

◌ Erewhon-Math ◌

Daniel Flipo
daniel.flipo@free.fr

7th February 2022

1 What is Erewhon-Math?

Erewhon-Math is an Utopia based Opentype mathematical font. The mathematical symbols and Greek letters are borrowed or derived from Michel Bovani's Fourier-GUTenberg, Latin letters and digits are borrowed from Michael Shape's Erewhon font.

It requires LuaTeX or XeTeX as engine and the unicode-math package¹.

It is meant to be used with Utopia based Opentype text fonts like Erewhon. For Fourier-GUTenberg users who want to switch to LuaLaTeX or XeLaTeX, the file `fourier-otf.sty` can be used as a replacement of `fourier.sty`.

Please note that the current version (0.50) is *experimental, do expect metrics and glyphs to change* until version 1.0 is reached. Comments, suggestions and bug reports are welcome!

2 Usage

2.1 Calling `\setmathfont`

A basic call for Erewhon-Math would be:

```
\usepackage{unicode-math}  
\setmathfont{Erewhon-Math.otf} % Call by file name or  
\setmathfont{Erewhon Math}    % Call by font name
```

this loads Erewhon-Math as math font with the default options, see subsections [3.1 on the following page](#), [3.2 on page 3](#) and [3.3 on page 4](#) for customisation.

Please note that the three sets of text fonts have to be chosen separately, f.i.:

```
\setmainfont{erewhon}2 % rm  
\setsansfont{Cabin}[Scale=MatchLowercase] % sf  
\setmonofont{Inconsolatazi4}[Scale=MatchLowercase] % tt
```

otherwise you would get Latin Modern for text fonts.

¹Please read the documentation `unicode-math.pdf`.

²Erewhon (with capital E) is fine with LuaTeX but would fail with XeTeX (`erewhon.fontspec` not found).

2.2 Calling `fourier-otf.sty` (recommended)

As an alternative to load Erewhon-Math you can type:

```
\usepackage[ options3 ]{fourier-otf}
```

it loads `unicode-math` with the default options, sets Erewhon-Math as Math font and Erewhon Text fonts as Roman fonts (families *sf* and *tt* left unchanged) but does a bit more:

1. it loads `fourier-orns.sty`, providing many text ornaments;
2. it checks at `\begin{document}` if packages `amssymb` or `latexsym` are loaded and issues warnings in case they are;
3. it provides aliases for glyphs named differently in Unicode, so that `latexsym` or AMS names are also available;
4. it defines specific Math characters like `\Bbbbackslash` (\backslash), `\vareemptyset` (\emptyset), `\parallelslant` (\parallel), `\shortparallelslant` (\parallel), etc.;
5. it reduces spacing in math mode: `\thinmuskip`, `\medmuskip` and `\thickmuskip` are reduced as in `fourier.sty`. The option `loose` disables these settings.

Apart from the `loose` option mentioned above, `fourier-otf.sty` provides two options `no-text` and `Scale=<decimal>` meant to be used to load the Erewhon-Math font together with roman text fonts other than Erewhon, while keeping the advantages 1. to 5. pointed in the preceeding list, f.i. `\usepackage[no-text,Scale=0.98]{fourier-otf}`

Option `no-text` can also be useful if Erewhon is to be loaded with specific options, f.i. `\usepackage[no-text]{fourier-otf}`
`\setmainfont{erewhon}[RawFeature=+onum;+ss01]`

3 What is provided?

Erewhon-Math provides all glyphs supplied by Fourier-GUTenberg plus all glyphs available in the `amssymb` and `latexsym` packages and many more. Therefore, these two packages *should not* be loaded as they might override Erewhon-Math glyphs.

Sans-serif, typewriter and fraktur styles are borrowed from Latin Modern fonts. See in section 3.5 on page 7 how to choose from other Math fonts for these styles.

A full list of available glyphs is shown in file `unimath-erewhon.pdf`.

3.1 Upright or slanted?

Package `unicode-math` follows $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ conventions for Latin and Greek letters: in math mode, the default option (`math-style=TeX`) prints Latin letters $a\dots z$ $A\dots Z$ and lowercase greek letters $\alpha\dots\omega$ slanted (italic) while uppercase greek letters $\mathrm{A}\mathrm{B}\mathrm{\Gamma}\dots\mathrm{\Omega}$ are printed upright. This can be changed by option `math-style` as shown in table 1 on the following page.

³Possible *options* are `loose`, `no-text`, `Scale=` or any of the options described in sections 3.1, 3.2 and 3.3.

Table 1: Effects of the `math-style` package option.

Package option	Latin	Greek
<code>math-style=ISO</code>	(a, z, B, X)	$(\alpha, \beta, \Gamma, \Xi)$
<code>math-style=TeX</code>	(a, z, B, X)	$(\alpha, \beta, \Gamma, \Xi)$
<code>math-style=french</code>	(a, z, B, X)	$(\alpha, \beta, \Gamma, \Xi)$
<code>math-style=upright</code>	(a, z, B, X)	$(\alpha, \beta, \Gamma, \Xi)$

Bold letters are printed upright except lowercase Greek letters which are slanted (the default option is `bold-style=TeX`). This can be changed by option `bold-style` as shown in table 2.

Table 2: Effects of the `bold-style` package option.

Package option	Latin	Greek
<code>bold-style=ISO</code>	$(\mathbf{a}, \mathbf{z}, \mathbf{B}, \mathbf{X})$	$(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\Xi})$
<code>bold-style=TeX</code>	$(\mathbf{a}, \mathbf{z}, \mathbf{B}, \mathbf{X})$	$(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\Xi})$
<code>bold-style=upright</code>	$(\mathbf{a}, \mathbf{z}, \mathbf{B}, \mathbf{X})$	$(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\Xi})$

Other possible customisation: ∇ is printed upright and ∂ is printed slanted by default, but `nabla=italic` and `partial=upright` can change this.

All these options are offered by the `unicode-math` package but they can be added to the `\setmathfont` call⁴, for example:

`\setmathfont{Erewhon Math}[math-style=french,partial=upright]`
will print for the code

`\[\frac{\partial f}{\partial x} = \alpha \mathbf{\nabla} + a \nabla \Gamma + \beta \mathbf{M} + \mathbf{\nabla} \beta \mathbf{M} \]`

$$\frac{\partial f}{\partial x} = \alpha \mathbf{V} + a \nabla \Gamma + \beta \mathbf{M}$$

while the default settings would print

$$\frac{\partial f}{\partial x} = \alpha \mathbf{V} + a \nabla \Gamma + \boldsymbol{\beta} \mathbf{M}$$

Both shapes remain available anytime: `$\uppi, \it\pi$` prints π, π .

If your text editor is able to handle greek letters or math symbols, they can be entered in the code instead control sequences (i.e. $\alpha, \beta, \Gamma, \dots$ for `\alpha, \beta, \Gamma, \dots`).

3.2 Character variants

Erewhon-Math provides fourteen “Character Variants” options, listed on table 3 on the following page, to choose between different glyphs for Greek characters and some others. Alternative calligraphic capitals have been added for E, Q and T in version 0.50.

⁴IMHO it is easier to add *all options* to the `\setmathfont` command.

Table 3: Character variants.

	Default	Variant	Name
cv00	0	0	0
cv01	\hbar	\hbar	<code>\hslash</code>
cv02	\emptyset	\emptyset	<code>\emptyset</code>
cv03	ϵ	ϵ	<code>\epsilon</code>
cv04	κ	κ	<code>\kappa</code>
cv05	π	π	<code>\pi</code>
cv06	ϕ	ϕ	<code>\phi</code>
cv07	ρ	ρ	<code>\rho</code>
cv08	σ	σ	<code>\sigma</code>
cv09	θ	θ	<code>\theta</code>
cv10	Θ	Θ	<code>\Theta</code>
cv20	\mathcal{E}	\mathcal{E}	<code>\symcal{E}</code>
cv21	\mathcal{Q}	\mathcal{Q}	<code>\symcal{Q}</code>
cv22	\mathcal{T}	\mathcal{T}	<code>\symcal{T}</code>

For instance, to get `\epsilon` and `\phi` typeset as ϵ and ϕ instead of ϵ and ϕ , you can add option `CharacterVariant={3,6}` to the `\setmathfont` call:

```
\setmathfont{Erewhon Math}[CharacterVariant={3,6}]
```

This works for all shapes and weights of these characters: f.i. `\symbf{\epsilon}`, `\symbf{\phi}` are output as ϵ , ϕ instead of ϵ , ϕ .

Similarly with `math-style=french`, `\epsilon` and `\phi` are output as ϵ and ϕ (upright).

Please note that curly braces are mandatory whenever more than one “Character Variant” is selected.

Note about `\hbar` (v0.43): `unicode-math` defines `\hbar` as `\hslash` (U+210F) while `amsmath` provides two different glyphs (italic h with horizontal or diagonal stroke).

`Erewhon Math` now follows `unicode-math`; the italic h with horizontal stroke can be printed using `\hslash` or `\hbar` together with character variant `cv01` or with `\mathbar` (replacement for AMS’ command `\hbar`).

3.3 Stylistic sets

`Erewhon-Math` provides five “Stylistic Sets” options to choose between different glyphs for families of mathematical symbols.

`StylisticSet=4`, alias⁵ `Style=leqslant`, converts (large) inequalities into their slanted variants, see table 5a on the next page.

`StylisticSet=5`, alias `Style=smaller`, converts some symbols into their smaller variants, see table 5b on the following page.

⁵These Style aliases are provided by `fourier-otf.sty`.

Table 4: Stylistic Sets 4 and 5

(a) Style=leqslant (+ss04)			(b) Style=smaller (+ss05)		
Command	Default	Variant	Command	Default	Variant
<code>\leq</code>	\leq	\leqslant	<code>\in</code>	\in	\in
<code>\geq</code>	\geq	\geqslant	<code>\ni</code>	\ni	\ni
<code>\nleq</code>	\nleq	\nleqslant	<code>\mid</code>	\mid	\mid
<code>\ngeq</code>	\ngeq	\ngeqslant	<code>\nmid</code>	\nmid	\nmid
<code>\leqq</code>	\leqq	\leqslant	<code>\parallel</code>	\parallel	\parallel
<code>\geqq</code>	\geqq	\geqslant	<code>\nparallel</code>	\nparallel	\nparallel
<code>\eqless</code>	\eqless	\eqless	<code>\parallelslant</code>	\parallel	\parallel
<code>\eqgtr</code>	\eqgtr	\eqgtr	<code>\nparallelslant</code>	\nparallel	\nparallel
<code>\lesseqgtr</code>	\lesseqgtr	\lesseqgtr			
<code>\gtreqless</code>	\gtreqless	\gtreqless			
<code>\lesseqqgtr</code>	\lesseqqgtr	\lesseqqgtr			
<code>\gtreqqless</code>	\gtreqqless	\gtreqqless			

StylisticSet=6, alias Style=subsetneq, converts some inclusion symbols, see table 6a.

StylisticSet=7, alias Style=parallelslant, converts “parallel” symbols into their slanted variants, see table 6b.

Table 5: Stylistic Sets 6 and 7

(a) Style=subsetneq (+ss06)			(b) Style=parallelslant (+ss07)		
Command	Default	Variant	Command	Default	Variant
<code>\subsetneq</code>	\subsetneq	\subsetneq	<code>\parallel</code>	\parallel	\parallel
<code>\supsetneq</code>	\supsetneq	\supsetneq	<code>\nparallel</code>	\nparallel	\nparallel
<code>\subsetneqq</code>	\subsetneqq	\subsetneqq	<code>\shortparallel</code>	\parallel	\parallel
<code>\supsetneqq</code>	\supsetneqq	\supsetneqq	<code>\nshortparallel</code>	\nparallel	\nparallel

To enable Stylistic Sets 4, 6 and 7 for Erewhon-Math, you should enter

```
\setmathfont{Erewhon Math}[StylisticSet={4,6,7}] or
\usepackage[Style={leqslant,subsetneq,parallelslant}]{fourier-otf}
```

then, `\[x\leq y \quad A \subsetneq B \quad D \parallel D' \]` will print as

$$x \leqslant y \quad A \subsetneq B \quad D \parallel D'$$

instead of

$$x \leq y \quad A \subsetneqq B \quad D \parallel D'$$

StylisticSet=3, alias⁶ Style=upint, converts integrals signs into their upright variants, see table 6.

Table 6: Style=upint (+ss03)

Command	<code>\int</code>	<code>\iint</code>	<code>\iiint</code>	<code>\iiiiint</code>	<code>\oint</code>	<code>\oiint</code>	<code>\oiiint</code>	
Default	\int	\iint	\iiint	\iiiiint	\oint	\oiint	\oiiint	\oiint
Upright	\int	\iint	\iiint	\iiiiint	\oint	\oiint	\oiiint	\oiint

Command	<code>\intclockwise</code>	<code>\awint</code>	<code>\varointclockwise</code>	<code>\ointctrlockwise</code>
Default	\int	\int	\oint	\oint
Upright	\int	\int	\oint	\oint

3.4 Standard \LaTeX math commands

All standard \LaTeX math commands, all `amssymb` commands and all `latexsym` commands are supported by Erewhon-Math, for some of them loading `fourier-otf.sty` is required.

Various wide accents are also supported:

☞ `\widehat` and `\widetilde`

\hat{x} \hat{xy} \hat{xyz} $\hat{A \cup B}$ $\hat{A \cup (B \cap C) \cup D}$ \tilde{x} \tilde{xy} \tilde{xyz} $\tilde{A \cup B}$ $\tilde{A \cup (B \cap C) \cup D}$

☞ `\overline` and `\underline`

\overline{x} \overline{xy} \overline{xyz} $\overline{A \cup B}$ $\overline{A \cup (B \cap C) \cup D}$ $\underline{m+n+p}$

☞ `\wideoverbar`, `\widecheck` and `\widebreve`

\overline{x} \overline{xy} \overline{xyz} \check{x} \check{xy} \check{xyz} \breve{x} \breve{xy} \breve{xyz}

☞ `\overparen` and `\underparen`

\widehat{x} \widehat{xy} \widehat{xyz} $\overset{\circ}{A \cup B}$ $\overset{\circ}{A \cup (B \cap C) \cup D}$ $\overset{2}{x+y}$ $\overset{26}{a+b+\dots+z}$

\underline{x} \underline{xy} \underline{xyz} $\underline{\frac{x+z}{2}}$ $\underline{\frac{a+b+\dots+z}{26}}$

⁶These Style aliases are provided by `fourier-otf.sty`.

☞ \overbrace and \underbrace

$$\overbrace{a} \quad \overbrace{ab} \quad \overbrace{abc} \quad \overbrace{abcd} \quad \overbrace{abcde} \quad \overbrace{a+b+c}^3 \quad \overbrace{a+b+\dots+z}^{26}$$

$$\underbrace{a} \quad \underbrace{ab} \quad \underbrace{abc} \quad \underbrace{abcd} \quad \underbrace{abcde} \quad \underbrace{a+b+c}_3 \quad \underbrace{a+b+\dots+z}_{26}$$

☞ \overbracket and \underbracket

$$\overbracket{a} \quad \overbracket{ab} \quad \overbracket{abc} \quad \overbracket{abcd} \quad \overbracket{abcde} \quad \overbracket{a+b+c}^3 \quad \overbracket{a+b+\dots+z}^{26}$$

$$\underbracket{a} \quad \underbracket{ab} \quad \underbracket{abc} \quad \underbracket{abcd} \quad \underbracket{abcde} \quad \underbracket{a+b+c}_3 \quad \underbracket{a+b+\dots+z}_{26}$$

☞ \overrightarrow and \overleftarrow

$$\overrightarrow{v} \quad \overrightarrow{M} \quad \overrightarrow{vv} \quad \overrightarrow{AB} \quad \overrightarrow{ABC} \quad \overrightarrow{ABCD} \quad \overrightarrow{ABCDEFGH}$$

$$\overleftarrow{v} \quad \overleftarrow{M} \quad \overleftarrow{vv} \quad \overleftarrow{AB} \quad \overleftarrow{ABC} \quad \overleftarrow{ABCD} \quad \overleftarrow{ABCDEFGH}$$

☞ Finally \widearc and \overrightarrow (loading fourier-otf.sty is required)

$$\widearc{AMB} \quad \overrightarrow{AMB}$$

3.5 Mathematical alphabets

☞ All Latin and Greek characters are available in italic, upright, bold and bold italic via the \symit{ }, \symup{ }, \symbf{ } and \symbf it{ } commands.

☞ Calligraphic alphabet (\symscr or \symcal or \mathcal command), uppercase:

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

☞ Blackboard-bold alphabet (\sybbb or \mathbb command), uppercase, lowercase and digits:

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

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 0 1 2 3 4 5 6 7 8 9

☞ Fraktur alphabet is borrowed from Latin Modern,

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 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

but this can be overwritten, i.e.

```
\setmathfont{Asana-Math.otf}[range=frak,Scale=MatchUppercase]
$\symfrac{ABCDEFGHJKLMN\ldots XYZ}{abcdefghijkl\ldots xyz}$
```

⌘ ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz

☞ Sans serif alphabet is borrowed from Latin Modern,
 ABCDEFGHIJKLMabcdefghijkl NOPQRSTUVWXYZmnopqrstuvwxyz
 but it can be borrowed from another Math font, i.e.

```
\setmathfont{STIXTwoMath-Regular.otf}[range={sfup,sfit},
                                         Scale=MatchUppercase]
$\symsfup{ABCD\ldots klm}\quad\symsfit{NOPQ\ldots xyz}$
```

ABCDEFGHIJKLMabcdefghijklm NOPQRSTUVWXYZnopqrstuvwxyz

☞ Typewriter alphabet is borrowed from Latin Modern,
 ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz
 but it can be borrowed from another Math font, i.e.

```
\setmathfont{STIXTwoMath-Regular.otf}[range=tt,Scale=MatchUppercase]
$\symtt{ABCDE\ldots XYZ}{abcde\ldots xyz}$
```

ABCDEFGHIJKLMNPOQRSTUVWXYZabcdefghijklmnopqrstuvwxyz

Like Latin Modern, Erewhon Math provides only four lowercase latin letters in script (or calligraphic) shape: *e*, *g*, *ℓ*, *σ* (`\mscre`, `\mscrg`, `\ell`, `\mscro`).

All others (range "1D4B6 to "1D4CF) have to be borrowed from another Math font if needed, i.e.

```
\setmathfont{LibertinusMath-Regular.otf}%
[range="1D4B6-"1D4CF, Scale=MatchLowercase]
```

3.6 Missing symbols

Erewhon-Math does not aim at being as complete as STIXTwoMath-Regular or Cambria, the current glyph coverage compares with TeXGyre Math fonts. In case some symbols do not show up in the output file, you will see warnings in the .log file, for instance:

Missing character: There is no \Rightarrow (U+2964) in font ErewhonMath

Borrowing them from a more complete font, say Asana-Math, is a possible workaround:

```
\setmathfont{Asana-Math.otf}[range={"2964"},Scale=1.02]
```

scaling is possible, multiple character ranges are separated with commas:

```
\setmathfont{Asana-Math.otf}[range={"294A-"2951","2964","2ABB-"2ABE"}]
```

Let's mention albatross, a useful tool to find out the list of fonts providing a given glyph: f.i. type in a terminal "albatross U+2964", see the manpage or albatross-manual.pdf.

3.7 Fourier ornaments

When loaded by `\usepackage{fourier-otf}`, Erewhon Math loads `fourier-orns.sty` which provides all logos and ornaments available in Fourier-GUTenberg.

`fourier-orns.sty` as of v2.x automatically fetches its glyphs in a specific Opentype font with LuaTeX or XeTeX engines and from a Type 1 font otherwise (pdfTeX).

- ☞ A variant of the euro symbol: `\eurologo` €, €, €, €.
- ☞ Two “starred” bullets: `\starredbullet` †, `\decosix` ✦.
- ☞ Decos and logos: `\warning` ⚠, `\noway` ☹, `\caution` ⚠, `\bomb` 💣, `\decoone` ✖, `\decotwo` ☞, `\decothreeleft` ☞, `\decothreeright` ☞, `\decofourleft` ☞, `\decofourright` ☞, `\floweroneleft` ☞, `\floweroneright` ☞, `\lefthand` ☞, `\righthand` ☞, `\textxswup` ✖, `\textxswdown` ✖.
- ☞ Smileys: `\grimace` ☹, `\texttthing` ☹.
- ☞ Leaves: `\aldineleft` ☞, `\aldineright` ☞, `\aldine` ☞, `\aldinesmall` ☞, `\leafleft` ☞, `\leafright` ☞, `\leafNE` ☞, `\leafNW` ☞, `\leafSE` ☞, `\leafSW` ☞.
- ☞ Pilcrows: `\oldpilcrowone` ¶, `\oldpilcrowtwo` ¶, `\oldpilcrowthree` ¶, `\oldpilcrowfour` ¶, `\oldpilcrowfive` ¶aaaa, `\oldpilcrowsix` ¶aaaaaaaaaaaa.

All these logos and ornaments are also available as described in the `fournier-orns` documentation: for instance, you could type `{\FourierOrns E 2 F}` to get ☞ ⚠ ☞.

Finally, some symbols are also provided in math mode, with other names:

- ☞ `\forbidden` (☹), `\beware` (⚠), `\boom` (💣),
- ☞ `\thething` (☹) is a *QED symbol* for a false proof. Of course, you don’t need it!
- ☞ `\xswordsup` (✖) and `\xswordsdown` (✖) may be used as tags for a debated statement, or for anything else.

4 Acknowledgements

All glyphs in Erewhon-Math are borrowed or derived either from Erewhon fonts or Fourier-GUTenberg package. Many thanks to Michael Sharpe and Michel Bovani for providing these.

I am grateful to George Williams and his co-workers for providing and maintaining FontForge and to Ulrik Vieth for his illuminating paper published in TUGboat 2009 Volume 30 about Open Type Math.

Thanks to Oliver Natt for providing valuable feedback!

