The metsymb package*

Frédéric P.A. Vogt
frederic.vogt@meteoswiss.ch

August 25, 2021

Abstract

This package introduces commands to generate meteorological symbols. As of August 25, 2021, these include: okta symbols (○, ⓘ, ◔, ...). This package effectively introduces a new font in which each symbol is assigned to a glyph, which can then be called individually from LaTeX documents via dedicated new commands.

Contents

1 Introduction

The creation of this package was initially motivated by the fact that, as of August 24, 2021, there were no dedicated Unicode element for all of the ten okta symbols. To the best of my knowledge, no LaTeX package provides a uniform set of these ten symbols either[1].

This humble package is a direct attempt to remedy –in part– to this sad state of affair. A new font, created using FontForge[2], is used to generate the new symbols, and assign them to individual glyphs. Individual glyphs can then be called individually using dedicated LaTeX commands.

2 Usage

Using the metsymb package is straightforward. By importing it via a not-so-surprising `\usepackage{metsymb}` in the preamble of your documents, you will gain access to the commands listed in Table 1.

*This document corresponds to metsymb v1.0, dated 2021/08/24.
[1]If you know of one, please let me know and I shall list it here !
[2]https://fontforge.org/en-US/

Table 1: metsymb commands for the okta symbols.

○	\zerookta	◒	\fiveoktas
◴	\oneokta	◕	\sixoktas
◔	\twooktas	◑	\sevenoktas
◔	\threeoktas	●	\eightoktas
◑	\fouroktas	⊗	\nineoktas

2.1 Using metsymb with matplotlib

The assembly of a dedicated font to store the metsymb symbols[3] is directly motivated by the fact that matplotlib requires proper font metrics to include symbols in Python plots.

Hence, metsymb can be used to include meteorological symbols inside Python plots, provided that the use of a system-wide LaTeX installation is enabled prior to generating the plots. Modifying the `rcparams` elements is one way to do so, as illustrated in the following minimal working example (stored in `metsymb_mwe.py`; see Fig. 1 for the result):

```
# -*- coding: utf-8 -*-
"""
Copyright (C) 2021 MeteoSwiss,
originally written by F.P.A. Vogt; frederic.vogt@meteoswiss.ch

This file may be distributed and/or modified under the conditions
of the BSD-3-Clause License.
The terms of this license are available at:

https://opensource.org/licenses/BSD-3-Clause

SPDX-License-Identifier: BSD-3-Clause

Module content: minimal working example of the metsymb LaTeX package
with matplotlib figures.
"""

# Import matplotlib
from matplotlib import pyplot as plt

# Set the proper rcparams elements
plt.style.use('./metsymb_mwe.mplstyle')

# Create a basic figure with some demo text in the center.
plt.close(1)
plt.figure(1, figsize=(4, 0.5))
plt.text(0.5, 0.5,
         r'\LARGE Hello World, here is a symbol: \threeoktas',
         ha='center')
plt.axis('off')

# Export to different format and display on-screen.
plt.savefig('metsymb_mwe.pdf')
plt.savefig('metsymb_mwe.png')
plt.show()
```

where `metsymb_mwe.mplstyle` contains:

```
text.usetex: True
text.latex.preamble: \usepackage{metsymb}
```

[3] instead of a simpler `tikz` approach, for example

Figure 1: Result of the `metsymb_mwe.py` demonstration script, illustrating how the metsymb package can be used with matplotlib.

WARNING !
The use of `text.latex.preamble` in the `rcparams` of matplotlib is not an officially supported feature of that package ! Proceed at your own risks !

3 Code development and bug reports

The metsymb package is being developed inside a dedicated Github repository under the MeteoSwiss organization, located at: https://github.com/MeteoSwiss/metsymb. User contributions are welcome and will be examined in details. So are bug reports and suggestions for new symbols, which are best submitted as *Github Issues* directly on the code's repo at: https://github.com/MeteoSwiss/metsymb/issues

4 License and copyright

5 Ackowledgments

The following resources proved immensely useful to assemble the first version of this package:

- *How to Package Your LaTeXPackage*, Scott Pakin (2015): https://mirror.foobar.to/CTAN/info/dtxtut/dtxtut.pdf
- The FontForge documentation, and in particular the *FontForge and TeX* article: https://fontforge.org/docs/techref/PfaEdit-TeX.html
- The *TeX font errors: Cheatsheet*: https://texdoc.org/serve/tex-font-errors-cheatsheet/0

Several StackOverflow users also proved extremely helpful when building metsymb, in particular:

- those that provided clarifications and help in this post, in that post, and in that other post.

Thank you also to the jklymak and anntzer.lee from the matplotlib discourse community for their clarifications in this post.

6 Font table

The complete font table for metsymb, generated via the command `pdftex testfont` with the `\sample` call, is visible in Fig. 2.

Test of metsymb on August 25, 2021 at 1100

	′0	′1	′2	′3	′4	′5	′6	′7	
′06x	○	◫	◔	◔	◑	◑	◕	◐	″3x
′07x	●	⊗							
	″8	″9	″A	″B	″C	″D	″E	″F	

◫◑ ◫●●◑ ◔◑

Figure 2: Complete font table for metsymb.

7 Implementation

The metsymb package very simply defines new commands to fetch individual glyphs from the `metsymb` font. As such, its LaTeX side is rather simple.

`\zerookta` The 0 okta symbol:

```
\newcommand{\zerookta}{{\usefont{U}{metsymb}{m}{n} 0}}%
```

`\oneokta` The 1 okta symbol:

```
\newcommand{\oneokta}{{\usefont{U}{metsymb}{m}{n} 1}}%
```

`\twooktas` The 2 oktas symbol:

```
\newcommand{\twooktas}{{\usefont{U}{metsymb}{m}{n} 2}}%
```

`\threeoktas` The 3 oktas symbol:

```
\newcommand{\threeoktas}{{\usefont{U}{metsymb}{m}{n} 3}}%
```

`\fouroktas` The 4 oktas symbol:

```
\newcommand{\fouroktas}{{\usefont{U}{metsymb}{m}{n} 4}}%
```

`\fiveoktas` The 5 oktas symbol:

```
\newcommand{\fiveoktas}{{\usefont{U}{metsymb}{m}{n} 5}}%
```

`\sixoktas` The 6 oktas symbol:

```
\newcommand{\sixoktas}{{\usefont{U}{metsymb}{m}{n} 6}}%
```

`\sevenoktas` The 7 oktas symbol:

```
\newcommand{\sevenoktas}{{\usefont{U}{metsymb}{m}{n} 7}}%
```

`\eightoktas` The 8 oktas symbol:

```
\newcommand{\eightoktas}{{\usefont{U}{metsymb}{m}{n} 8}}%
```

`\nineoktas` The 9 oktas symbol:

```
\newcommand{\nineoktas}{{\usefont{U}{metsymb}{m}{n} 9}}%
```