- Die Ursprünge der TrueType-Schriften liegen in den Bemühungen von Apple und Microsoft, unabhängig von der Schrifttechnologie von Adobe zu werden.
- Dies war insbesondere für die Rasterung von Schriften am Bildschirm relevant.
- Die ursprüngliche Fassung geht auf den finnischen Entwickler Sampo Kaasila zurück, der für Apple arbeitete.
- Apple integrierte TrueType-Schriften zuerst 1991 bei MacOS 6; Microsoft folgte 1992 bei Windows 3.1.
- Beide arbeiteten unabhängig voneinander weiter, was zu zahlreichen Inkompatibilitäten geführt hat.

- Das Dateiformat ist vollumfänglich binär, aber vollständig dokumentiert.
- Dokumentation gibt es von Apple und von Microsoft, die nicht deckungsgleich sind:
  - `http://developer.apple.com/fonts/TTRefMan/index.html`
  - `http://www.microsoft.com/typography/otspec/default.htm`

- Es gibt von Just van Rossum ein frei verfügbares Werkzeug, mit dessen Hilfe TTF-Dateien in ein XML-Format und wieder zurück überführt werden können:
  `http://sourceforge.net/projects/fonttools/`
- Unter Debian steht das Werkzeug über das Paket *fonttools* zur Verfügung.
- Dies wird im folgenden verwendet werden, um einzelne Teile einer TrueType-Datei zu betrachten und zu analysieren.

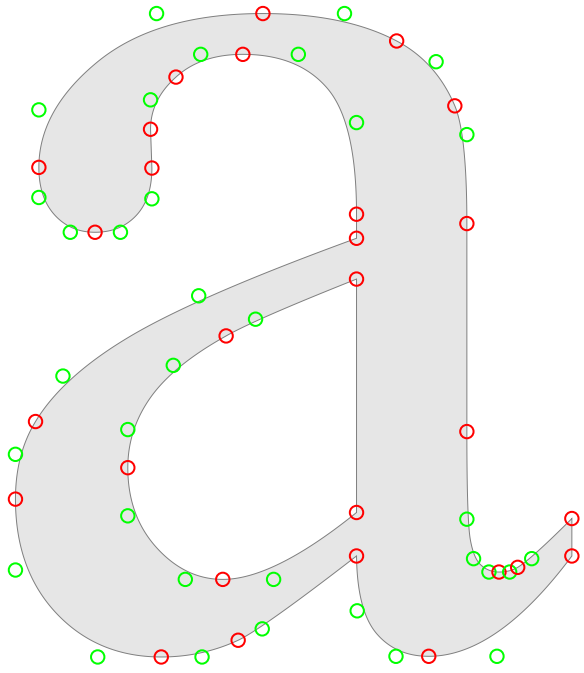
```
hochwanner$ ttx -l TimesNewRoman.ttf
Listing table info for "TimesNewRoman.ttf":
    tag      checksum   length   offset
    ----  ----------  -------  -------
    LTSH  0x814d6df0       663      316
    OS/2  0xd70a8dc7        86      980
    PCLT  0x59d381e3        54     1068
    VDMX  0x4e236882      4500     1124
    cmap  0x80a4ee32      1750     5624
    cvt   0xf3deda81      1990     7376
    fpgm  0x3775a72f      1395     9368
    gasp  0x00180009        16    10764
    glyf  0x784bf0ce    144434    10780
    hdmx  0x756f659c     15944   155216
    head  0xc17591b0        54   171160
    hhea  0x0f0308af        36   171216
    hmtx  0xb6f7a4a5      2636   171252
    kern  0xa677acd1      5220   173888
    loca  0x03143262      2640   179108
    maxp  0x08fa069b        32   181748
    name  0xf80cc1bf      4881   181780
    post  0x9141e4c3      4898   186664
    prep  0xc6afb762      3874   191564

hochwanner$
```

- Das TrueType-Dateiformat sieht auf der obersten Ebene ein Verzeichnis einzelner Tabellen vor. Die Tabellennamen haben dabei bis zu vier Zeichen und belegen einen zusammenhängenden Abschnitt der Datei.
- Die wichtigsten Tabellen sind *head* (globale Definitionen), *cmap* (Abbildung zwischen Zeichensätzen und den im Schriftschnitt enthaltenen Zeichen), *glyf* (Kurvendefinitionen der einzelnen Zeichen zusammen mit Instruktionen zur ihrer Rasterung), *hhea*, *OS/2* und *hmtx* (globale und individuelle Metriken für den horizontalen Satz) und *kern* (Kerning-Tabellen).

TimesNewRoman.ttx

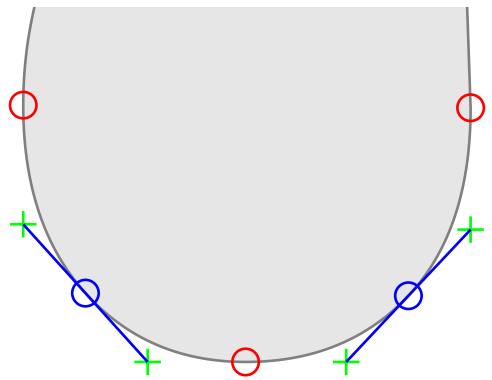
```
<TTGlyph name="a" xMin="73" yMin="-19" xMax="905" yMax="943">
  <contour>
    <pt x="583" y="132" on="1"/>
    <pt x="442" y="23" on="0"/>
    /* ... */
    <pt x="584" y="50" on="0"/>
  </contour>
  <contour>
    <pt x="583" y="197" on="1"/>
    <pt x="583" y="546" on="1"/>
    <pt x="432" y="486" on="0"/>
    <pt x="388" y="461" on="1"/>
    <pt x="309" y="417" on="0"/>
    <pt x="241" y="321" on="0"/>
    <pt x="241" y="264" on="1"/>
    <pt x="241" y="192" on="0"/>
    <pt x="327" y="97" on="0"/>
    <pt x="383" y="97" on="1"/>
    <pt x="459" y="97" on="0"/>
  </contour>
  <instructions><assembly>
      /* ... instructions for rasterization ... */
    </assembly></instructions>
</TTGlyph>
```

Kurvendefinition bei TrueType



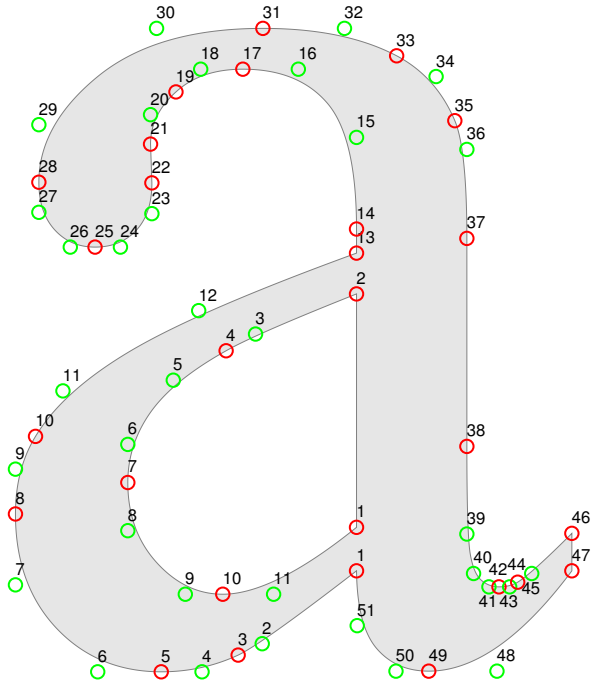
Buchstabe „a“ bei *TimesNewRoman*

- Die Kurvendefinition besteht aus beliebig vielen einzelnen jeweils geschlossenen Kurven (*contour*).
- Jede Kurvenspezifikation besteht aus beliebig vielen Punkten, die entweder auf der Kurve liegen ($on = 1$, hier rot) oder außerhalb ($on = 0$, hier grün).
- Daraus ergibt sich dann eine Folge quadratischer Bézier-Kurven und gerader Linien.

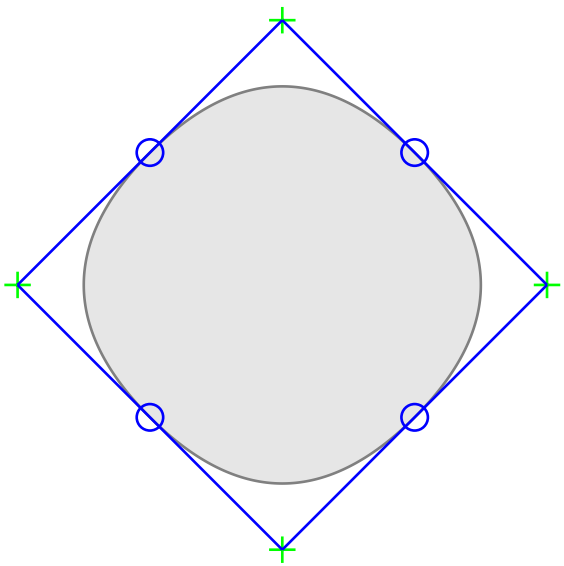


- Wenn mehrere Punkte außerhalb der zu zeichnenden Kurve aufeinanderfolgen ($on = 0$, in der Zeichnung grün), dann wird jeweils implizit genau dazwischen ein Punkt auf der Kurve gewählt (in der Zeichnung blau).
- Die jeweiligen Verbindungslinien (ebenfalls blau) zwischen zwei aufeinanderfolgenden Außenpunkten sind dann tangential zu der zu zeichnenden Kurve.

Kurvendefinition bei TrueType

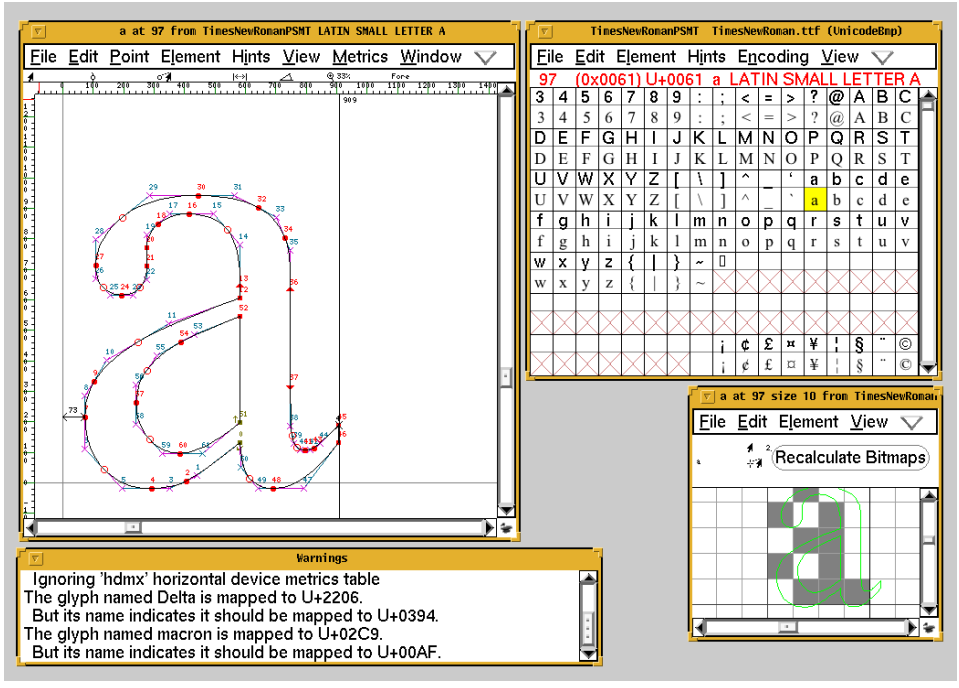


- Die einzelnen Punkte sind so aneinanderzureihen, dass die zu füllende Fläche entsprechend der gewählten Richtung immer rechts liegt.
- Außenlinien werden entsprechend im Uhrzeigersinn, Augen gegen den Uhrzeigersinn gezeichnet.
- Gefüllt wird anschließend entsprechend der *non-zero winding number rule*.

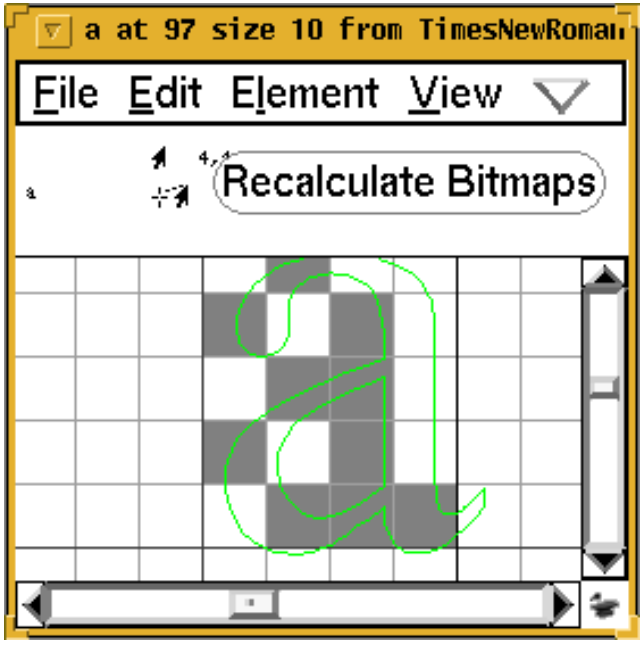


- Prinzipiell sind auch Kurvendefinitionen zulässig, bei denen alle Punkte außerhalb der Kurve liegen.
- Das Beispiel demonstriert dies an den Punkten $(200, 300)$, $(300, 400)$, $(400, 300)$ und $(300, 200)$, die ein Quadrat bilden.

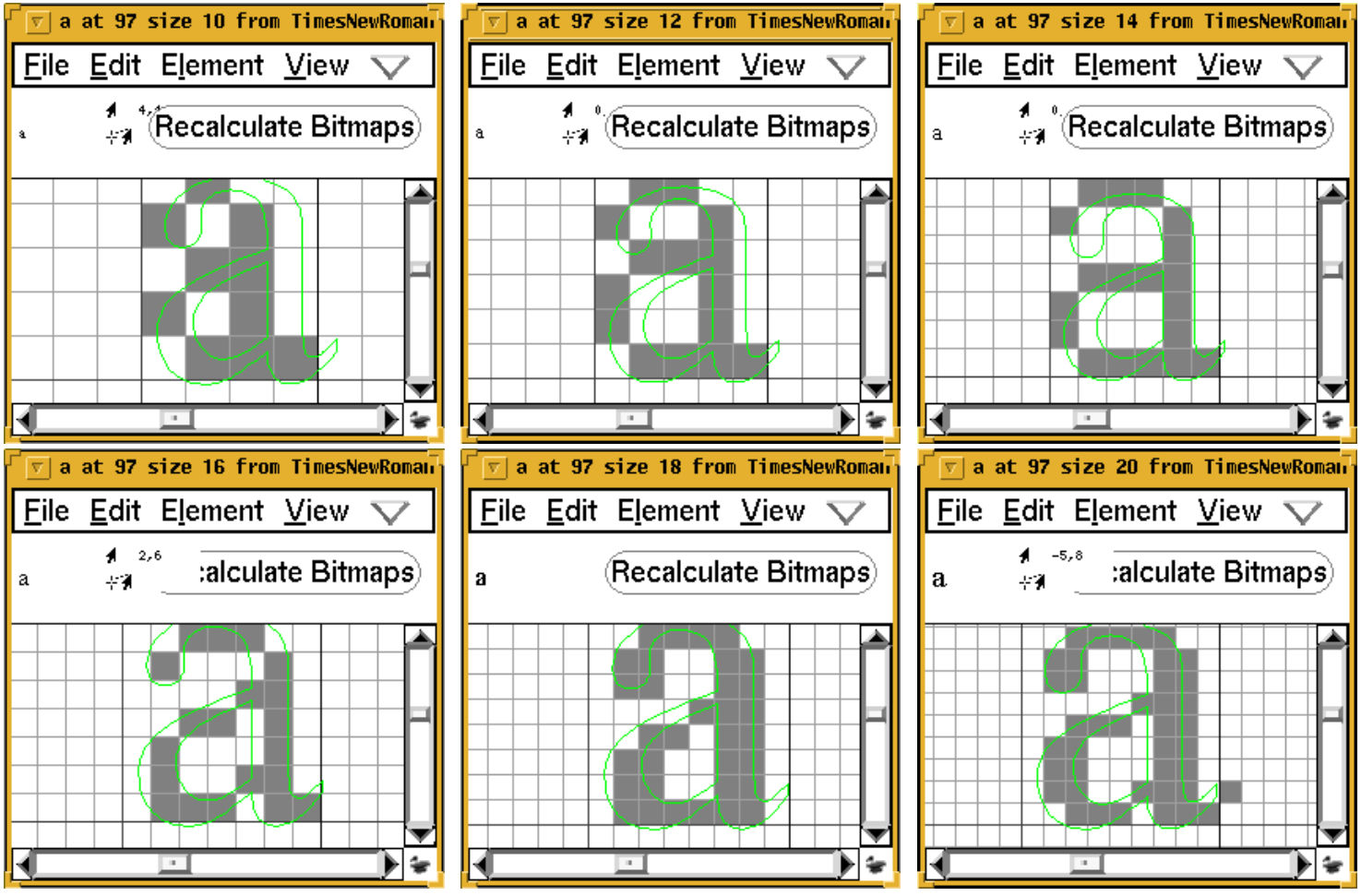
- Für die Modellierung erscheinen kubische Bézier-Kurven einfacher, da die beiden Tangenten auf sehr einfache Weise unabhängig voneinander gewählt werden können.
- Bei quadratischen Kurven sind daher mehr einzelne Bézier-Kurven notwendig.
- Die quadratischen Bézier-Kurven lassen sich jedoch sehr viel effizienter berechnen. Insbesondere vereinfacht sich die Rasterisierung.
- Auch werden unerwünschte Knicks und Scheitelpunkte bei quadratischen Kurven zuverlässig vermieden.



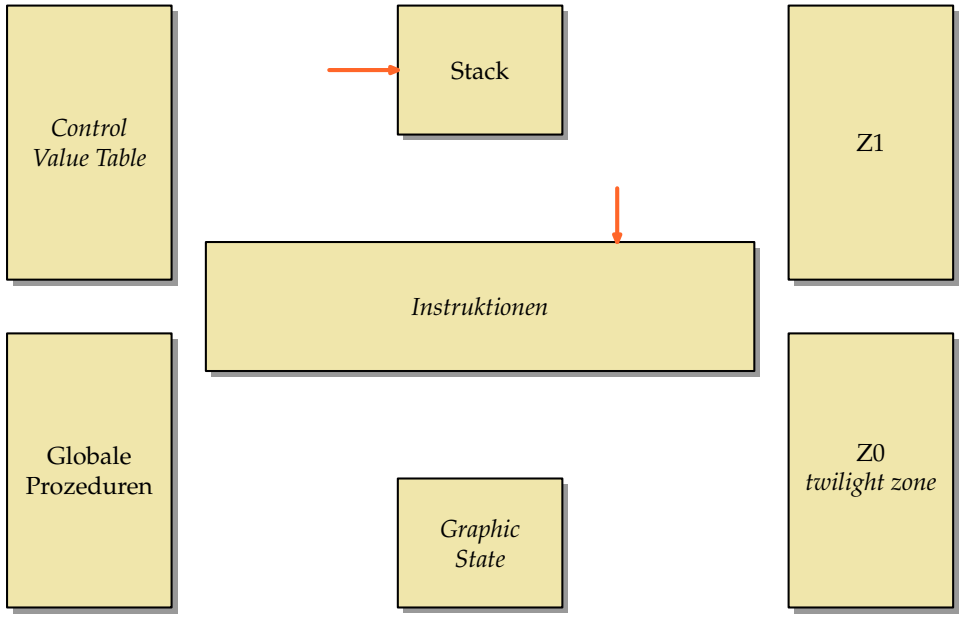
- Zur Visualisierung und insbesondere zum Testen der Rasterisierung empfiehlt sich der Einsatz des freien Software-Pakets *fontforge*.
- `http://fontforge.sourceforge.net/`
- Bei Debian steht es als Paket (*fontforge*) zur Verfügung und unter dem gleichen Namen gibt es das Paket auch bei den Macports.



- Anders als bei Type1-Schriften ist die Rasterung bei TrueType-Schriftschnitten frei programmierbar.
- Das Programm kann in Abhängigkeit von der Rasterung sämtliche Punkte frei verschieben und auch ganze Teile (wie etwa Serifen) zum Verschwinden bringen, wenn die Auflösung zu gering ist.
- In dem gezeigten Beispiel ist das „a“ trotz der Rasterung auf 4×5 Felder noch klar zu erkennen.


a at 97 size 10 from TimesNewRoman
File Edit Element View
Recalculate Bitmaps
a at 97 size 12 from TimesNewRoman
File Edit Element View
Recalculate Bitmaps
a at 97 size 14 from TimesNewRoman
File Edit Element View
Recalculate Bitmaps
a at 97 size 16 from TimesNewRoman
File Edit Element View
Recalculate Bitmaps
a at 97 size 18 from TimesNewRoman
File Edit Element View
Recalculate Bitmaps
a at 97 size 20 from TimesNewRoman
File Edit Element View
Recalculate Bitmaps

- Zu den TrueType-Formatbeschreibungen gehört eine virtuelle Maschine, die frei programmiert werden kann, um die Kontrollpunkte der Kurve an eine Rasterung anzupassen.
- Jedem einzelnen Zeichen kann eine entsprechende Prozedur beigefügt werden. Darüber hinaus werden globale Prozeduren unterstützt, die von den für ein einzelnes Zeichen zuständigen Prozeduren aufgerufen werden können.
- Für die globalen Prozeduren gibt es die Tabellen *fpgm* und *prep*. In *cvt* können globale Variablen zusammen mit ihren Initialwerten spezifiziert werden.
- Das TrueType-Format enthält nur den Code der virtuellen Maschine, der in dieser Form nur schwer lesbar ist.



- Die virtuelle Maschine führt die in einem separaten Speicher abgelegten Instruktionen aus.
- Zugänglich ist ein Stack für Berechnungen und Parameterübergabe, globale, bereits vorinitialisierte Variablen (CVT), der aktuelle Grafikzustand und die in Zonen Z1 und Z0 organisierten Kontrollpunkte der Kurven.

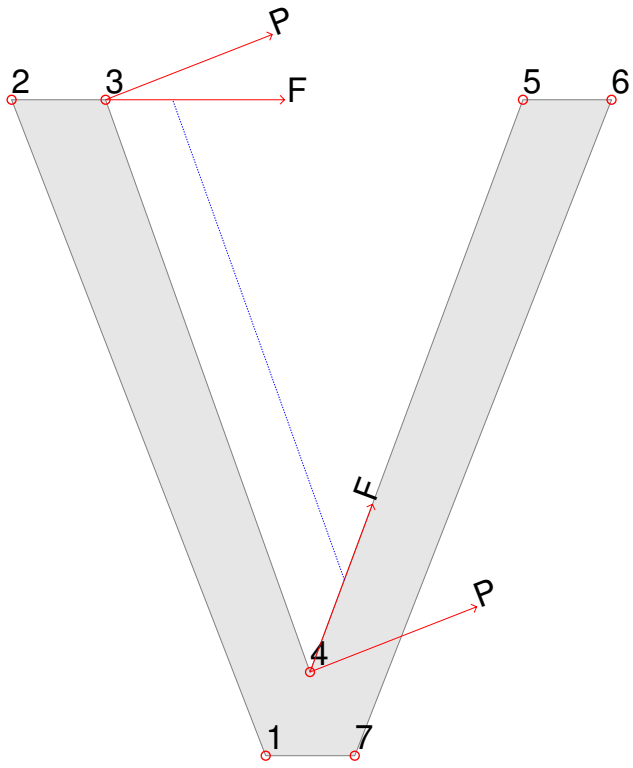
Folgende Datentypen werden unterstützt:

- *long* und *ulong*: ganze Zahlen, jeweils 32 Bit (für die Indizierung, insbesondere von Punkten, und für Boolean-Werte)
- *f26.6*: Zahl in Fixpunkt-Darstellung mit 26 binären Stellen vor dem Komma und 6 binären Stellen nach dem Komma (wird zur Repräsentierung der Koordinaten und von Distanzen verwendet)
- *f2.14*: Fixpunkt-Darstellung für Koordinaten normierter Richtungsvektoren (für Freiheits- und Projektionsvektor)

Der Stack besteht nur aus 32-Bit-Wörtern. Arithmetische Operationen gibt es nur für Werte des Typs *f26.6*.

- Alle Kontrollpunkte der einzelnen Kurven sind entsprechend der vorgegebenen Reihenfolge durchnummeriert (beginnend ab 1) und werden über eine Zone und einen Index ausgewählt.
- Für jeden Punkt gibt es die Koordinaten und die Information, ob dieser auf oder außerhalb der Kurve liegt.
- In der Zone *Z1* sind alle vorgegebenen Punkte zu finden.
- Die Zone *Z0* (auch *twilight zone* genannt) ist zu Beginn leer, kann aber frei verwendet werden.

- Ziel der Instruktionen ist es, die Punkte in *Z1* an das vorgegebene Raster anzupassen.
- Alle Punkte können individuell verschoben werden. Es ist auch möglich, die Eigenschaft zu verändern, ob ein Kontrollpunkt auf der Kurve oder außerhalb davon liegt.
- Das Verschieben der Punkte unterliegt dem Rastergitter und dem aktuellen graphischen Zustand.
- Wenn einige Punkte verschoben worden sind, können die dazwischenliegenden Punkte in dazu passender Weise nachträglich interpoliert werden.
- Wenn die Instruktionen abgearbeitet sind, kommt der klassische Rasterisierungs-Algorithmus zum Zuge, optional mit Erweiterungen, die einige Sonderfälle besser behandeln oder Anti-Aliasing unterstützen.



Nach einer Grafik von H. Schwarz
aus P. Karow: *Digitale Schriften*

- Zum graphischen Zustand gehört der Freiheits- und der Projektionsvektor.
- Distanzen werden immer nur relativ zum Projektionsvektor gemessen oder interpretiert. Die Distanzen haben daher auch ein Vorzeichen, das wahlweise berücksichtigt wird oder nicht.
- Punkteverschiebungen erfolgen immer nur in Richtung des Freiheitsvektors.
- Im Beispiel wird der Projektionsvektor orthogonal zu der Strecke $\overline{P_3P_4}$ gewählt, dann eine feste Distanz gewählt, dann P_3 entlang dem Freiheitsvektor $\overline{P_2P_3}$ verschoben, danach P_4 entlang von $\overline{P_4P_5}$.

Wenn Punkte entlang dem Freiheitsvektor verschoben werden, dann kann die tatsächlich gewählte Verschiebungsdistanz von den Rundungsmodi und dem Raster abhängig gemacht werden.
Konkret lassen sich bei den Rundungsmodi, die zum graphischen Zustand gehören, drei Parameter einstellen:

- Periode: bei 1 werden die Gitterpunkte angesprungen, bei 0.5 die Halbgitterpunkte, bei 2 nur jeder zweiter Gitterpunkt.
- Phase: bei 0 werden genau die Gitterpunkte genommen, bei größeren Werten werden die angesprungenen Punkte entsprechend zum Gitter verschoben.
- Schwelle: regelt, wohin gerundet wird: bei 1 immer zum kleineren Wert, bei 0.5 zum näher gelegenen Wert.