

```
%META:TOPICINFO{author="RainerSchoepf" date="1255366783" format="1.1" version="1.4"}%
%META:TOPICPARENT{name="Schriften"}%
```

METAFONT

 Was ist ein Meta-Font? Was sind tfm-, pl-, gf-, pk-, pxl-Dateien? Was sind virtuelle Fonts, was vpl-, vf-Dateien? Was ist METAPOST? Was ist MFT? Beim Übersetzen einer mf-Datei erhalte ich immer eine gf-Datei mit Endung „.2602gf“ (bzw. „.260“ bei Filesystemen, die nur drei Zeichen als Datei-Extension zulassen). Wieso? Wozu benötige ich für jeden Druckertyp eine andere Anpassung in Form einer „mode_def“? Wo finde ich für den Drucker XYZ eine solche Anpassung? Kann ich aus einer vorhandenen tfm-Datei die mf-Source erzeugen? Gibt es dafür ein Programm? Wie kann ich nur die tfm-Dateien einer Schrift erzeugen?

METAFONT ist wie TeX von Donald E. Knuth entwickelt worden. Es erlaubt die Beschreibung einer Schrift oder einer Schriftfamilie durch ein Programm, den sogenannten Meta-Font. Aus diesem Meta-Font kann METAFONT nach Wahl verschiedener Parameter Schriften generieren, die TeX verwenden kann.

Um mit TeX zu arbeiten, ist METAFONT nicht zwingend erforderlich. Es ist auch möglich, beispielsweise mit PostScript-Schriften zu arbeiten. TeX benötigt von einer Schrift nur die Maße der einzelnen Zeichen und weitere schriftabhängige Parameter, wie beispielsweise die Breite eines Wortzwischenraumes. Diese Informationen entnimmt TeX einer TeX-Font-Metric-(tfm)-Datei.

METAPOST ist ein aus METAFONT abgeleitetes Programm zum Erstellen von PostScript-Zeichnungen.

Was ist ein Meta-Font?

Um die einzelnen Zeichen einer Schrift, eines Font, auf Papier zu bringen, muß man wissen, an welchen Stellen das Papier mit Druckerfarbe geschwärzt werden soll und an welchen nicht. Diese Information kann auf verschiedene Arten repräsentiert werden: Die einfachste Form ist die ‚Bitmap‘ bzw. ‚Pixelmap‘, d.h. jedes Zeichen wird gerastert und für jeden einzelnen Punkt im Raster (= Pixel) wird abgespeichert, ob diese Stelle geschwärzt werden soll. Eine Bitmap-Repräsentation hat jedoch den Nachteil, daß man diese nicht ohne weiteres vergrößern, verkleinern oder die Rasterung an eine andere Auflösung eines Ausgabegeräts anpassen kann, da man in den meisten Fällen unerwünschte Seiteneffekte bekommt (Stufen, Moiré-Effekte, dünnere Linien verschwinden bei Verkleinerung, als regelmäßige oder unregelmäßige Muster sichtbare Rasterung u.ä.).

Man verwendet zur Beschreibung der Ausmaße und der Form eines Zeichens daher besser eine rasterungs-, also auflösungsunabhängige Repräsentation. Dazu gibt es nun mehrere Verfahren:

In METAFONT legt man Linienzüge durch Punkte und durch diese Punkte festgelegte Spline-Kurven fest und zieht diese mit einem vorher definierten ‚Pinsel‘ nach. Durch die Form und die Lage des verwendeten Pinsels können mit denselben Kurven sehr unterschiedliche Zeichenformen erzeugt werden. Nachdem das Zeichen aus einzelnen Kurvenzügen definiert und mit unterschiedlichen Pinseln gezeichnet wurde, rastert METAFONT die dadurch eingefärbten Flächenteile und erzeugt daraus einen Font in Form einer Bitmap. Da METAFONT zur Programmierung einer solchen Font-Beschreibung eine Programmiersprache definiert, kann ein Font-Designer einer Font-Beschreibung beliebig viele Parameter mitgeben, die die Form und Ausmaße der einzelnen Zeichen der erzeugten Schrift in

unterschiedlichen Richtungen verändern. Wegen dieser Möglichkeit, aus einer einzigen Beschreibung durch Wahl verschiedener Parameter viele verschiedene Schriften erzeugen zu können, nennt man diese Font-Beschreibungen auch ‚Meta-Font‘. Man kann also durch Wahl der in einem Meta-Font definierten Parameter unterschiedliche Fonts in verschiedenen Größen erzeugen. Durch Angabe der gewünschten Auflösung und weiterer Parameter des Ausgabegeräts kann wiederum für einen solchen Font eine optimal an das jeweilige Ausgabegerät angepaßte Bitmap-Wiedergabe dieses Fonts erzeugt werden.

Neben dieser von METAFONT verwendeten Beschreibung eines Fonts in Form von mit Pinseln gezeichneten Spline-Kurven kann man die Form eines Zeichens auch durch die Angabe des Umrisses der zu schwärzenden Fläche beschreiben. Auch hier werden wiederum Spline-Kurven zur Beschreibung der Umrisse eines Zeichens verwendet. Diese Beschreibung in Form eines sogenannten ‚Outline‘ wird beispielsweise in PostScript-Schriften verwendet.

Was sind tfm-, pl-, gf-, pk-, pxl-Dateien?

Neben dem Aussehen der einzelnen Zeichen in einem Font werden noch weitere Informationen über jedes einzelne Zeichen und über schrift-globale Parameter benötigt: Von jedem Zeichen muß TeX die Ausmaße in Form von Breite, Höhe und Tiefe in Bezug zur Grundlinie wissen. Zu bildende Ligaturen (z.B. f+f $\Rightarrow$ ff), das Kerning zwischen zwei Zeichen (z.B. Unterschneidung bei ‚Ta‘) und beispielsweise der zu verwendende Wortabstand sind weitere Informationen, die über einen Font abgelegt sein müssen, da sie von TeX benötigt werden.

Diese Informationen liest TeX aus der ‚TeX Font Metric‘ oder kurz tfm-Datei. Die Programme ‚pltotf‘ und ‚tftopl‘ wandeln die Informationen in einer tfm-Datei in eine lesbare Form, die ‚Property List‘ oder kurz pl-Datei um.

In den gf- (Generic Font), pxl- (Pixel) und pk- (Packed Pixel) Dateien sind die Bitmap-Repräsentationen jedes Zeichens einer Schrift abgelegt. Da pk- und pxl-Dateien genau dieselben Informationen enthalten, die pxl-Dateien jedoch mehr Platz benötigen, werden diese heutzutage nicht mehr verwendet und unterstützt. Mit den Programmen ‚gftopk‘ und ‚pktogf‘ können die Bitmap-Repräsentationen ineinander umgewandelt werden. Da gf-Dateien in der Lage sind, in einer Bitmap neben den einzelnen Pixeln noch weitere Informationen zu enthalten (z.B. ‚special‘-Anweisungen), können bei dieser Umwandlung Informationen verloren gehen. Da jedoch nur sehr wenige Programme und Gerätetreiber diese zusätzlichen Informationen verwenden und die pk-Darstellung viel kompakter ist, sind gf-Dateien normalerweise nur für METAFONT wichtig.

METAFONT erzeugt aus einer mf-Source eine tfm-Datei mit der auflösungs *un* abhängigen ‚TeX Font Metric‘ und eine auflösungsabhängige gf-Datei. Aus der gf-Datei wird üblicherweise mit ‚gftopk‘ eine pk-Datei erzeugt.

TeX verwendet nur die Informationen in der tfm-Datei, die Bitmap in der gf- bzw. der pk-Datei wird von TeX *nicht* gelesen. Ein Gerätetreiber, der eine dvi-Datei druckt oder darstellt, verwendet die pk-Dateien und in einigen Fällen auch die tfm-Datei.

Was sind virtuelle Fonts, was vpl-, vf-Dateien?

Ein virtueller Font ist, wie der Name schon sagt, eine Schrift, die keine real (in Form einer Bitmap) existierenden Zeichen enthält. Stattdessen wird in einem virtuellen Font auf andere Schriften und die

dort enthaltene Zeichen verwiesen. Durch diesen ‚Trick‘ kann man beispielsweise die Kodierung der Zeichen in einer Schrift umdefinieren oder eine Schrift aus Zeichen mehrerer Schriften zusammensetzen, um so z.B. einzelne Zeichen durch andere zu ersetzen. Darüber hinaus kann man in virtuellen Fonts auch einzelne Zeichen aus mehreren Zeichen zusammensetzen, indem man beispielsweise fertig akzentuierte Zeichen aus dem entsprechenden Akzent- und Basiszeichen bildet. Eine weitere Anwendung für virtuelle Fonts ist die Verwendung einer Ersatzschrift für eine nicht verfügbare Schrift, die dieser möglichst nahekommt und dieselben TeX-Font-Metric-Informationen besitzt.

Eine vf-Datei enthält die Definition eines solchen virtuellen Fonts in binärer, sehr kompakter Form. Dieser vf-Datei muß eine tfm-Datei zugeordnet sein, in der TeX die ‚TeX Font Metric‘ für diese Schrift finden kann. Eine vpl-Datei ist eine für den Menschen lesbare Darstellung der vf- und tfm-Datei in Form von ‚Property Lists‘. Mit dem Programm ‚vptovf‘ kann aus einer vpl-Datei eine vf- und die zugehörige tfm-Datei erzeugt werden, mit ‚vftovp‘ bekommt man aus diesen beiden Dateien wieder die vpl-Datei.

TeX ‚weiß‘ nichts von virtuellen Fonts. TeX liest keine vf-Dateien, sondern benötigt nur die zugehörige tfm-Datei, aus der nicht ersichtlich ist, ob dies ein ‚realer‘ oder ein virtueller Font ist. vf-Dateien werden nur von den Gerätetreibern verwendet. Für Treiber, die (noch) keine virtuellen Fonts unterstützen, kann das Programm ‚dvi-copy‘ verwendet werden, um eine dvi-Datei in eine dvi-Datei umzuwandeln, in der alle virtuellen Fonts ‚aufgelöst‘ worden sind, also nur noch ‚reale‘ Fonts verwendet werden. Achtung: Die meisten verfügbaren virtuellen Fonts sind unabhängig vom Ausgabegerät, sie können jedoch auch abhängig vom Ausgabegerät sein!

Was ist MFT?

‚mft‘ [id="10"> ist ein Programm, das aus METAFONT-Quellcode eine TeX-Datei erzeugt, mit der dieser Quellcode schön formatiert erscheint. Damit es wirklich schön aussieht, sind zusätzliche Kommentare im Quellcode notwendig. Nur sehr wenige Autoren von METAFONT verwenden ‚mft‘.](#)

Beim Übersetzen einer mf-Datei erhalte ich immer eine gf-Datei mit Endung ‚.2602gf‘ (bzw. ‚.260‘ bei Filesystemen, die nur drei Zeichen als Datei-Extension zulassen). Wieso?

Zur Erzeugung einer Schrift für ein bestimmtes Ausgabegerät benötigt METAFONT eine Beschreibung in Form einer ‚Mode Definition‘ (kurz: ‚mode_def‘), in der u.a. die Auflösung des Ausgabegeräts festgelegt wird. METAFONT wird meist mit

```
mf \mode:=; mag:=; input
```

aufgerufen (unter Unix Quote-Zeichen ‚ ‘ um die Argumente im METAFONT-Aufruf nicht vergessen), wobei mit der Namen einer solchen ‚mode_def‘ anzugeben ist. Wird kein ‚mode‘ angegeben oder ist unbekannt, verwendet METAFONT den Proof-Mode, der einen Korrekturabzug der Schrift erstellt. Damit die Zeichen in diesem Korrekturabzug groß genug sind, verwendet METAFONT dazu die Auflösung 2602dpi (= ‚dots per inch‘). Falls dies nicht erwünscht war, sollte man überprüfen, ob der ‚mode‘ korrekt gesetzt wurde, ob beispielsweise korrekt angegeben wurde (Schreibfehler) und ob bei der Erstellung der base-Datei (‚Dumpen‘ der Datei ‚plain.mf‘) mit ‚mode_def‘ definiert wurde.

Wozu benötige ich für jeden Druckertyp eine andere Anpassung in Form einer ,mode_def`? Wo finde ich für den Drucker XYZ eine solche Anpassung?

Jedes Ausgabegerät hat unterschiedliche Charakteristiken. Zum einen unterscheiden sie sich in der Auflösung, zum anderen beispielsweise auch in der Größe eines gezeichneten Pixels. So wird bei einem Tintenstrahldrucker im Vergleich zu einem Laserdrucker in den meisten Fällen ein einzelner Punkt etwas größer ausfallen. Sehr große Unterschiede treten prinzipbedingt beim Vergleich von sogenannten ,schwarz`- und ,weiß`-schreibenden Laserdruckern auf. Auch wird beim Schnitt zweier Kurven (z.B. in der Mitte des Zeichens ,X`) die geschwärzte Fläche etwas größer ausfallen, da die Druckerfarbe abhängig vom Druckprinzip und vom Druckwerk mehr oder weniger ineinander verlaufen kann. Damit METAFONT all diese und weitere Eigenschaften eines Ausgabegeräts bei der Erzeugung der Bitmap-Rasterung eines Zeichens in Betracht ziehen kann, werden die einzelnen Parameterwerte für ein Ausgabegerät in einer ,Mode Definition` mit der Anweisung ,mode\_def` abgelegt.

Da die Erstellung einer neuen ,Mode Definition` für ein Ausgabegerät einiges an Arbeit (Testdruck, überprüfen, anpassen, Testdruck usw.) bedeutet, wurden von Karl Berry<a name="13" id="13"></a> alle bisher für verschiedene Ausgabegeräte gefundenen ,Mode Definitions` in der Datei „modes.mf`` gesammelt. Da „modes.mf`` neben den ,Mode Definitions` noch dafür sorgt, daß man verschiedene Informationen (z.B. ,coding scheme`) über einen Font setzen kann und diese korrekt abgelegt werden, ist diese Datei jedem empfohlen. | **modes.mf:** | [fonts/modes/modes.mf](https://wiki.dante.de/doku.php?id=dantefaq:metafont#modesmf) |

(Hinweis: Seit Februar 2002 gibt es die Version 3.5 mit Voreinstellungen für neuere Drucker)

Kann ich aus einer vorhandenen tfm-Datei die mf-Source erzeugen? Gibt es dafür ein Programm?

Nein. tfm-Dateien enthalten keine Bitmap- oder ähnliche Information über das Aussehen der Zeichen in einer Schrift. Daher kann aus einem tfm-File weder eine pk-Datei noch die mf-Source restauriert werden.

Es ist jedoch möglich, aus einer pk- und der tfm-Datei einer Schrift eine mf-Source zu erstellen. Dabei ist das entstehende Resultat jedoch kein *Meta*-Font mehr, sondern ,nur` noch ein mf-Programm für genau diese Schrift. Das Ergebnis ist je nach Aufwand mehr oder weniger gut und verlangt in den meisten Fällen noch einiges an Handarbeit, so daß sich der Aufwand nur in seltenen Fällen lohnt.

Wie kann ich nur die tfm-Dateien einer Schrift erzeugen?

In der Datei „modes.mf`` ist dafür die Mode-Definition ,nullmode` gedacht, in der neben der tfm-zwar auch eine gf-Datei erzeugt wird, jedoch in einer Auflösung von 101dpi, so daß dies schneller als mit 300dpi oder 600dpi geschieht.

- Main.HerbertVoss - 22 Feb 2009

From:

<https://wiki.dante.de/> - **DanteWiki**

Permanent link:

<https://wiki.dante.de/doku.php?id=dantefaq:metafont>

Last update: **2023/02/28 10:39**

