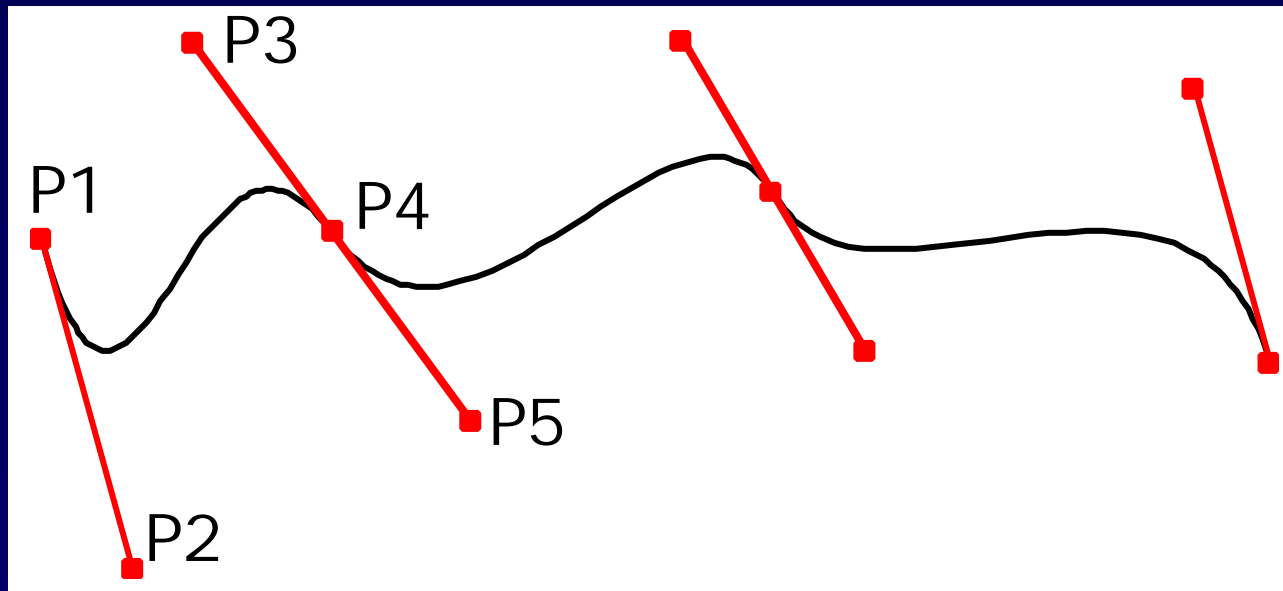
- Bézier-Kurven
  - Beschreibung der Kurve durch vier Punkte
    - Anfangspunkt und Endpunkt
    - 2 Kontrollpunkte legen Tangentenrichtung fest und „ziehen“ je nach Vektorlänge die Kurve an



# Schriftart



- zusammengesetzte Bézier-Kurven
  - Kontrollpunkte so legen, dass die Kurve glatt wird
    - 3 fortlaufende Kontrollpunkte  $P_3, P_4, P_5$  auf einer Geraden (1. Ableitung gleich)
    - gleicher Abstand zwischen  $P_3, P_4$  sowie  $P_4, P_5$



# Schriftart



- Berechnung von Beziér-Kurven

- Für 4 Punkte  $P_1 \dots P_4$  :

$$Q(t) = (1-t)^3 P_1 + 3t (1-t)^2 P_2 + 3t^2 (1-t) P_3 + t^3 P_4$$

- Durch frei gewählte Unterteilung von  $[0,1]$  für  $t$  erhält man beliebig viele Kurvenpunkte, die durch Strecken verbunden werden



# Schriftart



- OpenType
  - Dateiformat, um Type 1 und TrueType Fonts zu speichern
  - Entwicklung von Adobe und Microsoft, 1997/98



# Schriftart



- Multiple Master Fonts
  - Basiert auf Adobe Type 1
  - Eine Schriftart-Familie wird durch Parameter beschrieben
  - Designachsen
    - Gewicht ultra-light – ultra-bold
    - Breite condensed – extended
    - Optische Größe Proportionen der Glyphen, Kerning, Ligaturen, ...
    - Serifstil sans serif – serif
  - Font-Instanzen können aus einem Parametersatz generiert werden



# Schriftart



- Multiple Master Fonts
  - Nachteil
    - Multiple Master Font benötigt mehr Speicher als einzelner Font
  - Vorteile
    - Lokal nicht vorhandener Font kann leichter substituiert werden
    - Copyrighted Fonts müssen nicht im Dokument mitgespeichert werden
  - Beispiel
    - Adobe Portable Document Format, PDF
    - Acrobat Reader hat je einen generischen Serif und Sansserif Master Font



# Fortgeschrittene Textrepräsentationen



- Marked-up Text
  - Text enthält Inhalt und Form
    - z.B. troff .ce  
dies ist zentrierter Text
    - T<sub>E</sub>X, HTML, SGML
- Strukturierter Text
  - Text enthält Strukturinformationen
    - z.B. über Kapitelstruktur
    - XML, RTF



# Fortgeschrittene Textrepräsentationen



- Hypertext
  - Text ist nicht-linear
  - er enthält Knoten (den Text selbst) und Verbindungen (Links) zwischen Knoten
- Mischformen sind häufig anzutreffen
- Dazu später mehr ...

