



Introduction



tfm



tfm-validate



TeX Live



Conclusion

# TeX Font Metrics Validation: Application to TeX Live

~ TUG 2024 ~

Didier Verna  
EPITA / LRE

didier@lrde.epita.fr



[lrde/~didier](#)



[@didierverna](#)



[didier.verna](#)



[in/didierverna](#)

# Conclusion

Jump to: # 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

## TeX Live TFM Compliance Reports

Compliance reports for TFM files in the TeX Live distribution

TeX Live Master/texmf-dist/, 79016 fonts inspected, 2983 fonts skipped (OFM / JFM data; not supported yet).  
770 non-compliant fonts, 3 types of warning, 1 type of error found.

Reports generated on Wed Jul 10 13:37:32 2024 GMT+0 with [TFM Validate](#) 1.1 "QT Cloistered Monk" / [TFM](#) 1.3 "Eiad".

Non Compliant Fonts Issues

- ▶ TFM font validator for T<sub>E</sub>X Live: <https://texlive.info/tfm-validate/>
- ▶ Updated daily
- ▶ Currently finding 770 fonts with oddities *w.r.t.* the standard

# The tfm Library

- ▶ Part of ETAP
- ▶ Bring T<sub>E</sub>X Font Metrics information to Common Lisp

# The tfm Library

- ▶ Part of ETAP
- ▶ Bring T<sub>E</sub>X Font Metrics information to Common Lisp

## Entry Point

```
CL-USER> (tfm:load-font ".../cmr10.tfm")
#<TFM:FONT cmr10>
CL-USER>
```

## The tfm Library

- ▶ Part of ETAP
- ▶ Bring T<sub>E</sub>X Font Metrics information to Common Lisp

### Entry Point

```
CL-USER> (tfm:load-font ".../cmr10.tfm")
#<TFM:FONT cmr10>
CL-USER>
```

- ▶ Design goals: robustness & flexibility



# Robustness



Anticipate *all* potential problems to avoid crashes or erratic behavior.

# Robustness

Anticipate *all* potential problems to avoid crashes or erratic behavior.

- ▶ 20 potential problems identified, various degrees of severity

# Robustness


Anticipate *all* potential problems to avoid crashes or erratic behavior.

- ▶ 20 potential problems identified, various degrees of severity
- ▶ Examples





# Robustness

Anticipate *all* potential problems to avoid crashes or erratic behavior.

- ▶ 20 potential problems identified, various degrees of severity
- ▶ Examples
  - ▶  Truncated file / invalid section pointers → hopeless




# Robustness

Anticipate *all* potential problems to avoid crashes or erratic behavior.

- ▶ 20 potential problems identified, various degrees of severity
- ▶ Examples
  - ▶  Truncated file / invalid section pointers → hopeless
  - ▶  Cycle in a ligature program → some hope




# Robustness

Anticipate *all* potential problems to avoid crashes or erratic behavior.

- ▶ 20 potential problems identified, various degrees of severity
- ▶ Examples
  - ▶  Truncated file / invalid section pointers → hopeless
  - ▶  Cycle in a ligature program → some hope
  - ▶  Comment string inconsistency → benign

# Robustness

Anticipate *all* potential problems to avoid crashes or erratic behavior.

- ▶ 20 potential problems identified, various degrees of severity
- ▶ Examples
  - ▶  Truncated file / invalid section pointers → hopeless
  - ▶  Cycle in a ligature program → some hope
  - ▶  Comment string inconsistency → benign
- ▶ Context-dependent severity
  - ▶ *E.g.* (overridden) design size

# Flexibility

When possible, provide recovery options instead of just giving up.

# Flexibility

When possible, provide recovery options instead of just giving up.

- ▶ Currently a dozen

# Flexibility

When possible, provide recovery options instead of just giving up.

- ▶ Currently a dozen
- ▶ Examples
  - ▶ Discarding a bogus kerning / ligature program
  - ▶ Fixing `<table>[0]` entries (widths, heights, depths, italic corrections)
  - ▶ ...

## Flexibility

When possible, provide recovery options instead of just giving up.

- ▶ Currently a dozen
- ▶ Examples
  - ▶ Discarding a bogus kerning / ligature program
  - ▶ Fixing `<table>[0]` entries (widths, heights, depths, italic corrections)
  - ▶ ...
- ▶ Note: recovery is black magic (we can't be sure to fix the *real* problem)



# Flexibility

When possible, provide recovery options instead of just giving up.

- ▶ Currently a dozen
- ▶ Examples
  - ▶ Discarding a bogus kerning / ligature program
  - ▶ Fixing `<table>[0]` entries (widths, heights, depths, italic corrections)
  - ▶ ...
- ▶ Note: recovery is black magic (we can't be sure to fix the *real* problem)
  - ▶ Still, handy in interactive use

# Interactive Use vs. Production

## Example

```
CL-USER> (tfm:load-font "/tmp/cmr10.tfm")
```

While reading /tmp/cmr10.tfm,

while reading the character encoding scheme string,

padded string "TeX (ex)" is not in BCPL format.

See §10 of the TFtoPL documentation for more information.

```
[Condition of type NET.DIDIERVERNA.TFM:INVALID-PADDED-STRING]
```

Restarts:

0: [KEEP-STRING] Keep it anyway.

1: [FIX-STRING] Fix it using /'s and ?'s.

2: [DISCARD-STRING] Discard it.

3: [CANCEL-LOADING] Cancel loading this font.

--more--

# Flexibility

When possible, provide recovery options instead of just giving up.

- ▶ Currently a dozen
- ▶ Examples
  - ▶ Discarding a bogus kerning / ligature program
  - ▶ Fixing `<table>[0]` entries (widths, heights, depths, italic corrections)
  - ▶ ...
- ▶ Note: recovery is black magic (we can't be sure to fix the *real* problem)
  - ▶ Still, handy in interactive use

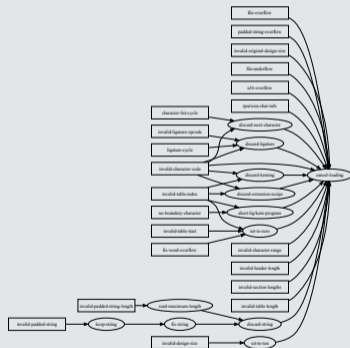
# Flexibility

When possible, provide recovery options instead of just giving up.

- ▶ Currently a dozen
- ▶ Examples
  - ▶ Discarding a bogus kerning / ligature program
  - ▶ Fixing `<table>[0]` entries (widths, heights, depths, italic corrections)
  - ▶ ...
- ▶ Note: recovery is black magic (we can't be sure to fix the *real* problem)
  - ▶ Still, handy in interactive use
- ▶ The whole point of tfm: no decision, only options

# tfm's Error Ontology

For the fun of it...



Introduction

tfm

tfm-validate

T<sub>E</sub>X Live

Conclusion

# The tfm-validate Library

# The tfm-validate Library

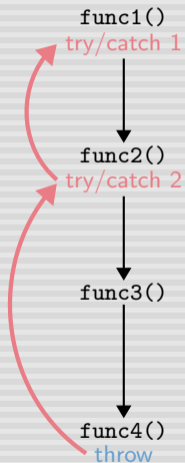
- ▶ Motivation
  - ▶ Curiosity rather than necessity
  - ▶ Easy to do

# The tfm-validate Library

- ▶ Motivation
  - ▶ Curiosity rather than necessity
  - ▶ Easy to do
- ▶ Key ingredient: the Common Lisp “condition system”

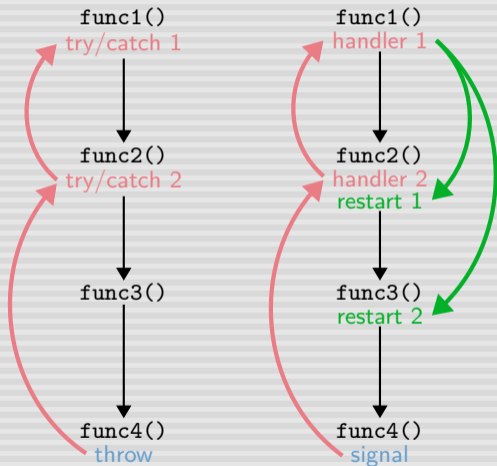


# Exception Handling



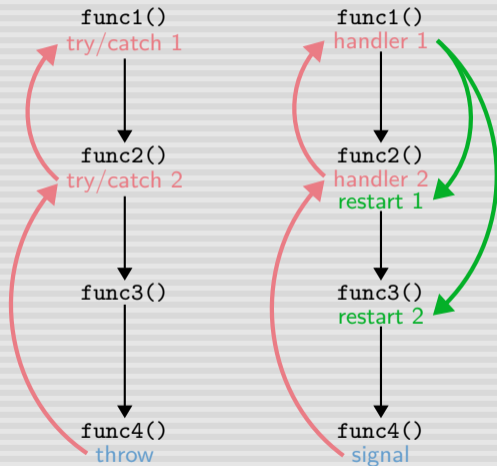
- ▶ Classic try/catch/throw
  - ▶ 2D SoC: handlers catch & resume

# Exception Handling



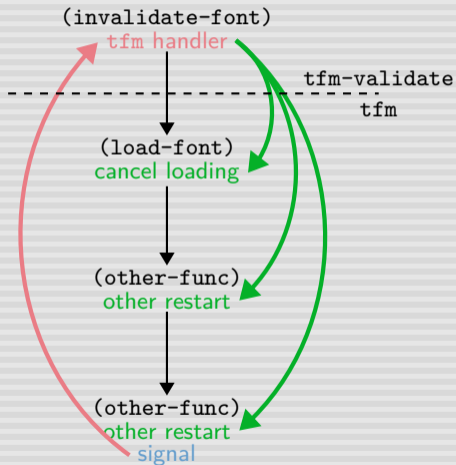
- ▶ Classic try/catch/throw
  - ▶ 2D SoC: handlers catch & resume
- ▶ Common Lisp condition system
  - ▶ 3D SoC: signal / handle / restart

# Exception Handling



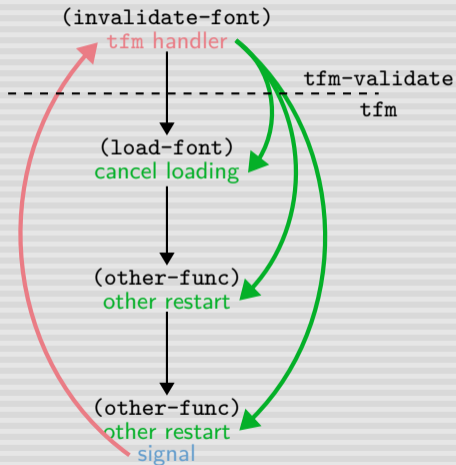
- ▶ Classic try/catch/throw
  - ▶ 2D SoC: handlers catch & resume
- ▶ Common Lisp condition system
  - ▶ 3D SoC: signal / handle / restart
- ▶ OO exception management
  - ▶ Hierarchy of conditions classes
  - ▶ First-class conditions (instances)

# Application: tfm-validate



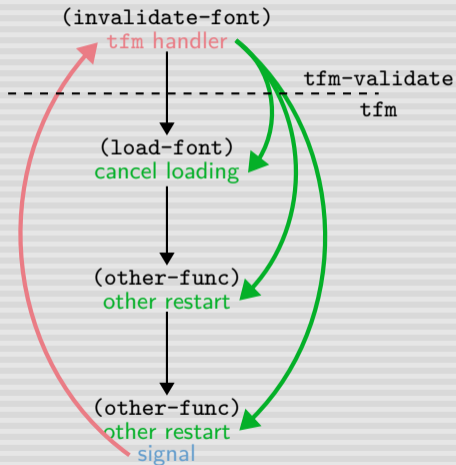
- ▶ Just a (big) handler around `tfm:load-font`
- ▶ Entry point: `invalidate-font`

# Application: tfm-validate



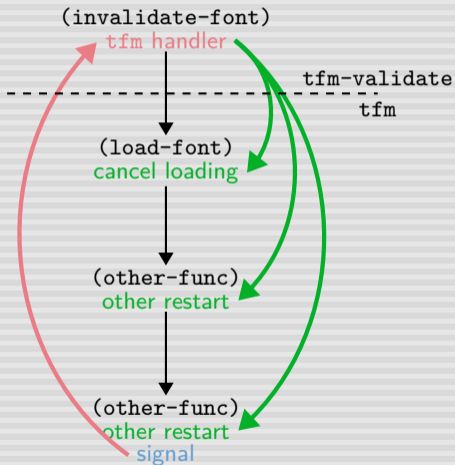
- ▶ Just a (big) handler around `tfm:load-font`
- ▶ Entry point: `invalidate-font`
  - ▶ Catch all `tfm` conditions
  - ▶ Select the “softest” restart

# Application: tfm-validate



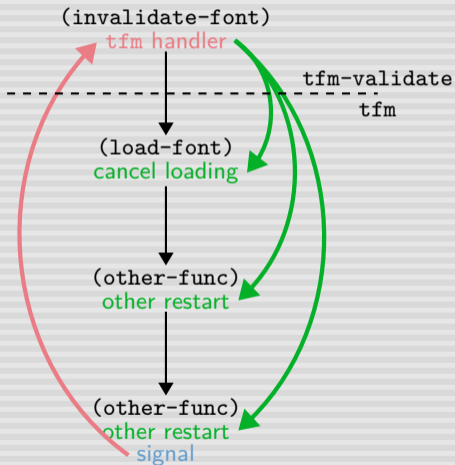
- ▶ Just a (big) handler around `tfm:load-font`
- ▶ Entry point: `invalidate-font`
  - ▶ Catch all `tfm` conditions
  - ▶ Select the “softest” restart
  - ▶ Collect the conditons before restarting
  - ▶ Return them at the end

# Application: tfm-validate



- ▶ Just a (big) handler around `tfm:load-font`
- ▶ Entry point: `invalidate-font`
  - ▶ Catch all `tfm` conditions
  - ▶ Select the “softest” restart
  - ▶ Collect the conditons before restarting
  - ▶ Return them at the end
- ▶ Convenience 1: `invalidate-directory`
  - ▶ Surprisingly high number of issues

# Application: tfm-validate



- ▶ Just a (big) handler around `tfm:load-font`
- ▶ Entry point: `invalidate-font`
  - ▶ Catch all `tfm` conditions
  - ▶ Select the “softest” restart
  - ▶ Collect the conditons before restarting
  - ▶ Return them at the end
- ▶ Convenience 1: `invalidate-directory`
  - ▶ Surprisingly high number of issues
- ▶ Convenience 2: `invalidate-texlive`
  - ▶ Issues in human-readable form (website)



# T<sub>E</sub>X Live Validation

## ▶ Current state

- ▶ 79016 fonts inspected
- ▶ 2983 fonts skipped (OFM & JFM)
- ▶ 770 fonts with non compliance issues

# T<sub>E</sub>X Live Validation

- ▶ Current state
  - ▶ 79016 fonts inspected
  - ▶ 2983 fonts skipped (OFM & JFM)
  - ▶ 770 fonts with non compliance issues
- ▶ But...
  - ▶ Only 4 different issues found
  - ▶ 3 warnings, 1 error

## Warning: File Overflow

- ▶ Actual size > declared size

### cmcb10 Compliance Report

[Return to toplevel.](#)

File: public/cmcyr/cmcb10.tfm

1 warning, 0 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate](#) 1.1 "QT Cloistered Monk" / [TFM](#) 1.3 "Eiad".

### Warnings

#### File Overflow

Declared file size 1308 is less than actual 1536.  
See §8 of the T<sub>F</sub>toPL documentation for more information.

## Warning: File Overflow

- ▶ Actual size > declared size
- ▶ 628 fonts affected

### cmcb10 Compliance Report

[Return to toplevel.](#)

File: public/cmcyr/cmcb10.tfm

1 warning, 0 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate](#) 1.1 "QT Cloistered Monk" / [TFM](#) 1.3 "Eiad".

### Warnings

#### File Overflow

Declared file size 1308 is less than actual 1536.  
See §8 of the TFMtoPL documentation for more information.

## Warning: File Overflow

- ▶ Actual size > declared size
- ▶ 628 fonts affected
- ▶ Only a warning
  - ▶ The fonts seem functional
  - ▶ `tftopl` ignores the junk

### cmcb10 Compliance Report

[Return to toplevel.](#)

File: public/cmcyr/cmcb10.tfm

1 warning, 0 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate](#) 1.1 "QT Cloistered Monk" / [TFM](#) 1.3 "Eiad".

### Warnings

#### File Overflow

Declared file size 1308 is less than actual 1536.  
See §8 of the TFM documentation for more information.

## Warning: File Overflow

- ▶ Actual size > declared size
- ▶ 628 fonts affected
- ▶ Only a warning
  - ▶ The fonts seem functional
  - ▶ `tftopl` ignores the junk
- ▶ The junk doesn't seem meaningful  
*Padding?*

### cmcb10 Compliance Report

[Return to toplevel.](#)

File: public/cmcyr/cmcb10.tfm

1 warning, 0 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate](#) 1.1 "QT Cloistered Monk" / [TFM](#) 1.3 "Eiad".

### Warnings

#### File Overflow

Declared file size 1308 is less than actual 1536.  
See §8 of the TFM documentation for more information.

# Warning: Padded String Overflow

▶ BCPL string tail  $\neq 0$

## pbkd8y Compliance Report

[Return to toplevel.](#)

File: adobe/ly1/pbkd8y.tfm

1 warning, 0 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate](#) 1.1 "QT Cloistered Monk" / [TFM](#) 1.3 "Eiad".

## Warnings

### Padded String Overflow

While reading the character encoding scheme string, padded string contains non-null overflow characters ("Y&Y Inc"). This font may have been created before April 1983. See §87 of the PLtoTF documentation, or "TeX Font Metrics Files" (David Fuchs, TUGBoat, Volume 2, #1) for more information.

## Warning: Padded String Overflow

- ▶ BCPL string tail  $\neq 0$
- ▶ 74 fonts affected

### pbkd8y Compliance Report

[Return to toplevel.](#)

File: adobe/ly1/pbkd8y.tfm

1 warning, 0 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate](#) 1.1 "QT Cloistered Monk" / [TFM](#) 1.3 "Eiad".

### Warnings

#### Padded String Overflow

While reading the character encoding scheme string, padded string contains non-null overflow characters ("Y&Y Inc"). This font may have been created before April 1983. See §87 of the PLtoTF documentation, or "TeX Font Metrics Files" (David Fuchs, TUGBoat, Volume 2, #1) for more information.



## Warning: Padded String Overflow

- ▶ BCPL string tail  $\neq 0$
- ▶ 74 fonts affected
- ▶ Only a warning
  - ▶  $\text{pltotf} \geq 1.3$
  - ▶ Strings purely informational

### pbkd8y Compliance Report

[Return to toplevel.](#)

File: adobe/ly1/pbkd8y.tfm

1 warning, 0 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate](#) 1.1 "QT Cloistered Monk" / [TFM](#) 1.3 "Eiad".

### Warnings

#### Padded String Overflow

While reading the character encoding scheme string, padded string contains non-null overflow characters ("Y&Y Inc"). This font may have been created before April 1983. See §87 of the PLtoTF documentation, or "TeX Font Metrics Files" (David Fuchs, TUGBoat, Volume 2, #1) for more information.

## Warning: Padded String Overflow

- ▶ BCPL string tail  $\neq 0$
- ▶ 74 fonts affected
- ▶ Only a warning
  - ▶ `pltotf`  $\geq 1.3$
  - ▶ Strings purely informational
- ▶ The junk can be meaningful

### pbkd8y Compliance Report

[Return to toplevel.](#)

File: adobe/ly1/pbkd8y.tfm

1 warning, 0 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate](#) 1.1 "QT Cloistered Monk" / [TFM](#) 1.3 "Eiad".

### Warnings

#### Padded String Overflow

While reading the character encoding scheme string, padded string contains non-null overflow characters ("Y&Y Inc"). This font may have been created before April 1983. See §87 of the PLtoTF documentation, or "TeX Font Metrics Files" (David Fuchs, TUGBoat, Volume 2, #1) for more information.

# Warning: Spurious Char Info

- ▶ Absent character  $\neq 0$

## fcmb7t Compliance Report

Return to [toplevel](#).

File: public/cm-lgc/fcmb7t.tfm

1 warning, 0 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate](#) 1.1 "DT Cloistered Monk" / [TFM](#) 1.3 "Elaïd".

## Warnings

### Spurious Char Info

While parsing char info table at position 32/170, for character code 32, char-info structure for a non-existent character is not blank

```
#S(CHAR-INFO
:WIDTH-INDEX 0
:HEIGHT-INDEX 0
:DEPTH-INDEX 0
:ITALIC-INDEX 0
:LIG/KERN-INDEX 58
:NEXT-CHAR NIL
:EXTEN-INDEX NIL).
```

See §11 of the T<sub>F</sub>toPL documentation for more information.

# Warning: Spurious Char Info

- ▶ Absent character  $\neq 0$
- ▶ 66 fonts affected

## fcmb7t Compliance Report

Return to [toplevel](#).

File: public/cm-lgc/fcmb7t.tfm

1 warning, 0 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate](#) 1.1 "DT Cloistered Monk" / [TFM](#) 1.3 "Elaïd".

## Warnings

### Spurious Char Info

While parsing char info table at position 32/170, for character code 32, char-info structure for a non-existent character is not blank

```
#S(CHAR-INFO
:WIDTH-INDEX 0
:HEIGHT-INDEX 0
:DEPTH-INDEX 0
:ITALIC-INDEX 0
:LIG/KERN-INDEX 58
:NEXT-CHAR NIL
:EXTEN-INDEX NIL).
```

See §11 of the T<sub>F</sub>to<sub>P</sub>L documentation for more information.

## Warning: Spurious Char Info

- ▶ Absent character  $\neq 0$
- ▶ 66 fonts affected
- ▶ Only a warning
  - ▶ `pltotf` zeroes out those chars
  - ▶ `tftopl` ignores them

### fcmb7t Compliance Report

Return to [toplevel](#).

File: public/cm-lgc/fcmb7t.tfm

1 warning, 0 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate](#) 1.1 "OT Cloistered Monk" / [TFM](#) 1.3 "Elaïd".

### Warnings

#### Spurious Char Info

While parsing char info table at position 32/170, for character code 32, char-info structure for a non-existent character is not blank

```
#S(CHAR-INFO
:WIDTH-INDEX 0
:HEIGHT-INDEX 0
:DEPTH-INDEX 0
:ITALIC-INDEX 0
:LIG/KERN-INDEX 58
:NEXT-CHAR NIL
:EXTEN-INDEX NIL).
```

See §11 of the TFM documentation for more information.

## Warning: Spurious Char Info

- ▶ Absent character  $\neq 0$
- ▶ 66 fonts affected
- ▶ Only a warning
  - ▶ `pltotf` zeroes out those chars
  - ▶ `tftopl` ignores them
- ▶ Obscure matter  
*Apparently not related to boundary chars*

### fcmb7t Compliance Report

Return to [toplevel](#).

File: public/cm-lgc/fcmb7t.tfm

1 warning, 0 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate](#) 1.1 "DT Cloistered Monk" / [TFM](#) 1.3 "Elaïd".

### Warnings

#### Spurious Char Info

While parsing char info table at position 32/170, for character code 32, char-info structure for a non-existent character is not blank

```
#S(CHAR-INFO
:WIDTH-INDEX 0
:HEIGHT-INDEX 0
:DEPTH-INDEX 0
:ITALIC-INDEX 0
:LLIG/KERN-INDEX 58
:NEXT-CHAR NIL
:EXTEN-INDEX NIL).
```

See §11 of the TFMtoPL documentation for more information.

# Error: Fix Word Overflow

▶ Numerical value outside ]-16,+16[

## ArevSans-Bold Compliance Report

[Return to toplevel.](#)

File: public/arev/ArevSans-Bold.tfm

0 warnings, 92 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate 1.1](#) "QT Cloistered Monk" / [TFM 1.3](#) "Eiad".

## Errors

### Fix Word Overflow

While parsing widths table at position 1/64,  
fix word 168225287/524288 (320.86426) is outside ]-16,+16[.  
See §9 of the TFMtoPL documentation for more information.

### Fix Word Overflow

# Error: Fix Word Overflow

- ▶ Numerical value outside ]-16,+16[
- ▶ 2 fonts affected

## ArevSans-Bold Compliance Report

[Return to toplevel.](#)

File: public/arev/ArevSans-Bold.tfm

0 warnings, 92 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate 1.1](#) "QT Cloistered Monk" / [TFM 1.3](#) "Eiad".

## Errors

### Fix Word Overflow

While parsing widths table at position 1/64,  
fix word 168225287/524288 (320.86426) is outside ]-16,+16[.  
See §9 of the TFMtoPL documentation for more information.

### Fix Word Overflow



## Error: Fix Word Overflow

- ▶ Numerical value outside ]-16,+16[
- ▶ 2 fonts affected
- ▶ Error
  - ▶ Character metrics messed up
  - ▶ T<sub>E</sub>X refuses those fonts

### ArevSans-Bold Compliance Report

[Return to toplevel.](#)

File: public/arev/ArevSans-Bold.tfm

0 warnings, 92 errors.

Report generated on Mon Jul 15 13:37:15 2024 GMT+0 with [TFM Validate 1.1](#) "QT Cloistered Monk" / [TFM 1.3](#) "Eiad".

### Errors

#### Fix Word Overflow

While parsing widths table at position 1/64,  
fix word 168225287/524288 (320.86426) is outside ]-16,+16[.  
See §9 of the TFMtoPL documentation for more information.

#### Fix Word Overflow

Introduction

tfm

tfm-validate

T<sub>E</sub>X Live

Conclusion

# (Real) Conclusion

## (Real) Conclusion

### ▶ Wrap Up

- ▶ Motivation: curiosity rather than necessity  
*Incentive to fix (?), monitor new additions*
- ▶ One merit: the shape of TFM compliance today  
*Many non-compliant fonts, but only 2 really unusable*

## (Real) Conclusion

- ▶ Wrap Up
  - ▶ Motivation: curiosity rather than necessity  
*Incentive to fix (?), monitor new additions*
  - ▶ One merit: the shape of TFM compliance today  
*Many non-compliant fonts, but only 2 really unusable*
- ▶ Perspectives
  - ▶ Support OFM & JFM
  - ▶ Support other formats (Open Type)
  - ▶ Improve layout (indexes, cross-references, *etc.*)
  - ▶ Add a `write-font` function to `tfm`