JustFontIt TEX Edition 1.1 for Linux

User Manual

Contents

1 Working with JustFontIt TEX Edition

JustFontIt TEX Edition (TE) is a command line program, that is it does not have a graphical user interface (GUI). The author's opinion is that such a design is mostly relevant for the JustFontIt TE purpose as a subsidiary tool for type professionals working with TEX. JustFontIt command line switches are straightforward and easy to use.

Command line switches are divided by tasks and options. Tasks are used to define input or output files to be processed; options specify additional parameters. So the command line looks as follows:

```
jfite [tasks] [options]
```

Note: square brackets mean that text inside them is optional.

Tasks and options are not necessarily must follow each other. However the order of tasks may be significant as explained below.

Read tasks take the form of

```
-R<format>[=<input file name>]
```

Write tasks take the form of

```
-W<format>[=<input file name>]
```

Options take the form

```
-<option>[optional text]
```

Note: angular brackets mean that text inside them is obligatory.

Here is description of all JustFontIt TE tasks and options.

1.1 Adobe Font Metric (AFM) Files

```
-Rafm[=<input file name>]
```

Purpose: read Adobe font metrics file.

1.2 Encoding Files

```
-Renc[=<input file name>]
```

`-Wenc[=<output file name>]`

Purpose: read/write PostScript encoding file.

Encoding file sets font encoding for 256 characters in PostScript notation. Here is an example of an encoding file content for Adobe Standard encoding:

```
/STDE [ /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef
/.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef
/.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef
/.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /space /exclam /quotedbl
/numbersign /dollar /percent /ampersand /quoteright /parenleft /parenright /asterisk
/plus /comma /hyphen /period /slash /zero /one /two /three /four /five /six /seven
/eight /nine /colon /semicolon /less /equal /greater /question /at /A /B /C /D /E /F
/G /H /I /J /K /L /M /N /O /P /Q /R /S /T /U /V /W /X /Y /Z /bracketleft /backslash
/bracketright /asciicircum /underscore /quoteleft /a /b /c /d /e /f /g /h /i /j /k
/l /m /n /o /p /q /r /s /t /u /v /w /x /y /z /braceleft /bar /braceright /asciitilde
/.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef
/.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef
/.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef
/.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /exclamdown /cent
/sterling /fraction /yen /florin /section /currency /quotesingle /quotedblleft
/guillemotleft /guilsinglleft /guilsinglright /fi /fl /.notdef /endash /dagger
/daggerdbl /periodcentered /.notdef /paragraph /bullet /quotesinglbase /quotedblbase
/quotedblright /guillemotright /ellipsis /perthousand /.notdef /questiondown
/.notdef /grave /acute /circumflex /tilde /macron /breve /dotaccent /dieresis
/.notdef /ring /cedilla /.notdef /hungarumlaut /ogonek /caron /emdash /.notdef
/.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /.notdef
/.notdef /.notdef /.notdef /.notdef /.notdef /.notdef /AE /.notdef /ordfeminine
/.notdef /.notdef /.notdef /.notdef /Lslash /Oslash /OE /ordmasculine /.notdef
/.notdef /.notdef /.notdef /.notdef /ae /.notdef /.notdef /.notdef /dotlessi
/.notdef /.notdef /lslash /oslash /oe /germandbls /.notdef /.notdef /.notdef
/.notdef ] def
```

Specifying encoding file makes possible to override built-in encoding of the font file being read. For example, TFM file of the following command line will be written in TEX EC encoding (assuming that the encoding file `ec.enc` represents it):

```
jfite -Rotf=myfont.otf -Renc=ec.enc -Wtfm
```

1.3 Adobe INF Font File

`-Rinf[=<input file name>]`

Purpose: read Adobe INF font file.

1.4 Ligature Files

```
-Rlig[=<input file name>]
-Wlig[=<output file name>]
```

Purpose: read/write ligature file.

Ligature file specifies font ligatures in PostScript notation. Here is an example of a ligature file content useful for generating TFM/PL files for TEX:

```
/SampleLigatures [
[/f /l /fl]
[/f /i /fi]
[/f /f /ff]
[/ff /l /ffl]
[/ff /i /ffi]
[/quoteright /quoteright /quotedblright]
[/quoteleft /quoteleft /quotedblleft]
[/hyphen /hyphen /endash]
[/endash /hyphen /emdash]
] def
```

1.5 OpenType/CFF Files

```
-Rotf[=<output file name> ]
```

Purpose: read OpenType/CFF font file.

1.6 Type1 Printer Font ASCII (PFA) Files

```
-Rpfa[=<input file name>]
```

Purpose: read PostScript Type1 ASCII font file.

1.7 Type1 Printer Font Binary (PFB) Files

```
-Rpfb[=<input file name>]
```

Purpose: read PostScript Type1 binary font file.

1.8 Printer Font Metric (PFM) Files

```
-Rpfm[=<input file name>]
```

Purpose: read/write Printer font metrics file.

1.9 TEX Property List (PL) Files

```
-Rpl[=<input file name>]
-Wpl[=<output file name>]
```

Purpose: read/write TEX font property list file.

1.10 TEX Font Metric (TFM) Files

```
-Rtfm[=<input file name>]
-Wtfm[=<output file name>]
```

Purpose: read/write TEX font metrics file.

1.11 TrueType Font (TTF) Files

```
-Rttf[=<input file name>]
```

Purpose: read TrueType font file.

1.12 TEX Virtual Font (VF) Files

```
-Rvf[=<input file name>]
-Wvf[=<output file name>]
```

Purpose: read/write TEX virtual font file.

1.13 TEX Virtual Property List (VPL) Files

```
-Rvpl[=<input file name>]
-Wvpl[=<output file name>]
```

Purpose: read/write TEX virtual font property list file.

While processing VF/VPL files, JustFontIt TE covers the functionality of vftovp and vptovf utilities by Prof. D. E. Knuth.

IMPORTANT NOTE:

The notation used in this section should not mislead the user to the opinion that read/write tasks must be used in pairs, that is after a read task for font format `<format>`, the respective

write task for that `<format>` must follow. Read/write tasks can be mixed. Also there can be multiple input/output tasks. For example, the following command line makes sense:

```
jfite -Rttf=myfont.ttf -Wtfm -Wpl
```

1.14 Forming Default File Names

To save you time in typing command line, JustFontIt TE allows to skip input/output file name in case when the file name can be derived from the file name of a previous task. For example, the following command line

```
jfite -Rafm=myfont.afm -Rinf -Wtfm
```

is a shortcut for

```
jfite -Rafm=myfont.afm -Rinf=myfont.inf -Wtfm=myfont.tfm
```

However, when forming default output file names, JustFontIt TE points the output stream to the current directory. This is done for security reasons to avoid overwriting files in the production folder. For example, the TFM file in following command line

```
jfite -Rpfb=/home/fonts/myfont.pfb -Wtfm
```

will be written in the current directory which is not necessarily `/home/fonts/`.

To override this behavior, use

```
-f
```

option. With this option output file names are formed basing on full path in the base input path.

If a file name contains spaces, the command line switch must be surrounded by double quotes, for example

```
jfite "-Rpfm=/home/Collection of Great Fonts/The Best Font Family/Just a Great Font.pfm" -Rpfb
-Wpl
```

1.15 Response Files

```
-@<file name>
```

Purpose: take the list of commands from `<file name>`. With this option it is possible to put the list of JustFontIt TE commands into a text file. This is useful if command lines are lengthy, or when it is not possible to describe the desired JustFontIt TE action by means of a single

command line. <file name> must be ASCII text file, each command should be placed on a new line.

1.16 Directory for Recursive Search

`-i<directory>`

Purpose: read files of the specified directory and its subdirectories.

This allows to specify a directory where JustFontIt TE will look for input files. With this option it makes sense to specify input file names by wildcards. For example, by means of the following command line it is possible to generate PL files for all Type1 fonts residing in `/home/font` and all its subdirectories:

```
jfite -i/home/fonts -Rpfa=*.pfa -Rpfb=*.pfb -Renc=ec.enc -Wpl -f
```

Notice that `-f` option tells JustFontIt TE to form full output file paths.

1.17 Options to Set Verbosity Level

`-c`

Purpose: run in concise mode (only most important warnings).

`-s`

Purpose: run in silent mode (no warnings).

`-b`

Purpose: run in batch mode (no messages).

1.18 Program Help

`-h`

or no command line switches.

Purpose: put out brief program help and exit.

Notice that JustFontIt TE command line switches are case-sensitive.

2 JustFontIt TE Messages

2.1 Task Processing Messages

`Info: Reading <file name>.`

By this message JustFontIt TE informs about the file being read.

`Info: Writing <file name>.`

By this message JustFontIt TE informs about the file being written.

`Info: <number> file(s) processed, <number> of them failed.`

By this message JustFontIt TE informs about the current session statistics.

`Error: Incorrect command line option <option>.`

This error appears when an incorrect command line options is specified. For example, JustFontIt TE would produce this error if a command line option

`-Rpf=myfont.pfb`

would be passed to it.

Troubleshooting: Fix the command line or response file.

`Error: Cannot start up. Why?..`

This error appears when JustFontIt TE fails to perform internal consistency tests at startup.

Troubleshooting: Please report the problem to JustFontIt TE support.

`Error: Incorrect command line option.`

This error indicates incorrect sequence of command line options, namely a default file name for a read or write task cannot be formed. For example, the following command line would trigger the error above:

`jfite -Rafm -Rpfb=myfont.pfb -Wtfm`

because input file name for the Read AFM task is unspecified.

Troubleshooting: Fix the command line or response file. For example, the command line above can be fixed as

```
jfite -Rpfb=myfont.pfb -Rafm -Wtfm

Error:  Incorrect name of response file.
```

The error appears when the `-@` switch was not followed by the response file name. For example, the following command line would cause the error above:

```
jfite -@ myres.txt
```

Troubleshooting: Fix the command line. For example, the command line above should be fixed as

```
jfite -@myres.txt

Error:  Incorrect name of recursive-search directory.
```

The error appears when the `-i` switch was not followed by the directory path. For example, the following command line would cause the error above:

```
jfite -i /home/myfonts -Rttf=*.ttf
```

Troubleshooting: Fix the command line. For example, the command line above should be fixed as

```
jfite -i/home/myfonts -Rttf=*.ttf

Error:  Cannot form input/output file name.
```

This error occurs when JustFontIt TE fails to form the output file name. For example, the following command line would trigger the error above:

```
jfite -Rpl=myfont.pl -Wtfm=myfont.pl
```

The reason is that an output file name cannot be same as any of the input file name. The logic is supposed to help loosing user's data by accident.

Troubleshooting: Fix the command line. For example, the command line above can be fixed as

```
jfite -Rpl=myfont.pl -Wtfm=myfont.tfm
```

Alternatively, specify the `-f` switch to override the safe logic.

`Error: Severe run-time fault occurred.`

This error may appear due to an unexpected run-time error.

Troubleshooting: Please report the error to JustFontIt support.

2.2 Generic Messages

`Error: Out of memory.`

This error indicates an internal fault of JustFontIt TE while allocating a memory block from heap. The error may indicate the lack of free memory in the system or a severe problem in the font file.

Troubleshooting: Try closing unneeded running applications and retry JustFontIt TE. If the problem recurs, please report it to JustFontIt support.

`Error: Range error.`

This error indicates an internal fault of JustFontIt TE. The error may occur due to unusual input data.

Troubleshooting: Please report the problem to JustFontIt support.

`Error: Overflow error.`

This error indicates an internal fault of JustFontIt TE while converting data representation. The error may occur due to unusual input data.

Troubleshooting: Please report the problem to JustFontIt support.

`Error: Internal fault in <source name> at line <number>.`

This error indicates an internal fault of JustFontIt TE.

Troubleshooting: Please report the problem to JustFontIt support.

`Error: Unable to open output file <file name>.`

This error occurs when JustFontIt TE fails to open a file stream for writing.

Troubleshooting: Make sure that the specified file can be opened for writing. For example, such a file cannot reside on CD/DVD read-only media, turned off file drives, any file locations access to which is forbidden for the user etc.

`Error: Unable to open input file <file name>.`

This error occurs when JustFontIt TE fails to open a file stream for reading.

Troubleshooting: Make sure that the specified file exists and access to it is allowed for the user etc.

`Error: Cannot save output data.`

This error occurs when JustFontIt TE fails to save data to the file stream.

Troubleshooting: Make sure that the file media has enough free space to save the data.

`Error: Cannot round <value> to integer.`

This error occurs when JustFontIt TE fails to cast a large floating point `<value>` to an integer. Usually such problem occurs when JustFontIt TE processes badly corrupted input data.

Troubleshooting: Try to inspect input data on corruption. If it does not help, please report the problem to JustFontIt support.

`Error: Cannot convert <source data type> <value> to '<target data type>'.`

This error occurs when JustFontIt TE fails to cast a `<value>` of `<source data type>` to another data type. Usually such problem occurs when JustFontIt TE processes badly corrupted input data.

Troubleshooting: Try to inspect input data on corruption. If it does not help, please report the problem to JustFontIt support.

2.3 Generic Text Parser Messages

`Error: Key requires a value.`

This error is raised when the generic text parser fails to fetch the token required syntactically in the current context.

`Error: Delimiter expected.`

This error is raised when the generic text parser fails to fetch the delimiter required syntactically in the current context.

2.4 Input and Output Messages

`Error: Unable to open file <file name>, mode <mode name>.`

This error occurs when the <file name> cannot be opened in the `<mode name>` (C notation).

Troubleshooting: Make sure that the specified file can be opened with `<mode name>`. Ask for JustFontIt support assistance if you are in a quandary.

`Error: While opening file <file name>, invalid access mode %s detected.`

This error occurs when JustFontIt TE attempts to open a file in an impossible mode. This may be internal problem of JustFontIt TE.

Troubleshooting: Please report the error to JustFontIt support.

`Error: Bad file pointer and illegal seek on device.`

This error may appear on JustFontIt TE attempt to perform file-pointer seek operations on a file stream. Also this error may indicate an internal problem in JustFontIt TE.

Troubleshooting: Do not pass stream incapable for file seek operations (like stdin) as JustFontIt TE file name. If this is not a case, please report the problem to JustFontIt support.

`Error: Empty input file <file name>.`

This error means that JustFontIt TE encountered an empty (zero size) in a context when it was not allowed.

Troubleshooting: Make sure that an input file passed to JustFontIt TE is not empty.

`Error: Cannot read file <file name>.`

The error means that JustFontIt TE failed to get read access rights for the `<file name>`.

Troubleshooting: Make sure that the specified file can be opened for reading. For example, such a file cannot reside in a location access to which is forbidden for the user etc.

`Error: Cannot write file <file name>.`

The error means that JustFontIt TE failed to get write access rights for the `<file name>.`

Troubleshooting: Make sure that the specified file can be opened for writing. For example, such a file cannot reside on CD/DVD read-only media, turned off file drives, any file locations access to which is forbidden for the user etc.

2.5 AFM File Related Messages

`Error: Key requires a value.`

JustFontIt TE failed to resolve the current key-value pair. The AFM file parsing terminated.

Troubleshooting: Open the AFM file in an ASCII file editor, locate the line where the error occurred and fix it.

`Error: Unexpected end of file.`

JustFontIt TE encountered premature end of file while parsing the AFM file. This error indicates a severe problem in the AFM file.

`Error: Invalid numeric value for <keyword>.`

JustFontIt TE failed to evaluate the numeric value for the `<keyword>`. The value for the keyword is not set. This problem may indicate corrupted or incorrectly saved AFM file. Unlike similar warning, this error appears for a keyword which a essential for AFM file parsing.

Troubleshooting: Open the AFM file in an ASCII file editor, locate the problem keyword and enter correct numeric value for it.

`Error: Invalid number of characters (<number>).`

This occurs when the number of declared characters in the CharMetrics section is negative. JustFontIt TE terminated further file parsing.

Troubleshooting: Open the AFM file in an ASCII file editor, locate the CharMetrics section and enter correct numeric value for characters.

`Error: Not an AFM file.`

JustFontIt TE failed to recognize the file being parsed as an AFM file.

Troubleshooting: Make sure that the file being entered is actually an AFM.

`Error: Metric files for CID-keyed fonts are not supported.`

JustFontIt TE does not support AFM for CID-keyed fonts.

`Error: Metric files for composite fonts are not supported.`

JustFontIt TE does not support AFM for composite fonts fonts.

`Error: Incorrect AFM file.`

This error appears when the input AFM has been truncated somewhere before StartCharMetrics or inside EndCharMetrics section. Probably, such an AFM is corrupted beyond repair.

`Error: Invalid numeric value for character position.`

While evaluating the character position JustFontIt TE failed to convert the position from either decimal or hexadecimal notation to a number. Further file parsing terminated.

Troubleshooting: Open the AFM file in an ASCII file editor, locate the problem character entry location and enter correct numeric value for character position.

2.6 Encoding File Related Messages

`Error: Invalid empty glyph name.`

The encoding file being processed contains an entry with empty glyph name.

2.7 INF File Messages

`Error: Values of FontName entry mismatch in font data and INF files.`

This warning appears when JustFontIt TE compares data gathered from font file and the INF file. It means that the font file and the INF file do not correspond to each other. If they are supposed to belong to same font data, for example, when JustFontIt TE was invoked as

```
jfite -Rpfb=myfont.pfb -Rinf
```

this error indicates a problem in the INF file.

Troubleshooting: Open the INF file in an ASCII file editor, locate the FontName entry and fix it in accordance with the font file.

2.8 Type1 Related Messages

`Error: FontDirectory not found.`

This error appears when after completing interpreting the Type1 font as a generic PostScript program, a FontDirectory dictionary as required attribute of a Type1 font was not fetched. This error means a severe problem in the Type1 font.

Troubleshooting: Make sure that the PostScript data really represents a Type1 font.

`Error: Font interpretation failed.`

This error occurs when JustFontIt TE fails to create the internal input stream for the Type1 font.

Troubleshooting: Make sure that the PostScript data really represents a Type1 font.

`Error: FontDirectory empty.`

This error appears when after completing interpreting the Type1 font as a generic PostScript program, a FontDirectory dictionary as required attribute of a Type1 font is empty. This error means a severe problem in the Type1 font.

`Error: Font contains no glyphs.`

This error appears when after completing interpreting the Type1 font as a generic PostScript program, JustFontIt TE detected that the font contains no descriptions for glyphs. Such Type1 font is meaningless.

`Error: /.notdef not found.`

This error occurs when the Type1 font contains no description for the required `/.notdef`. Type1 font is not operable when this glyph is missing. The error indicates a bug in certain versions of popular font creation software.

Troubleshooting: To fix the problem, disassemble the font file by means of JustFontIt[1]:

```
jfi -Rpfb=myfont.pfb -Wpsf
```

Open the resulting myfont.psf in an ASCII text editor. Locate the end of CharStrings dictionary in the end of myfont.psf. It may look like

```
 -11 -42 rlineto
 -4 -15 7 -12 12 0 rrcurveto
 13 0 13 12 4 15 rrcurveto
 12 42 rlineto
 4 15 14 12 14 0 rrcurveto
 14 0 15 13 4 15 rrcurveto
 closepath
 endchar
 } ND

end end
readonly put put
```

[1] JustFontIt is separate commercial program.

```
dup/FontName get exch definefont pop
mark currentfile closefile
%%T1Disasm Trailer
```

etc

Add the missing `/.notdef` entry

```
/.notdef { 0 1000 hsbw
 endchar
 } ND
```

before `end end` operators. Save `myfont.psf` and assemble it to PFB by

```
jfi -Rpsf=myfont.psf -Wpfb
```

`Error: Cannot open interpreter.`

This error indicates an internal fault of JustFontIt TE while allocating a memory for PostScript interpreter. The error may indicate the lack of free memory in the system or a severe problem in the font file.

Troubleshooting: Try closing unneeded running applications and retry JustFontIt TE. If the problem recurs, please report it to JustFontIt support.

`Error: /limicheck: <source>.`

This error occurs when `<source>` exceeds its implementation limit. Please see PostScript reference for more details. Appearing such an error indicates a severe corruption of the Type1 font.

Troubleshooting: Try to resave the font by means of a newer version of font creation software.

`Error: Internal: <error details>.`

This error indicates an internal problem of JustFontIt TE PostScript interpreter. It may indicate a bug of either font creation software or JustFontIt TE itself.

Troubleshooting: Try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to JustFontIt support.

`Error: Memory allocation error.`

This error occurs when JustFontIt TE fails to allocate memory from system heap. The error may indicate the lack of free memory in the system.

Troubleshooting: Try closing unneeded running applications and retry JustFontIt TE. If the problem recurs, please report it to JustFontIt support.

`Error: <stack name> stack overflow.`

This error shows `<stack name>` overflow. The problem may indicate a severe problem in the Type1 font.

Troubleshooting: Try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software.

`Error: %s stack underflow.`

This shows `<stack name>` underflow. The problem may indicate a severe problem in the Type1 font.

Troubleshooting: Try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

`Error: Type check error (<context>).`

The error is triggered by the type check error in the `<context>`. This is a severe error of the Type1 font.

Troubleshooting: Try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

`Error: Unmatched mark error (%s).`

The error is triggered by the unmatched mark error in the `<context>`. This is a severe error of the Type1 font.

Troubleshooting: Try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

`Error: Syntax error: <error details>.`

The syntax error is a severe error of the Type1 font.

Troubleshooting: Try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

`Error: Bad Type 1 binary file: <error details>.`

This error indicates a structural problem in the Type1 font.

Troubleshooting: Back up the Type1 font file. Disassemble and then assemble it back by means of JustFontIt. If it does not help, try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

`Error: Dictionary stack underflow.`

This error indicates a severe problem in the Type1 font.

Troubleshooting: Try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

`Error: Dictionary stack overflow.`

This error indicates a severe problem in the Type1 font.

Troubleshooting: Try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

`Error: Range check error in <context>.`

This error indicates a check error in the `<context>`. This is a severe error of the Type1 font.

Troubleshooting: Try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

`Error: Unknown control '<control name>'.`

This error often appears in corrupted (binary modified) Type1 font file.

Troubleshooting: Disassemble the Type1 font by means of JustFontIt `-Wpsf` command. Open the disassembled font in an ASCII text editor. Look through the disassembled data. Locate a point when disassembled PostScript commands add, and a binary data begins. If this point resides inside CharStrings dictionary, there is a chance to partially recover the font. For example, your PSF file may look like

```
-67 -164 rlineto
-52 -64 rlineto
58 166 rlineto
closepath
```

```
 endchar
 } ND
/one { 176 1050 hsbw
 289 -44 vstem
<binary mumbo-jumbo>
```

Erase all data in the file from the last ND command to the end of the file.
Append a reasonable ending of the file, i.e.

```
end end
readonly put put
dup/FontName get exch definefont pop
mark currentfile closefile
%%T1Disasm Trailer

0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
cleartomark{restore}if
```

Assemble the PSF back to Type1 font. Recheck the result by disassembling it and watching to JustFontIt messages.

If it does not help, try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

`Error: Operator '<operator name>' not implemented.`

This error occurs when the PostScript interpreter built in JustFontIt TE hits a not implemented PostScript operator. It is known that certain Multiple Master Type1 font developed by Adobe contain unusual PostScript operators like `save` or `dtransform`. JustFontIt TE does not support such Type1 font programs.

`Error: Bad charstring procedure: <details>.`

This error occurs in an incorrect charstring procedure. This is a severe Type1 font problem.

Troubleshooting: Try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

`Error: Zero divide error.`

This error may be triggered by an attempt to divide by zero in PostScript `div` operator. This is a severe problem in Type1 font.

Troubleshooting: Try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

`Error: Bad Type 1 Font Program: <details>.`

This error occurs when the PostScript program fails to comply with the necessary condition identifying it as a Type1 program. As usual, most errors of this group reveal a serious problem in the Type1 font.

Troubleshooting: If you are a PostScript guru, disassemble the Type1 font and fix the respective PostScript coding problem. If not, resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

`Error: type1 charstring: <details>.`

This error occurs in an incorrect charstring. This is a severe Type1 font problem.

Troubleshooting: Try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

`Error: Invalid seac reference to missing /<glyph name>.`

This error appears when the seac reference to a glyph missing in the font program. While rendering such a seac command on actual PostScript device, a PostScript error will occur. The source of the problem is a bug in certain version of a popular Type1 creation software.

Troubleshooting: Disassemble the Type1 font file. Look through seac commands, locating the problem seac. For example, you may found that the seac command like

```
/igrave { 95 600 hsbw
 151 26 0 245 193 seac
 } |-
```

references not existent glyph `/dotlessi`. If so, erase all `seac` commands which make use of `/dotlessi`. Disassemble the font, and test it.

In case of problems, try to resave the font by means of a newer version of font creation software. If it does not help, report the problem to the font creation software vendor.

2.9 PFM Related Messages

`Error: Incorrect Adobe Standard encoded font - no glyphs in valid range.`

This error may occur while generating PFM file. It means that the PFM file being generated contains no glyphs in the range 32–255 what is valid for Adobe Standard encoded. In other words, no glyphs will be displayed in such a font.

Troubleshooting: If you must create a font which contains glyphs in 0–31 range, use Symbol character set.

`Error: Inconsistent data types.`

This error appears when JustFontIt TE fails to pass internal tests of PFM data parsing validity.

Troubleshooting: Please report this error to JustFontIt support.

`Error: Font menu name cannot exceed 31 characters.`

This error may occur while generating PFM file. It means that the font data used to generate the PFM file contains too long (exceeding 31 characters) menu name for the name by which Windows identifies font family.

Troubleshooting: Fix MSMenuName in the INF file used generate the PFM file.

`Error: Font name cannot exceed 31 characters.`

This error may occur while generating PFM file. The problem is that the font data contains too long (exceeding 31 characters) font name. Such a PFM file is not valid for Windows.

Troubleshooting: Regenerate the PFM file with correct value of font name. Notice that the font name in a PFM file must be same as FontName entry in the corresponding PFB file.

`Error: PostScript or Windows font name unknown.`

This error may occur while generating PFM file. The font data used to generate the PFM file do not contain FontName or MSMenuName and FamilyName.

Troubleshooting: Use correct font data to generate the PFM file. If the font data was produced by reading AFM and INF file, correct them to specify the missing properties.

`Error: Cannot read the <name> PFM section.`

The error may happen while reading input PFM file. It means that JustFontIt TE failed to read the `<name>` section from the PFM file. The error indicates a severe structural problem in

the PFM file.

Troubleshooting: Regenerate the PFM file.

`Error: Invalid PFM version <hex number>.`

The error may happen while reading input PFM file. JustFontIt TE encountered unsupported version of the PFM file. The problem may indicate that the PFM file is binary corrupted.

Troubleshooting: Regenerate the PFM file.

`Error: Invalid DeviceName <name> (must be 'PostScript').`

The error may happen while reading input PFM file. Any PFM file must have hardwired device name "PostScript".

Troubleshooting: Regenerate the PFM file.

`Error: Mismatch number of kern pairs in ETM (<number>) and KERNPAIR (<number>) sections.`

The error may occur while reading input PFM file. Its meaning is that the PFM file inconsistently specifies the number of kern pairs it contains. Such a problem may indicate a bug in software used to generate the PFM file.

Troubleshooting: Regenerate the PFM file.

`Error: Unexpected end of file while reading a zero-terminated buffer.`

The error may occur while reading input PFM file. It means that JustFontIt TE failed to complete reading of a zero terminated string residing in a PFM section. The error indicates a severe structural problem in the PFM file

Troubleshooting: Regenerate the PFM file.

`Error: LastChar <number> is smaller than FirstChar <number> (PFMHEADER).`

The error may occur while reading input PFM file. The PFM file specifies contradictious values for the first and last character of the PFM encoding. The problem may indicate a bug in software used to generate the PFM file.

Troubleshooting: Regenerate the PFM file.

2.10 TFM and PL Related Messages

JustFontIt TE displays all warning and error self-explaining messages as if it were tftopl or pltotf utilities designed by Prof. D. E. Knuth.

2.11 VF and VPL Related Messages

JustFontIt TE displays all warning and error self-explaining messages as if it were tftopl or pltotf utilities design by Prof. D. E. Knuth.

2.12 Ligature File Related Messages

`Error: Ligature dictionary entry must contain 3 names.`

This error occurs in an ill-formed ligature file. Each entry of a ligature file must contain exactly three names like

```
[/f /l /fl]
```

Troubleshooting: Open the ligature file in an ASCII file editor and fix the problem entry.

`Error: Ligatures not found.`

This error occurs in an ill-formed ligature file. Ligature file must contain at least one ligature definition.

Troubleshooting: Open the ligature file in an ASCII file editor and insert a ligature entry.

2.13 OpenType Related Messages

`Error: Not a TrueType font.`

This error appears when JustFontIt TE failed to identify the input file as TrueType.

Troubleshooting: Make that the input file is TrueType.

`Error: Not an OpenType/CFF font.`

This error appears when JustFontIt TE failed to identify the input file as OpenType/CFF.

Troubleshooting: Make that the input file is OpenType/CFF.

`Error: Required TrueType table '<table name>' not found.`

This error appears when JustFontIt TE failed to locate the <table name>, required for further file processing. An OpenType font which misses this table does not meet the OpenType font format specification.

Error: Cannot read <number> bytes from the '<file name>' font.

This occurs when JustFontIt TE fails to read <number> bytes from the <file name> font. This error indicates that the font is badly damaged, and attempt to render it may cause software faults, including possible system crash. Avoid using such font.

Error: "loca" mapping for glyph <num> points outside of the glyph range.

This error occurs when the entry <num> in the loca table that points outside the range(s) of glyf data. This may happen if the loca table has not been updated to reflect the removal of the final glyph(s) from a font. This error indicates that the font is damaged, and attempt to render it may cause software faults.

Error: Unique font ID not found.

This error occurs when an Unique font ID is not found in the font. An OpenType font which misses Unique font ID does not meet the OpenType font format specification. Most probably, such a font will be rejected by the operating system rasterizer.

Error: Unique font ID for PlatformID=<num>, EncodingID=<num>, LanguageID=<num> is empty.

This error occurs when an Unique font ID is empty for for PlatformID=<num>, EncodingID=<num>, LanguageID=<num>. An OpenType font with an empty Unique font ID does not meet the OpenType font format specification.

2.14 TrueType Related Messages

Error: Font contains no glyphs.

This error appears when the font contains no glyphs.

3 Support and Contacts

If you encounter any problems with JustFontIt TE that you cannot resolve yourself, please email to `support@justfontit.com`. By regular mail, please write to

Dr. Alexei V. Kostin
ul. Zaytseva, 34–34.
Saint Petersburg, 198188
Russia

4 Copyright and Trademarks