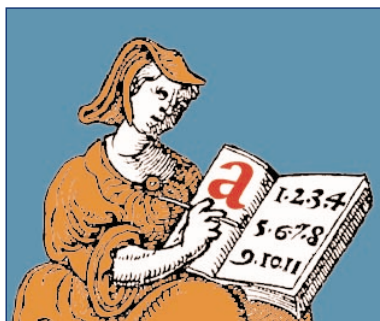


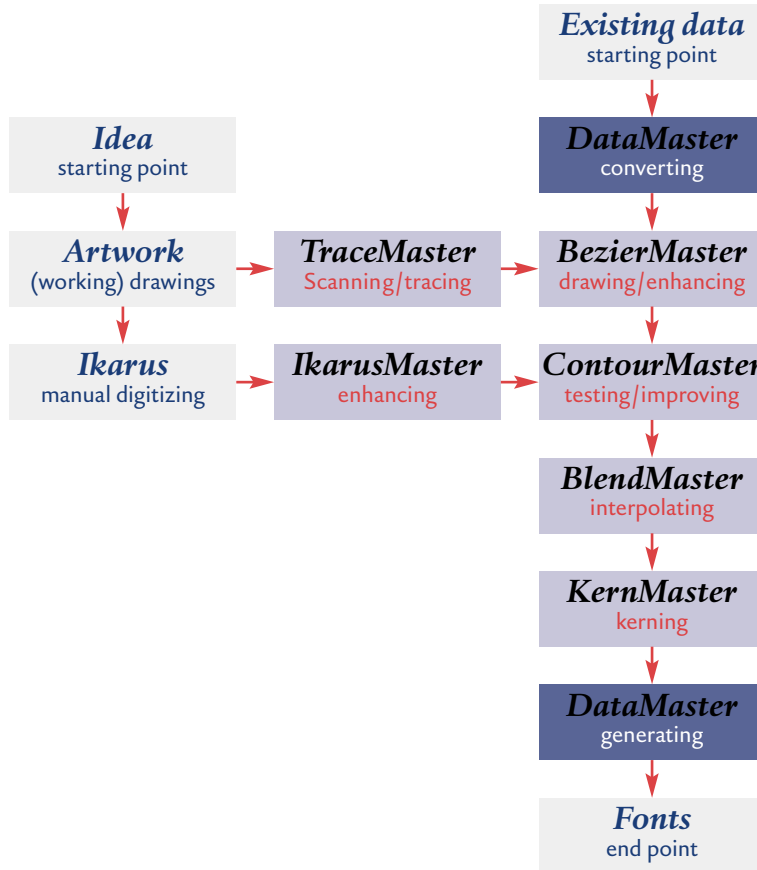
Dutch Type Library

DTL DataMaster



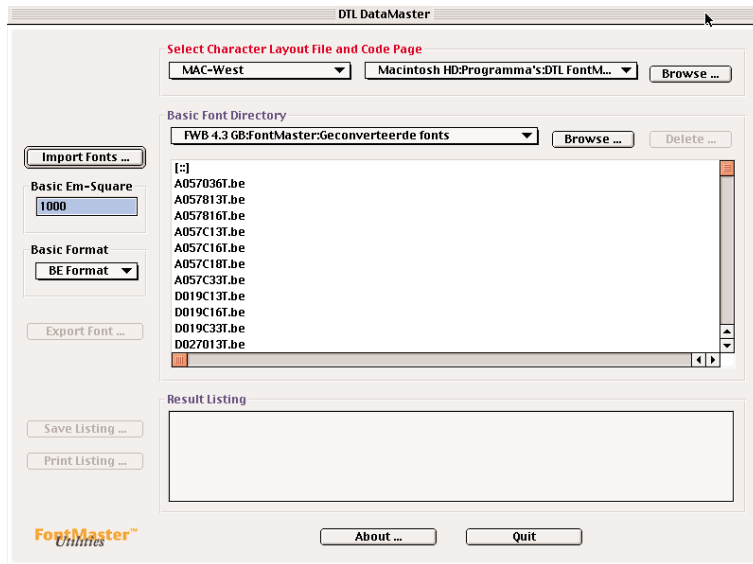
's-Hertogenbosch/Hamburg
Autumn 2004

The diagram shows a typical workflow based on the modules of DTL FontMaster.



DTL DataMaster is the module for managing, converting and generating database and font formats. DTL DataMaster converts the PostScript Type 1 and TrueType fonts for Mac OS and Windows to the BE and IK formats that are used by the DTL FontMaster utilities. DTL DataMaster can also generate PostScript Type 1 and TrueType fonts for Mac OS and Windows. In addition, DTL DataMaster creates OpenType fonts according to the Adobe standard. OpenType support in DTL DataMaster is based on Adobe's OpenType SDK and therefore all existing features from Adobes Pro fonts are covered. DTL DataMaster also provides an easy way to generate OTFs also for nonexperts; simply selecting different layouts and/or feature files is sufficient to create fonts with different character and feature sets.

The hints that DTL DataMaster generates for the TrueType fonts can be imported into Microsoft's VTT (Visual TrueType) and used as a basis for delta hinting. The data associated with naming the fonts for the various formats for Mac OS and Windows is saved in special database files, which are platform independent.



Starting DTL DataMaster

The first dialog has two major functions:

- *Import fonts*
- *Export fonts*

Furthermore you can select a Character Layout File (*.cha) and to change the resolution of the imported font. DTL DataMaster accepts PostScript Type 1, TrueType, OpenType (OTF and TTF), BE and IK format for input. The BE format can be converted to IK format and vice versa. From the BE and IK format it is possible to generate PostScript Type 1, TrueType and OpenType (OTF and TTF) fonts.

OpenType is a rich specification which allows thousands of possible combinations of language lookups and features. Its quite obvious that writing a GUI for the OpenType tables is a huge task. The DTL FontMaster approach is to try to make it easy to generate an OpenType font.

- The OpenType production is based on Adobe's SDK.
- Both OTF and TTF production are supported.
- DTL DataMaster automatically generates as many features as possible.
- Advanced users can create their own set of features.
- No fancy graphic user interface.

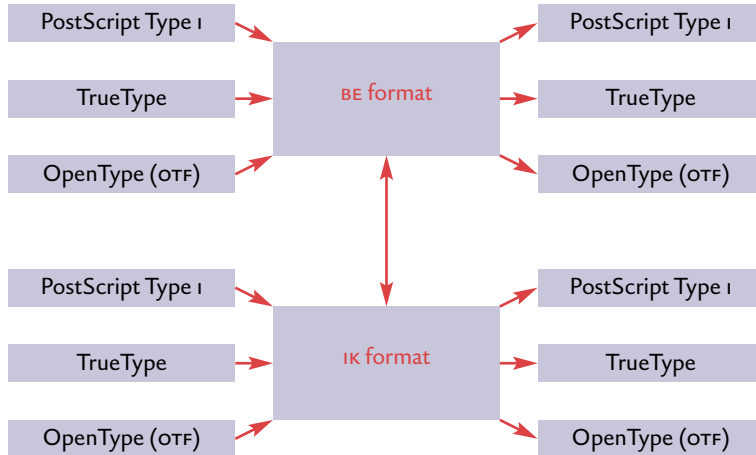
The OpenType production is essentially governed by two files:

- The Character Layout File.
- The OpenType Feature File.



For more information, see Appendix vi: OpenType Technology.

DTL DATAMASTER: STARTING



FontMaster™
Utilities

DTL DataMaster accepts PostScript Type 1, TrueType, OpenType, BE and IK format for input. The BE format can be converted to IK format and vice versa. From the BE and IK format it is possible to generate PostScript Type 1, TrueType and OpenType fonts.

```
C urwotf.cha
C 20020403
Version 002.000
Starttable
URWNum;urwCOMP;urwCOMP;UNINum;ANNum;QDNum;PSNum;PSName;KernClass;StatClass
999;;;x0020;32;32;32;space;;
101;;;x0041;65;65;65;A;V;
102;;;x0042;66;66;66;B;V;
103;;;x0043;67;67;67;C;V;1
104;;;x0044;68;68;68;D;V;1
105;;;x0045;69;69;69;E;V;1
106;;;x0046;70;70;70;F;V;
107;;;x0047;71;71;71;G;V;
108;;;x0048;72;72;72;H;V;1
109;;;x0049;73;73;73;I;V;1
```

OpenType production is essentially governed by the Character Layout File and the OpenType Feature File. The urwotf.cha is the default Character Layout File for the OpenType production.

1. Import Fonts ...

Before fonts can be imported a series of selections have to be made, such as the selection of the Character Layout File, the Basic Format and, of course, the Basic Font Directory.



By default the *.cha files are installed in the same directory as the program files.

1.1 Character Layout File and Code Page

The code page defines the layout of the font. A code page consists of 256 character slots, of which a part is used for system functions. There are different code pages for many scripts, such as West-European, East-European, Greek and Cyrillic. There are differences between the layout of

the fonts for Mac OS and Windows; the code pages for West-European are not the same on both platforms. The selection of the code page only has influence on an exported font. It is of no importance when importing a font into DTL DataMaster. The choice of the Character Layout File (*.cha) is very important when a font is imported. A BE or IK database does not contain any information concerning the PostScript name of a character (like *adieresis* for the ä) or Unicode number (00E4 for the same character).

Based on the original character names and Unicode numbers of the original PostScript Type 1 or TrueType fonts, each character gets a decimal Character Number in the BE or Ikarus database. This process is reversed when a font is generated; each character gets a PostScript name based on its Character Number and in case of TrueType and OpenType fonts, a Unicode number as well. For more info, see the appendices III and IX.

For importing and exporting PostScript Type 1 or TrueType fonts *beeditor.cha* should be selected. For importing and exporting OpenType fonts select the *urwotf.cha* file. An OpenType font produced with the *beeditor.cha* will possibly not support all OTF features because of the simple fact that not all characters will be exported.

1.2 Basic EM-Square

The resolution of the description of a font can vary, although there are standards for the different font and database formats. The resolution is defined in the *EM-square* and its units have no absolute size nor any relation to the resolution of any output device. The standard for PostScript fonts is 1000 x 1000 units and for TrueType 2048 x 2048 units. This does not mean that for these font formats no other EM-squares can be used. However the behaviour of such fonts can be unpredictable if the standards are not followed.

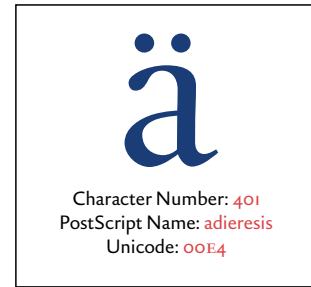
The default EM-square for the Ikarus format is 15000 x 15000 units. For the BE format an EM-square of 1000 x 1000 units is recommended. The smallest EM-square supported by DTL DataMaster is 256 x 256 units.

1.3 Basic Format


There are two options: BE format and Ikarus format. The selection normally depends on the editor that will be used: DTL BezierMaster or DTL IkarusMaster. In case DTL DataMaster is used only for converting PostScript Type 1 or TrueType fonts into other formats the default BE format is normally the best choice.

1.4 Basic Font Directory

The Basic Font Directory is the place where the converted fonts are stored. So, although named 'Basic' this is the *target* directory. By default this is the directory where the FM modules are installed. The *Browse ...* button can be used to select a different directory or folder.



For more technical details about the Character Layout Files and Character Numbers, see Appendices III and IX.

 **NOTE:** The selection of the code page only has influence when a font is exported. However, the choice of the Character Layout File (*.cha) is very important when a font is imported.

For importing and exporting PostScript Type 1 or TrueType fonts the *beeditor.cha* file should be selected. For importing and exporting OpenType fonts select the *urwotf.cha* file.

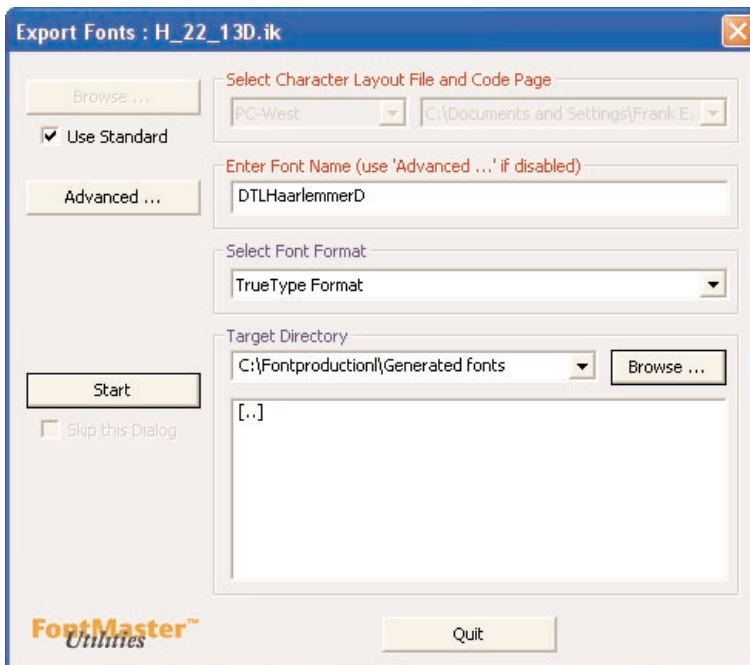


If a font is imported in DataMaster, all information concerning Font Naming and Copyright Information, Common Metrics Information, Unique PostScript ID, etc. is stored in the UFM file. This file gets the same name as the imported font. In case the file name of the font is changed, the name of the corresponding UFM file must be changed accordingly. Details about the UFM file are revealed in Appendix IV.

After all selections are made, the *Import Fonts ...* function can be selected and the fonts or BE and IK data can be imported from the *Open* dialog.

2. Export Fonts ...

For exporting fonts, first make the same selections as for importing fonts. Next the *Export Fonts ...* function must be selected for opening the *Export Fonts* dialog.



At the Dutch Type Library the database names consist of eight characters, like A057016T.be for DTL Argo T Bold (FullName) and F063C14T.ik for DTL Fleischmann ST Medium (FullName). The eight characters guarantee that exchanging the files between (older) platforms does not have any influence on the file names.

2.1 Character Layout File and Code Page

Before a font can be generated, the Character Layout File and the Code Page must be selected. For PostScript Type 1 and TrueType fonts the Character Layout File *beeditor.cha* file is recommended. For the production of OpenType fonts, the *urwotf.cha* file must be selected to preserve the implementation of all the OTF features. The selection of the code page depends on which script the font has to support. Take care that if characters are not available in the font database, the generated font will be incomplete. Use the *Font Administration* tool in *BezierMaster* or *IkarusMaster* to check if your font database supports the character set of the code page. Also be aware that if characters are stored in slots that do not correspond with the Character Layout Files, the generated code pages will be incorrect.

2.2 Enter Font Name

Enter the FontName here. The first time a font database is selected an UFM file will be generated. The FontName is taken from the Font Header and will be shown here. The FontName will be used by the program to generate automatically the FamilyName, FaceName, FullName, FondName, etcetera.

In case you want to control the font naming completely, select the *Advanced ...* button. You have to do this also when the *Enter Font Name* function is disabled. If a UFM file already has been generated, you can not change the Font Name in the *Export Fonts* dialog.

A UFM file contains all information concerning Font Naming and Copyright Information, Common Metrics Information, Unique PostScript ID, etcetera. This file is automatically generated by DataMaster in case a PostScript Type 1 or TrueType font is converted into a BE or IK database or when changes in the *Advanced: view/change UFM and PAR file entries* dialog are saved for the first time. The file is placed in the same directory as the related font database. The UFM file is stored in ASCII format and can be edited directly in a text editor. The UFM file is platform independent; you can move it between Mac OS and Windows. More information about the UFM file format can be found in Appendix IV.

```
Version 002.00E
FamilyName DTLDocumentaST
FontName DTLDocumentaST-Bold
FullName DTL Documenta ST Bold
UniqueID 5060740
Weight Bold
IsFixedPitch false
Ascender 766
Descender -234
UnderlinePosition -133
UnderlineThickness 20
Bodysize 1000
```



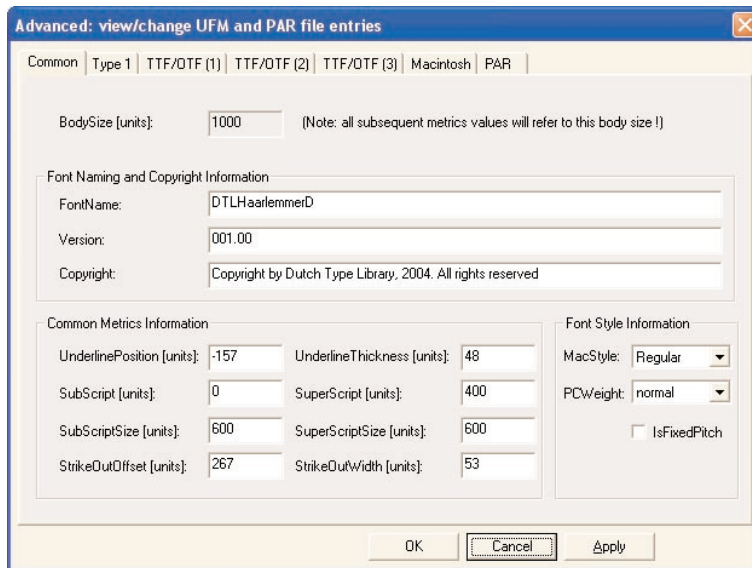
Use the *Font Administration* tool in *BezierMaster* or *IkarusMaster* to check the code pages you want to generate with *DataMaster*.

Part of a UFM file. The plain text file is stored in the same directory as the font database. The UFM file can be altered directly using a text editor.

The way to handle the naming of the font can differ. The mostly used naming conventions for Mac OS and Windows show differences. On a Mac it is common practice to put all the weight/style names behind the hyphen (FamilyName-MediumItalic), even if this is a medium weight. On a Windows system such naming will not be recognized by the system, except for the Regular and Bold weight. So for Windows the resulting name would be: FamilyNameMedium-Italic. To preserve an optimal exchange of documents between Macintosh and PC with the use of the same fonts, the naming for both platforms should be identical based on the Windows convention, although for this purpose identical *FondNames* and *FullNames* should work in most cases.

2.2.1 Advanced ...

The *Advanced ...* button opens the *Advanced: view/change UFM and PAR file entries* dialog. Here you can control all details concerning for example the Font Naming, Common Metrics Information and Table Information. Be aware of the fact that the behaviour of your font depends on what you enter in the following dialogs:



– Common

Here you can change the *FontNaming and Copyright Information*. Changing the *FontName* will not change automatically any of the naming in the other UFM dialogs; you will have to do this manually. The *FontName*, also sometimes called the *PostScript Name* must not contain any spaces or special characters, with exception of the hyphen or underscore. It can be a condensation of the *FullName* by removing the spaces. It is customary to

A typical example of font naming, as used by the Dutch Type Library.

For Mac OS:
 File: A057014T
 FontName: DTLArgoT-Medium
 FamilyName: DTLArgoT
 Weight: Medium
 FondName: DTL Argo T Medium
 MacStyle: Regular

For Windows:
 File: A057014T
 FontName: DTLArgoTMedium
 FamilyName: DTLArgoTMedium
 Weight: Regular
 FaceName: DTLArgoTMedium
 FullName: DTL Argo T Medium
 PCWeight: Medium

All details about the items in the UFM dialogs can be found in Appendix IV.

limit its length to less than 40 characters.

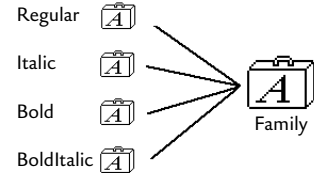
The *Common Metrics Information* is taken from the font that originally was converted in the BE or IK database and in case of a new font it is generated automatically. Any other value can be entered here.

The *MacStyle* in the *Font Style Information* is used on the Macintosh for building font families and is used in the OS 2 tabel for TrueType and OpenType fonts. The *PCWeight* has only an informative function under Windows.

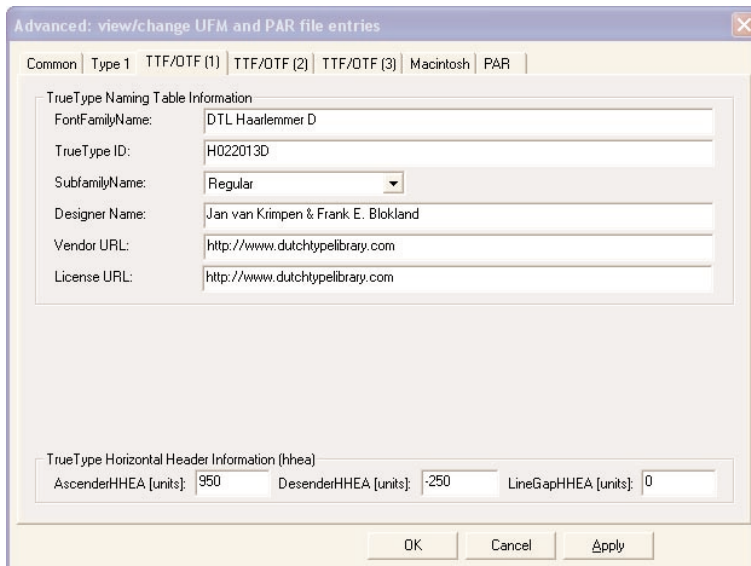
In case you are not going to generate a font family on the Macintosh (an option in the UFM *Macintosh* dialog), always select 'Regular', even if the font is Bold. Otherwise select the appropriate style. The *MacStyle* information does not influence the actual naming but offers the possibility –when a font family is built– to change styles using shortcuts.

The *PCWeight* should represent the weight of the font but in case this is not covered by the nine defined styles ('thin', 'extra light', 'light', 'normal', 'medium', 'semi bold', 'bold', 'extra bold' and 'heavy'), 'Regular' should be selected.

The option *IsFixedPitch* should only be activated for a monospaced font, like Courier.



Although not common practice, on a Macintosh the styles 'Regular', 'Italic', 'Bold' and 'BoldItalic' can be assembled in a font family.



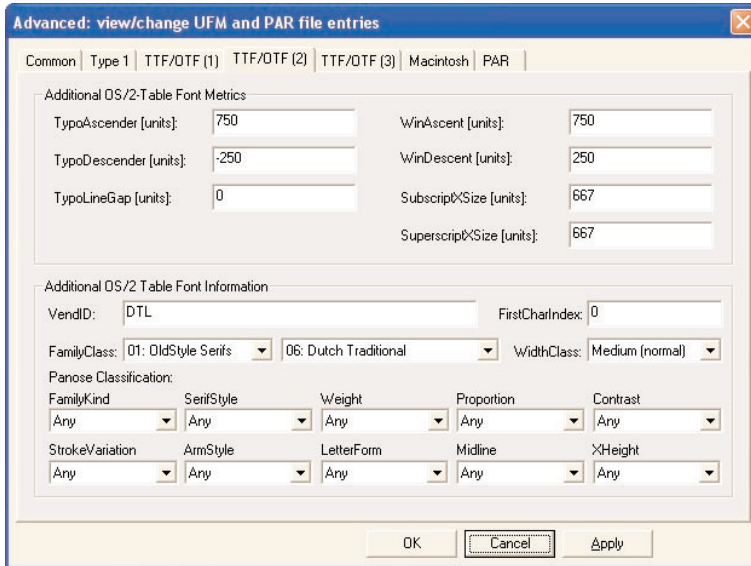
NOTE: For TrueType and OpenType fonts the font style information for the MacOS is taken from the MacStyle info in the Common dialog.

– TTF/OTF (1)

Here you can enter the *TrueType Naming Table Information*. The *FontFamilyName* is normally identical to the *FaceName* in the UFM *Type 1* dialog. The *TrueType ID* entry should be an unique name and is automatically filled in by the program. At the Dutch Type Library the file

name of the font database is also used as the unique TrueType ID.

The *TrueType Horizontal Header Information* (HHEA) is generated by the program automatically.



The values for the *TypoAscender* and *TypoDescender* are by default taken from the information entered in the *Font Header* in *DTL BezierMaster*.

– TTF/OTF (2)

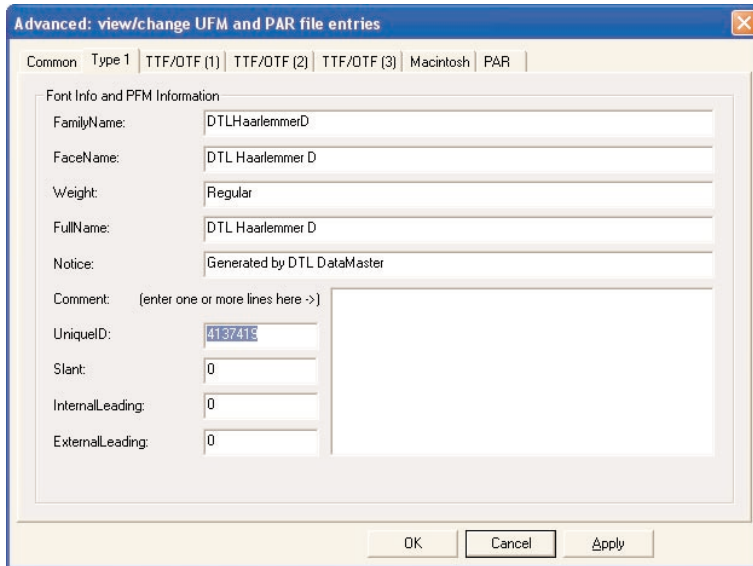
Here you can enter the *Additional OS/2-Table Font Metrics* and the *Additional OS/2-Table Font Information*. The values for the *TypoAscender*, *TypoDescender*, *WinAscent* and *WinDescent* can be defined here. The *TypoAscender* and *TypoDescender* form together the EM-square. The *WinAscent* and *WinDescent* are calculated automatically by the program. Normally the values for *WinAscent* and for *FondAscent* in the *Macintosh* dialog are the same. This is also the case for *WinDescent* and *FondDescent*.

The *TypoLineGap* is by default zero. The *SubscriptXSize* and *SuperscriptXSize* are also automatically calculated by the program.

The *VendID* can be entered in the *Additional OS/2-Table Font Information*. It may consist of four characters at maximum. It is common practice to register this ID via Microsoft, although this is not necessary.

By default the *FirstCharIndex* is set to zero. The default for the *WidthClass* is 'Medium' (normal) and this value should normally not be altered.

The *FamilyClass* and *Panose Classification* don't influence the behaviour of the font but are used to describe the characteristics. This can be of use in case the font has to substitute another font with the same characteristics. The default settings ('No Classification' and 'Any') will of course not support this system but do not harm anything either.



Here the *FullName* could also be: *DTL Haarlemmer D Regular*. In that case the *FondName* in the *Macintosh* dialog should normally be altered the same way.

– Type 1

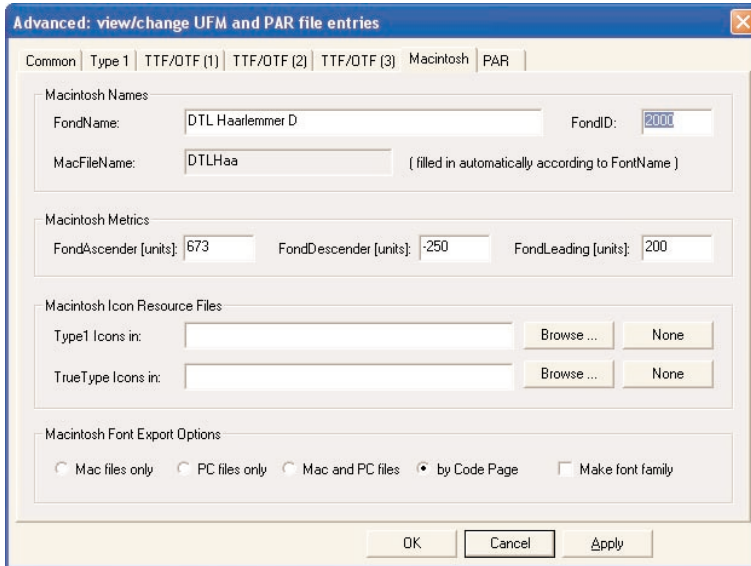
In this dialog most of the naming of the font can be controlled and you can enter the unique PostScript ID here.

The *FaceName* is normally identical to the *FontFamilyName* in the *TTF/OTF (1)* dialog. It may contain spaces. The *FaceName*, also known as *Windows Font Name*, is the name you will see in a Windows program in the font dialog. It is normally the part of the *FontName* before the hyphen; the *FaceName* should not contain style information, like *Regular*, *Italic*, *Bold* or *BoldItalic*. This means that in case more fonts with different styles are part of a family, the *FaceName* should be identical in all these fonts. The same is, of course, the fact for the *FamilyName*. The *FaceName* and *FamilyName* can also be identical, although it is allowed to put spaces in the *FaceName*. Although not common practice, the *FaceName* can differ completely from the *FamilyName*. The *Weight* indicates the boldness of the font. The *FullName* has a commentary purpose. It may contain spaces and looks normally like the *FontName* whereby the name and style parts are separated by a space instead of a hyphen. The *Notice* box can be used to put global information, like a trademark or copyright notice in the font.

In the *Comment* area you can add extra information about for instance the font or the designer.

The *UniqueID* is used to store font information in the cache of a PostScript printer. This way the next time the font is used, it will print faster. It is allowed to fill in just a zero for the *UniqueID* or to enter a number between 4.000.000 and 5.000.000. This range is allocated for free use. For professional font production it is recommended to obtain *UniqueID* numbers from Adobe Systems Inc.

Slant indicates the italic angle counter-clockwise from the vertical. *InternalLeading* and *ExternalLeading* can be used to influence the spacing between the lines in an application. The default is in both cases zero.



Here the *FondName* could also be: *DTL Haarlemmer D Regular*. In that case the *FullName* in the *Type 1* dialog should normally be altered the same way.

– *Macintosh*

With this dialog all UFM File information relevant for the use of the font under Mac OS can be controlled.

The *FondName* is the name that will appear in the menu with the font information of an application. It may contain spaces and could be made identical to the *FullName* in the *UFM Type 1* dialog. The *MacFileName* is used by the operating system and it is calculated automatically by DataMaster from the *FontName*. The *MacFileName* has a fixed structure and is constructed as follows: 5-3-3-3-3-3... This means that of the first part of the *FontName*, which always has to start with a capital, the first five characters are taken. Of the following parts that start with a capital the first three letters are taken. In case the *FontName* is *TestingMyFont-Italic*, the *MacFileName* will be *TestiMyFonIta*.

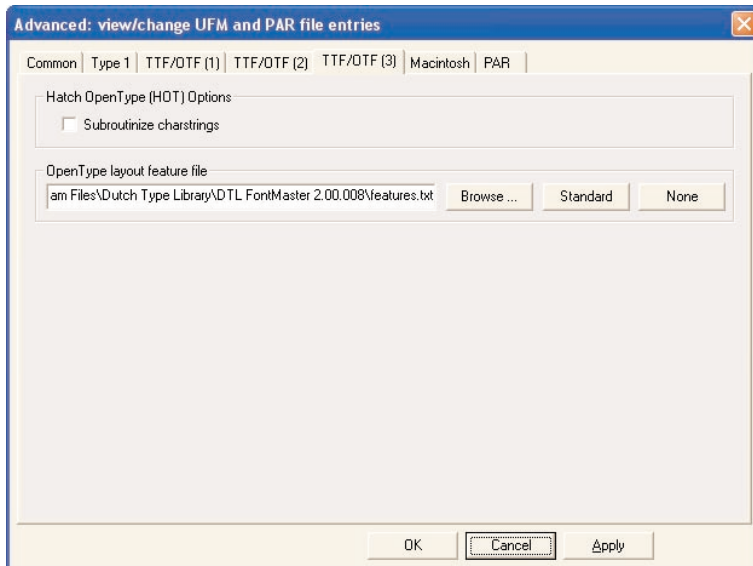
The *FondID* is used by the operating system to identify the font. Each font should have a unique Font Family (Fond) ID. There are ranges for the different scripts. For Roman the range 1024 – 16382 has been defined by Apple for commercial use. Fond ID conflicts should be solved by the operating system automatically. See Appendix IX for more information.

The parts that make the *Macintosh Metrics* information: the *FondAscender*, the *FondDescender* and the *FondLeading* should normally be identical to the

Additional Os/2-Table Font Metrics information (respectively the *TypoAscender*, the *TypeDescender* and the *TypoLineGap*) in the *UFM TrueType (2)* dialog.

The fonts generated with DataMaster will show generic icons. It is possible to select proprietary icons for Type 1 and TrueType fonts from the program by browsing to a directory that contains the *Macintosh Icon Resource Files*. The default icons reside in the directory that contains the DTL FontMaster program files. Be aware of the fact that the icons you select for Type 1 fonts are a somewhat artificial solution; the icons are pasted on the generic icons. The creator information of the font will not be changed. To change the Finder Icons and the BNDL information, etcetera, use DTL IconMaster. You will find details about this small program in Appendix XI.

You can use the Macintosh version of DataMaster to generate fonts for Mac OS and PC through the *Macintosh Font Export Options*. This can be done separately but also in batch. Take care of selecting the appropriate code page in the *Export Fonts* dialog. Generating for instance a font for PC with the Mac-West code page selected will result in a non-standard font for the PC. Also batching the production of the fonts for Mac OS and PC will always result in one non-standard font.



More information about the OpenType format can be found in Appendix v.

– TTF/OTF (3)

DataMaster supports the production of OpenType (.OTF) in a very intelligent way. Features are generated automatically on basis of the available characters in the font database. The OpenType support is based on Adobe's OpenType SDK. The CFF conversion and the generation of the

GPOS and GSUB tables are integrated. The approach for generating the typographic features is based on the fact that all GSUB features can be generated automatically if the necessary glyphs are in the font. Contextual substitutions are also supported as well as changes to the language system behaviour. There is currently no support by DataMaster for GPOS features, except for kerning.

The default setting for the *Hatch OpenType (HOT) Options* is off.

The *OpenType Layout Feature File* is installed under the name *features.txt* in the same directory as the DTL FontMaster program files. The feature file can be edited by an experienced user.

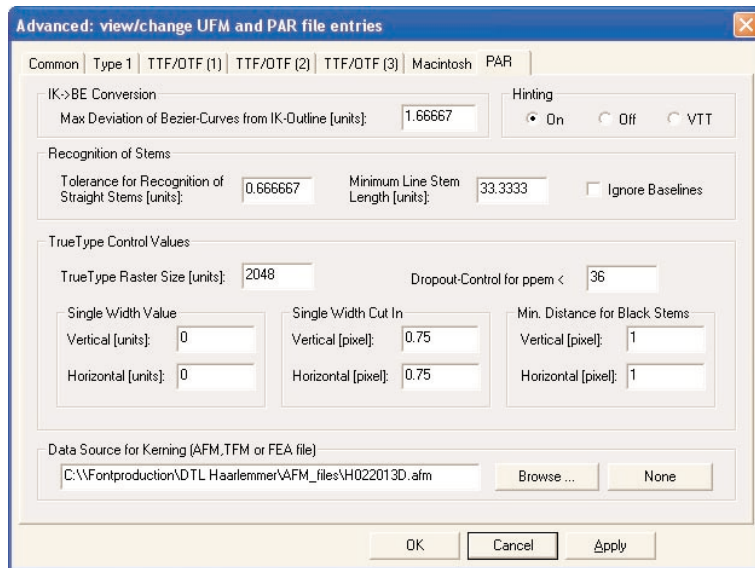
```
# DTL FontMaster Pro Feature File# Version 01.00# -- glyph groupings
@LETTERS_UC = [ A - Z AE Lslash Oslash OE Thorn Eth Aacute Abreve
Acircumflex Adieresis Agrave Amacron Aogonek Aring Atilde Cacute Ccaron
Ccedilla Dcaron Deroat Eacute Ecaron Ecircumflex Edieresis Edotaccent
Egrave Emacron Eogonek Gbreve Gcommaaccent Iacute Icircumflex Idieresis
Igrave Imacron Iogonek Kcommaaccent Lacute Lcaron Lcommaaccent Nacute
Ncaron Ncommaaccent Ntilde Oacute Ocircumflex Odieresis Ograve
Ohungarumlaut Omacron Otilde Racute Rcaron Rcommaaccent Sacute Scaron
Scedilla Scommaaccent Tcaron Tcommaaccent Uacute Ucircumflex Udieresis
Ugrave Uhungarumlaut Umacron Uogonek Uring Yacute Ydieresis Zacute
Zcaron Zdotaccent];@ACCENTS_UC = [ Acute Breve Caron Cedilla Circumflex
commaaccent.cap Dieresis Dotaccent Grave Hungarumlaut Macron Ogonek
Ring Tilde];@LETTERS_LC = [ a - z ae lslash oslash oe thorn eth aacute
abreve acircumflex adieresis agrave amacron aogonek aring atilde cacute
ccaron ccedilla dcaron deroat eacute ecaron ecircumflex edieresis
edotaccent egrave emacron eogonek gbreve gcommaaccent iacute
icircumflex idieresis igrave imacron iogonek kcommaaccent lacute lcaron
lcommaaccent nacute ncaron ncommaaccent ntilde oacute ocircumflex
odieresis ograve ohungarumlaut omacron otilde racute rcaron
rcommaaccent sacute scaron scedilla scommaaccent tcaron tcommaaccent
uacute ucircumflex udieresis ugrave uhungarumlaut umacron uogonek uring
yacute ydieresis zacute zcaron zdotaccent];@ACCENTS_LC = [ acute breve
```

Part of the DTL FontMaster Pro Feature File that DataMaster uses to generate OpenType fonts. Because the Adobe feature file is used (and also the syntax), everything which is supported by the Adobe SDK is supported by DataMaster too.

In the *Additional OpenType Name Table Entries* information about the designer and relevant URL's can be entered.

f + f + i = ffi

A typical example of a GSUB feature.



The values for for instance the *Max Deviation of Bezier-Curves from IK-Outline* and *Recognition of Stems* are automatically calculated by the program based on the size of the EM-square.

– PAR

In this dialog for the parameters necessary for the different conversions can be entered.

With the *IK-BE Conversion* the maximum deviation of Bezier curves from the *IK* outline can be controlled. Because the standard *BodySize* of an *Ikarus* font database is 15000 and the standard for *BE* is 1000, some scaling is involved with the conversion. The default value for the deviation is 25, assuming an *IK* *BodySize* of 15000. Because TrueType fonts are generated always from the *BE* format, the basic *BodySize* for all the related actions, like hinting, is 1000 units.

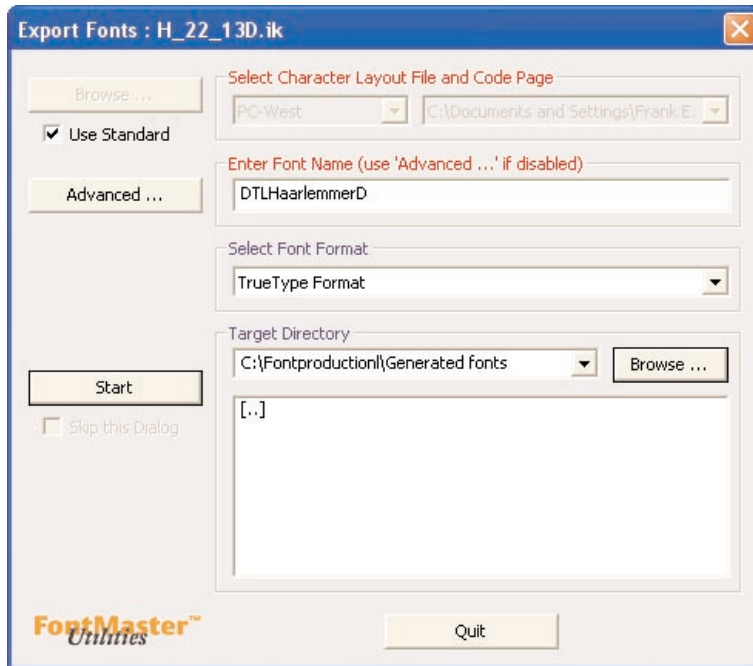
The *Hinting* options are *on*, *off* and *VTT*. The default setting is *on* but in case the of very complex contours, for instance in ornaments, the hinting can be put off. The *VTT* option makes it possible to generate hinting information that can be imported in VisualTrueType, the de facto program for delta hinting from Microsoft.


The *Recognition of Stems* parameters are for controlling the way stems are handled during hinting by DataMaster. The defined value for the *Tolerance for Recognition of Straight Stems* will make DataMaster handle stems with slight curves that are within this tolerance, as stems with straight lines. The default value is 10, assuming an *IK* *BodySize* of 15000. The *Minimum Line Length* is the minimal length of stems that will contain instructions. The default value is 500, assuming an *IK* *BodySize* of 15000.

In case the *BodySize* deviates from the standard 15000, DataMaster will calculate automatically the values for the *Tolerance for Recognition of Straight Stems* and the *Minimum Line Length* based on the default values. The default for the *Ignore Baselines* option is off.

The *TrueType Control Values* are meant for controlling the details of the conversion to TrueType. The default for the *TrueType Raster Size* is 2048; it is recommended not to alter this value. The *Dropout-Control for PPEM* indicates that the dropout control is activated for the PPEM's (pixels per EM square) that are smaller than the indicated value. The default is 36. This value can be up to 127 for extremely light weights.

The default for the *Single Width Value* is for both options (*Vertical* and *Horizontal*) zero units. The default value for the *Single Width Cut* is threequarter of a pixel for both directions. The *Min. Distance for Black Stems* has a default value for the *Vertical* and *Horizontal* of one pixel.



 **TIP:** Produce fonts in batch by activating the *Skip this Dialog* option.

2.3 Select Font Format

Select the font format you want to generate. Take care of the fact that for PostScript Type 1 and TrueType the Character Layout File *beeditor.cha* should be selected. For OpenType the *urwotf.cha* file must be selected, otherwise the resulting font will not contain all possible OTF features.

2.4 Target Directory

Select a target directory for the font that has to be generated.

2.4.1 Start

Starts the actual generation. Enable *Skip this Dialog* for batch production.