

Schriften unter Linux

- ein wenig Hintergrundwissen

verfasst MadMax, überarbeitet von wobo

Um gut lesbare Schriften am Bildschirm darzustellen gelten ganz andere Kriterien im Vergleich zur Darstellung von Schriften auf gedrucktem Papier.

Der Unterschied liegt ganz einfach in der viel geringeren Auflösung des Monitors (ca. 75..90 dpi) im Vergleich zu einem Drucker (ca. 600dpi) begründet - während sich bei einem hochauflösenden Drucker ein einzelnes Pixel nicht sehr stark auf das Schriftbild auswirkt, entscheidet bei der Darstellung am Monitor jedes einzelne Pixel, ob eine Schrift leserlich wirkt oder nicht.

Bei Pixel-Schriften sind die Schriftzeichen Punkt für Punkt aufgebaut. Sie eignen sich sehr gut für eine Darstellung auf Medien mit geringer Auflösung, haben aber den Nachteil, daß man sie nur in ungenügender Qualität drucken kann. Reine skalierbare Schriften (wie z.B. die unter Linux weit verbreiteten Type1-Schriften) sind vektoriell aufgebaut und lassen sich beliebig vergrößern oder verkleinern. Sie eignen sich sehr gut für den hochauflösenden Druck, die Darstellung auf dem Bildschirm ist aber nicht so gut wie bei TrueType-Schriften. Warum das so ist, erfährt man im nächsten Absatz:

Schrift-Hinting

TrueType-Schriften sind ebenfalls skalierbar, dennoch eignen sie sich gleichermaßen für Bildschirmdarstellung und Druck, da sie im Idealfall zusätzliche Informationen für eine optimierte Darstellung bei geringen Auflösungen enthalten. Genauer gesagt enthält eine gut erstellte TTF-Schrift Anweisungen, welches Pixel Teil der Schrift sein soll und welches nicht, um insbesondere bei kleinen Schriften ein besseres Gesamtbild der Schrift am Bildschirm zu erhalten. Man spricht vom "Hinting" - nicht zu verwechseln mit dem "Subpixel-Hinting", dazu mehr aber erst weiter unten im Text.

Kantenglättung (Antialiasing)

Wenn eine lediglich aus schwarzen Pixeln bestehende Schrift am Bildschirm dargestellt wird, wirkt sie zwar scharf aber je nach Schrifttyp auch "krakelig", da bei Rundungen oder diagonalen Linien sehr leicht ein "Treppenstufen-Effekt" entstehen kann.

Bei der Kantenglättung versucht man solche Stufen durch die Platzierung von zusätzlichen Farben (im einfachsten Fall grau) an den Stufen zu "runden". Dadurch wirken Konturenverläufe weicher. Das Schriftbild kann durch die zusätzlichen

Pixel aber auch leicht verschwommen wirken - insbesondere bei kleinen Schriftgrößen. Erkennbar ist die herkömmliche Kantenglättung an einem grauen Saum um die Schriftzeichen herum.

Subpixel-Hinting

Das Subpixel-Hinting ist eine spezielle Form der Kantenglättung, die bei TFT-Bildschirmen Anwendung findet. Dabei macht man sich die bauartbedingten Eigenheiten der TFT's zunutze. Ein einzelnes Farb-Pixel wird beim TFT jeweils aus mehreren Subpixeln in den Grundfarben rot, grün und blau erzeugt. Die Subpixel kann man sehr gut erkennen, wenn man das TFT-Display an weißen Bereichen mit einer starken Lupe betrachtet. Zum Glätten der Schrift wird an den "harten" Stufen gezielt ein farbiges Pixel erzeugt. Da die roten und grünen Subpixel viel heller leuchten als das blaue Subpixel, entsteht dadurch der Eindruck einer ebenfalls geglätteten Schrift. Allerdings sind die Konturenverläufe feiner und damit das Schriftbild weniger verschwommen als bei der herkömmlichen Kantenglättung wie oben beschrieben, da die dunkel erscheinenden Subpixel ja kleiner sind als ein graues Pixel.

Da ein herkömmlicher Röhrenmonitor ein anderes Verfahren zur Darstellung von Farben verwendet, kann Subpixel-Hinting dort nicht wie gewünscht funktionieren.

Erkennbar ist die herkömmliche Kantenglättung an einem grauen Saum um die Schriftzeichen herum. Unter Windows findet man Subpixel-Hinting unter der Bezeichnung "ClearType" wieder.

Der Font-Renderer

Ein Font-Renderer hat die Aufgabe, skalierbare Schriften in Pixel umzuwandeln. Er beinhaltet auch die Routinen zum Schrift-Hinting, zur Kantenglättung und zum Subpixel-Hinting. Unter Linux wird zum Darstellen von Schriften in "X" der Font-Renderer von FreeType eingesetzt <http://www.freetype.org/>

Leider ist der von FreeType entwickelte Algorithmus zum Benutzen des Schrift-Hintings - der "Byte Code Interpreter" (BCI) patentrechtlich geschützt. Es existiert aber noch ein zweiter Algorithmus - der "AutoHinter" - welcher in allen Ländern legal eingesetzt werden kann. Allerdings kann dieser den Ergebnissen des BCI's nicht ganz das Wasser reichen.

Aufgrund der patentrechtlichen Situation werden die meisten Distributionen ohne den BCI zur Verfügung gestellt.

Will man aber perfekt gerenderte Schriften in Linux, muß man den Font-Renderer von FreeType mit aktiviertem BCI neu kompilieren.

Für Mandriva hat einem die "Penguin Liberation Front" (PLF) diese Arbeit bereits abgenommen und man kann das fertige RPM-Paket (*libfreetype6*, derzeit in der Version 2.1.10) von deren Servern herunterladen.

Jetzt wird's ernst - Einstellungen am System vornehmen

In Linux werden die Einstellungen zum Schrift-Rendering in dieser Datei gespeichert:
`~/.fonts.conf`

Natürlich kann man z.B. auch im KDE-Kontrollzentrum (unter "*Erscheinungsbild - Schriften*") Optionen zum Schrift-Rendering einstellen, allerdings sind dort nicht alle Optionen wie weiter beschrieben verfügbar. Deshalb beschränke ich mich im weiteren auf das Editieren der `~/.fonts.conf`

Die `~/.fonts.conf` hat folgendes Grundgerüst im XML-Format:

```
<?xml version="1.0"?>
<!DOCTYPE fontconfig SYSTEM "fonts.dtd">
<fontconfig>
...
...
</fontconfig>
```

Je nach Bedarf werden die nachstehenden Optionen einfach zwischen die Zeilen "`<fontconfig>`" und "`</fontconfig>`" hinzugefügt:

Schriften-Hinting (BCI)

```
<match target="font" >
<edit mode="assign" name="hinting" >
  <bool>OPTION</bool>
</edit>
</match>
```

Mögliche Werte für "OPTION" sind:
* true
* false

Achtung: Diese Einstellung zeigt natürlich nur Wirkung, wenn zuvor der FreeType-Renderer mit aktiviertem BCI (s.o.) installiert worden ist!

Schriften-Hinting (AutoHinter)

```
<match target="pattern" >
<edit mode="assign" name="autohint" >
  <bool>OPTION</bool>
</edit>
</match>
```

Mögliche Werte für "OPTION" sind:
* true
* false

Hinting-Intensität:

```
<match target="font" >
<edit mode="assign" name="hintstyle" >
  <const>OPTION</const>
</edit>
</match>
```

Mögliche Werte für "OPTION" sind:

- * hintnone
- * hintslight
- * hintmedium
- * hintfull

Subpixel-Anordnung (Monitortyp)

```
<match target="font" >
<edit mode="assign" name="rgba" >
  <const>OPTION</const>
</edit>
</match>
```

Mögliche Werte für "OPTION" sind:

- * rgb (gewöhnlicher TFT, Subpixelmuster ist RGB)
- * bgr (eher seltener TFT, Subpixelmuster ist BGR)
- * vrgb (gewöhnlicher TFT um 90° gedreht, Subpixelmuster ist RGB)
- * vbgr (eher seltener TFT um 90° gedreht, Subpixelmuster ist BGR)
- * none (herkömmlicher Röhrenmonitor)

Wie man an den Optionen sieht, gibt es verschiedene Bauarten von TFT's, die sich durch die Anordnung der Subpixel unterscheiden. Damit das Subpixel-Hinting richtig funktionieren kann, muß die genaue Anordnung der Subpixel bekannt sein. Dazu sieht man am Besten mit einer starken Lupe an weißen Bereichen des Bildschirms nach:

Erkennt man (von links nach rechts) z.B. ein Farbmuster nach der Reihenfolge rot, grün, blau, rot, grün, blau, etc. - hat man ein TFT-Display mit RGB-Subpixelmuster - usw. Die meisten TFT's werden ein RGB-Muster aufweisen.

Kantenglättung

```
<match target="font" >
<edit mode="assign" name="antialias" >
  <bool>OPTION</bool>
</edit>
</match>
```

Mögliche Werte für "OPTION" sind:
* true
* false

Abhängig davon, ob zuvor der Wert "rgba" auf "none" gesetzt oder aber ein bestimmtes Subpixel-Muster angegeben wurde, wird entweder herkömmliche Kantenglättung (Graustufen) oder Subpixel-Hinting angewandt.

Kantenglättung für kleine Schriften deaktivieren

Obwohl die Kantenglättung bei normalen Schriftgrößen i.d.R. die Lesbarkeit erhöht, kann sie bei kleinen Schriftgrößen durch die zusätzlichen Farbpixel genau das Gegenteil bewirken. Deshalb ist es ratsam, die Kantenglättung für kleine Schriften abzuschalten. Dies geschieht durch den folgenden Eintrag in der `~/fonts.conf`:

```
<match target="font" >
  <test compare="less" name="pixelsize"
qual="any" >
    <double>11</double>
  </test>
  <edit mode="assign" name="antialias" >
    <bool>>false</bool>
  </edit>
</match>
```

Im obigen Beispiel wird die Kantenglättung für Schriftgrößen unter 11px deaktiviert. Für mein persönliches Empfinden ist diese Einstellung ideal.

Anmerkung: Die Schriftgröße 11px ist NICHT gleichbedeutend mit Schriftgröße 11pt! Die sonst übliche Größenangabe bei Schriften "pt" ist eine Längeneinheit, "pixelsize" bzw. "px" dagegen beschreibt die tatsächlich auf dem Monitor dargestellte Schriftgröße in Pixeln!

So sollten die Schriften in selber Pixelgröße bei jeder Auflösung des Monitors gleich gerendert werden - je nach verwendeter Monitor-Auflösung kann es aber sein, daß bei gleicher Schriftgröße in "pt" die Kantenglättung bereits aktiviert, oder auch deaktiviert ist.

Beispiel 1:

Schrift-Hinting (AutoHinter), Kantenglättung durch Subpixel-Hinting, TFT-Monitor (RGB)

```
<?xml version="1.0"?>
<!DOCTYPE fontconfig SYSTEM "fonts.dtd">
<fontconfig>
<dir>~/fonts</dir>
<match target="font" >
  <edit mode="assign" name="autohint" >
    <bool>>true</bool>
  </edit>
</match>
<match target="font" >
  <edit mode="assign" name="hintstyle" >
    <const>hintfull</const>
  </edit>
</match>
<match target="font" >
  <edit mode="assign" name="antialias" >
    <bool>>true</bool>
  </edit>
</match>
<match target="font" >
  <edit mode="assign" name="rgba" >
    <const>rgb</const>
```

```
</edit>
</match>
</fontconfig>
```

Beispiel 2:

Schrift-Hinting (AutoHinter), Kantenglättung durch Graustufen, Röhren-Monitor

```
<?xml version="1.0"?>
<!DOCTYPE fontconfig SYSTEM "fonts.dtd">
<fontconfig>
<dir>~/fonts</dir>
<match target="font" >
  <edit mode="assign" name="autohint" >
    <bool>>true</bool>
  </edit>
</match>
<match target="font" >
  <edit mode="assign" name="hintstyle" >
    <const>hintfull</const>
  </edit>
</match>
<match target="font" >
  <edit mode="assign" name="antialias" >
    <bool>>true</bool>
  </edit>
</match>
<match target="font" >
  <edit mode="assign" name="rgba" >
    <const>none</const>
  </edit>
</match>
</fontconfig>
```

Beispiel 3:

Schrift-Hinting (BCI), Kantenglättung durch Subpixel-Hinting, TFT-Monitor (RGB)

```
<?xml version="1.0"?>
<!DOCTYPE fontconfig SYSTEM "fonts.dtd">
<fontconfig>
<dir>~/fonts</dir>
<match target="font" >
  <edit mode="assign" name="hinting" >
    <bool>>true</bool>
  </edit>
</match>
<match target="font" >
  <edit mode="assign" name="hintstyle" >
    <const>hintfull</const>
  </edit>
</match>
<match target="font" >
  <edit mode="assign" name="antialias" >
    <bool>>true</bool>
  </edit>
</match>
<match target="font" >
  <edit mode="assign" name="rgba" >
    <const>rgb</const>
  </edit>
</match>
</fontconfig>
```

Einstellungen systemweit vornehmen

Die zuvor beschriebenen Einstellungen sind lediglich für den jeweiligen Benutzer wirksam. Sollen die Einstellungen systemweit gelten, gibt es andere Möglichkeiten. Neben der zuvor beschriebenen Datei "*~/fonts.conf*" gibt es noch eine globale Konfigurationsdatei "*fonts.conf*" in "*/etc/fonts/*"

Ein Auszug aus "*/etc/fonts/fonts.conf*":

```
<!--
  Load per-user customization file
-->
  <include ignore_missing="yes">~/fonts.conf</include>

<!--
  Load local system customization file
-->
  <include ignore_missing="yes">conf.d</include>
  <include
ignore_missing="yes">local.conf</include>
```

Der Befehl *<include>* bewirkt das Laden weiterer Konfigurationsdateien, neben der zuvor beschriebenen "*~/fonts.conf*" der jeweiligen Benutzer werden also - sofern vorhanden - eine Datei "*local.conf*" und weitere Dateien im Verzeichnis "*conf.d*" geladen.

In Mandriva existiert die Datei "*/etc/fonts/local.conf*" nicht - eine Möglichkeit, systemweite Einstellungen vorzunehmen wäre also die Erstellung einer solchen Datei im XML-Format.

Die zweite Möglichkeit stellen die Dateien im Verzeichnis "*/etc/fonts/conf.d*" dar. Als Besonderheit werden hier nur Dateien berücksichtigt, die die Form *[0-9][0-9]** haben.

Im Verzeichnis "*/etc/fonts/conf.d*" sind bereits einige Dateien:

- * 00-mdk-urwfonts.conf
- * 01-mdk-CJK-dualwidth.conf
- * 02-mdk-disable-antialias.conf
- * 03-mdk-disable-hinting.conf
- * 04-mdk-avoid-bitmap.conf
- * autohint.conf
- * no-bitmaps.conf
- * no-sub-pixel.conf
- * sub-pixel.conf
- * unhinted.conf
- * yes-bitmaps.conf

Möchte man z.B. den AutoHinter aktivieren, genügt es die Datei "*autohint.conf*" beispielsweise in "*05-autohint.conf*" umzubenennen, oder noch eleganter - einen symbolischen Link auf "*autohint.conf*" mit führender Nummer zu erzeugen:

```
ln -s autohint.conf 05-autohint.conf
```

Weitere Anmerkungen:

Manche Programme reagieren nicht auf diese Einstellungen, da sie die Schriften selbst rendern. Zu solchen Programmen gehören z.B.

- * OpenOffice
- * Acrobat Reader
- * KGhostview
- * Kpdf
- * Xpdf

DPI-Wert des Monitors zum Schrift-rendern einstellen:

Zuvor wurde kurz der Unterschied zwischen den Schriftgrößenangaben "px" und "pt" angeschnitten - während "pt" eine Längeneinheit ist (1pt entspricht 1/72 Zoll), beschreibt "px" dagegen die tatsächlich auf dem Monitor dargestellte Schriftgröße in Pixeln.

Aus wievielen Pixeln wird nun aber eine Schrift mit der Größe 12pt gerendert? Zur Beantwortung dieser Frage muß etwas weiter ausgeholt werden:

Der Font-Renderer muß zur korrekten Darstellung von Schriftgrößen nicht nur die Auflösung des Monitors in Bildpunkten kennen, sondern auch die tatsächlichen Abmessungen des Bildschirms. Hierfür kann in der Datei "*/etc/X11/xorg.conf*" den Eintrag "DisplaySize" gesetzt werden, der bei einer Standardinstallation von Mandriva NICHT existiert. Dies kann man natürlich nachholen:

Auszug aus einer modifizierten "*xorg.conf*":

```
Section "Monitor"
| | ...
| | DisplaySize 375 300
| | ...
```

Der Parameter wie im Beispiel oben gibt dem X-Server vor, daß ein Monitor mit den Abmessungen 375mm x 300mm am Computer angeschlossen ist. Zusammen mit der verwendeten Pixel-Auflösung (z.B. 1.280x1.024 Bildpunkte) resultiert daraus eine Monitorauflösung von ca. 87dpi (Rechnung: 1.280 / 375 * 25,4).

Moderne Monitore sind DDC-fähig (DDC: Display-Data-Channel), d.h. der X-Server holt sich die Werte direkt vom Monitor. Eine Angabe von "DisplaySize" in der "*xorg.conf*" wird daher überflüssig - vorausgesetzt die Werte die der Monitor an den Computer sendet sind richtig.

Falls man noch einen älteren Monitor ohne DDC-Funktion besitzt und "DisplaySize" nicht definiert ist, nimmt der X-Server eine Bildschirmauflösung von 75dpi an.

Anmerkung: Unter welcher Auflösung der X-Server derzeit betrieben wird kann man herausfinden, indem man z.B. in der Konsole eintippt:

```
xdpinfo
```

Die Auflösung des X-Servers hat aber nicht unbedingt Auswirkungen auf die Schriftgrößen. Denn speziell für's Schrift-Rendering gibt es in Linux noch eine zweite Möglichkeit:

Man kann dem Font-Renderer direkt - unabhängig von der Auflösung des X-Servers - die Auflösung des Monitors mitteilen. Dies geschieht mit dem Parameter "Xft.dpi" in der Datei "/etc/X11/Xresources". Bei Mandriva ist dieser Wert als Standard auf 90dpi gesetzt:

```
Xft.dpi: 90
```

Dieser Parameter hat für das Schriftrendern Vorrang vor den DPI-Einstellungen des X-Servers, d.h. sobald "Xft.dpi" gesetzt ist, werden die einzelnen Schriftgrößen mit immer gleicher Pixelzahl gerendert, egal mit welcher Auflösung der X-Server läuft.

So hat man also unter Linux die Wahl:

Sollen die Schriftgrößen am Bildschirm 1:1 dem späteren Ausdruck entsprechen, kommentiert man "Xft.dpi" einfach aus. Voraussetzung sind natürlich korrekt eingestellte DPI-Werte des X-Servers. Je nach Monitorauflösung kann die Lesbarkeit der Schriften aber darunter leiden.

Sollen die Schriftgrößen am Bildschirm immer mit gleicher Pixelanzahl gerendert werden, setzt man "Xft.dpi" auf einen bestimmten Wert. Dadurch erreicht man bei beliebiger Monitorauflösung ein gleich gutes Schriftbild, die tatsächliche Schriftgröße kann dabei variieren.

Persönlich bevorzuge ich letztere Variante.

Im übrigen rendert auch MS-Windows Schriften unabhängig von den tatsächlichen Bildschirmabmessungen - mit 96dpi. Das ist der Grund, warum unter Windows eine 10pt-Schrift etwas größer ausfällt als unter Mandriva. Will man also Schriftgrößen wie unter Windows, setzt man "Xft.dpi" einfach auf 96dpi.

Linux-Schriften - die Spreu vom Weizen trennen:

Unter Mandriva werden bei einer Standard-Installation eine Vielzahl an Schriften installiert. Doch nur einige wenige sind mit größerer Sorgfalt erstellt worden - oft enthalten die Schriften weder Umlaute noch Sonderzeichen.

Mit dem Programm "FontViewer" (kfontview) kann man sich Schriften anzeigen lassen und auf das Vorhandensein von Umlauten und Sonderzeichen testen. Der "FontViewer" zeigt die Schriften in verschiedenen Größen mit folgendem Standard-Text an:

"The quick brown fox jumps over the lazy dog"

Diesen ändert man - indem man auf das kursive "T" klickt - einfach ab in:

```
Jeder größere Übersee-Riese aus Ägypten kostet 150€
```

Nun kann man mit diesem neuen Text die installierten Schriften begutachten. Unter Mandriva befinden sich Schriften üblicherweise in folgenden Verzeichnissen:

```
* /usr/share/fonts
* /usr/X11R6/lib/X11/fonts
* ~/.fonts
```

Diese Schriften sind für die Texterstellung mit Umlauten in meinen Augen völlig unbrauchbar:

```
* /usr/share/fonts/ttf/decoratives
* /usr/share/fonts/ttf/western
* /usr/share/fonts/otf/mdk
```

Viele Schriften enthalten zwar noch Umlaute, aber beispielsweise kein €-Zeichen mehr - so gibt es für Mandriva keine einzige Courier-Schrift mit €-Zeichen! In Ermangelung dieses Schriftzeichens wird z.B. bei der Verwendung der Schrift "Courier 10 Pitch" auf eine alternative Schrift zurückgegriffen, die dieses Zeichen enthält - das €-Zeichen schaut für meine Begriffe dann aber wie ein Fremdkörper im Gesamtschriftbild aus.

Einige wenige TTF-Schriften sind wohl vollständig:

```
* sämtliche "Bitstream Vera" (serif, sans, mono)
* sämtliche "DejaVu"-Schriften (serif, sans, mono)
* sämtliche "Luxi"-Schriften (serif, sans, mono)
```

Bevorzugte Schriften einer Font-Familie - Reihenfolge anpassen:

Die Schriften sind in verschiedene Font-Familien eingeordnet. Die gängigsten sind:

```
* serif
* sans-serif
* monospace
```

Diese Kategorisierung ist gerade bei der Gestaltung von Webseiten eine übliche Methode um den Browser anzuweisen, welcher Schriftstil bei der Anzeige einer Seite verwendet werden soll. Der HTML-Befehl hierzu lautet "font-family". Ein typischer Eintrag im Quelltext einer HTML-Seite lautet z.B.:

```
font-family: Verdana, Arial, sans-serif
```

Auch auf den Seiten von MandrivaUser.de wird das beispielsweise so gemacht. Dies bedeutet, daß der Browser nach Möglichkeit die Schriftart "Verdana" verwenden soll, falls die nicht auf dem System installiert sein sollte "Arial", und falls auch diese nicht existiert wird eine Schrift aus der Familie der sans-serifen Schriften verwendet.

Woher "weiß" nun aber der Browser, welche Schrift er nun genau als sans-serife Schrift verwenden soll?

Im Browser Firefox kann man unter "Bearbeiten - Einstellungen - Allgemein - Schriften & Farben" die zu verwendende Schrift für die jeweilige Font-Familie definieren. Allerdings sind dort als Standard-Konfiguration ebenfalls nur die Font-Familien "serif", "sans-serif" und "monospace" angegeben.

Wieder einmal ist es die globale Konfigurationsdatei "fonts.conf", die in so einem Fall weiterhilft.

Ein Auszug aus `"/etc/fonts/fonts.conf"`:

```
<!--
Provide required aliases for standard names
-->
<alias>
  <family>serif</family>
  <prefer>
    <family>DejaVu Serif</family>
    <family>Bitstream Vera Serif</family>
    <family>Times New Roman</family>
    <family>Thorndale AMT</family>
    <family>Luxi Serif</family>
    <family>Nimbus Roman No9 L</family>
    <family>Times</family>
    ...
  </prefer>
</alias>
<alias>
  <family>sans-serif</family>
  <prefer>
    <family>BPG Glaho
International</family> <!-- lat,cyr,arab,geor -->
    <family>DejaVu Sans</family>
    <family>Bitstream Vera Sans</family>
    <family>Luxi Sans</family>
    <family>Nimbus Sans L</family>
    <family>Arial</family>
    <family>Albany AMT</family>
    <family>Helvetica</family>
    <family>Verdana</family>
    <family>Lucida Sans Unicode</family>
    <family>Tahoma</family>
<!-- lat,cyr,greek,heb,arab,thai -->
    ...
  </prefer>
</alias>
<alias>
  <family>monospace</family>
  <prefer>
    <family>DejaVu Sans Mono</family>
    <family>Bitstream Vera Sans Mono</family>
    <family>Luxi Mono</family>
    <family>Nimbus Mono L</family>
    <family>Andale Mono</family>
    <family>Courier New</family>
    <family>Cumberland AMT</family>
    <family>Courier</family>
    ...
  </prefer>
</alias>
```

In Mandriva 2006 ist sind weder die Microsoft-Schriften "Verdana" und "Arial", noch die sans-serife Schrift "BPG Glaho International" installiert, also wird für das Beispiel (font-family: Verdana, Arial, sans-serif) die nächste Schrift aus der Auflistung für sans-serife Schriften verwendet - "DejaVu Sans".

Allerdings ist die Darstellungsqualität am Bildschirm der DejaVu-Schriften für meinem Geschmack nicht so besonders, was vor allem bei der Kursivdarstellung der Schrift "DejaVu Serif" auffällt - das Schriftbild mit aktiver Kantenglättung wirkt sehr verwaschen. Auf der Homepage der freien DejaVu-Schriften

<http://dejavu.sourceforge.net/>

werden die Schriften z.T. als noch "experimentell" bezeichnet, was der Grund für obiges Verhalten sein dürfte. Meiner Meinung nach wesentlich besser eignen sich die Schriften "Bitstream Vera". Stellt man die Reihenfolge der Schriften in der globalen "fonts.conf" einfach um, werden in Zukunft die gewünschten Schriften verwendet. Die Modifikation der globalen "fonts.conf" hat auch den Vorteil, daß nicht für jedes einzelne Programm diese Einstellungen vorgenommen werden müssen (KDE, Firefox, Thunderbird, etc.).

Nützliche Links zur Nachlese:

<http://fontconfig.org/fontconfig-user.html>

<http://www.bwalle.de/docs/schriften.pdf>

<http://avi.alkalay.net/linux/docs/font-howto/Font.html>

<http://www.antikoerperchen.de/deutsch/howto-24.php>

<http://scanline.ca/dpi/>

<http://grc.com/ctwhat.htm>

<http://www.freetype.org/freetype2/index.html>

<http://www.freetype.org/patents.html>

Anmerkung d. Herausgebers:

Dieser Artikel liegt im Bereich „HowTos“ des Forums <http://www.mandrivauser.de> vor und wird bei Bedarf aktualisiert und ergänzt.