The mercatormap package

Manual for version 1.02 (2020/08/06)

Thomas F. Sturm
\begin{tcolorbox}[spread,blankest]
\mermapset{flex tile size=40mm}
\begin{tikzpicture}
\mrcmap[type=areafit,west=5,east=15,south=46,north=54,
source = opentopomap,
flex area scale=4 000 000,
tex width=\tcbtextwidth,
tex height=\tcbtextheight,
]{title}
\mrcclipmap
\ExplSyntaxOn
\sys_gset_rand_seed:n {15}
\int_set:Nn \l_tmpa_int {1}
\int_until_do:nNnn \l_tmpa_int > {77}
{
\fp_set:Nn \l_tmpa_fp {\tcbtextwidth/2
 + \l_tmpa_int*2.5mm*cosd(18*\l_tmpa_int)}
\fp_set:Nn \l_tmpb_fp {\tcbtextheight/2
 + \l_tmpa_int*2.5mm*sind(18*\l_tmpa_int)}
\node[inner sep=0pt,draw=blue!50!gray,line width=1mm,
at=\fp_to_dim:Nn\l_tmpa_fp,rotate=\fp_eval:n{18*\l_tmpa_int+10*rand()}]
{\includegraphics[width=55mm,height=55mm] \tiles/opentopomap_
 \int_use:N \l_mermap_tile_zoom_int_
 \fp_eval:n {
 \l_randint(\l_mermap_tile_xmin_int,\l_mermap_tile_xmax_int) \}
 \fp_eval:n {
 \l_randint(\l_mermap_tile_ymin_int,\l_mermap_tile_ymax_int) }.png}
};
\int_incr:N \l_tmpa_int
\ExplSyntaxOff
\node[font=small,footnotesize,fill=white,opacity=0.75,text opacity=1]
at (\tcbtextwidth/2,1cm) {\mrcmapattribution};
\node at (\tcbtextwidth/2,\tcbtextheight*0.667)
{\begin{tcolorbox}[center upper,fontupper=\bfseries,boxsep=15mm,boxrule=4mm,
sharp corners,oversize=5mm,
colback=white,colframe=blue!50!gray,
enhanced jigsaw,opacityback=0.8,opacityframe=0.9]
{\Huge The mercatormap package\par}
\vspace{1cm}
Manual for version \version\ (\datum)\par
\vspace{5mm}
Thomas F.-Sturm
\end{tcolorbox}};
\end{tikzpicture}
\end{tcolorbox}
The mercatormap package

Manual for version 1.02 (2020/08/06)

Thomas F. Sturm

https://www.ctan.org/pkg/mercatormap
https://github.com/T-F-S/mercatormap

Abstract

The mercatormap package extends TikZ with tools to create map graphics. The provided coordinate system relies on the Web Mercator projection used on the Web by OpenStreetMap and others. The package supports the seamless integration of graphics from public map tile servers by a Python script. Also, common map elements like markers, geodetic networks, bar scales, routes, orthodrome pieces, and more are part of the package.

Contents

1 Introduction .......................... 5
  1.1 Quick Start .......................... 5
  1.2 Installation of Python and Packages .......................... 6
    1.2.1 Python 3 .......................... 6
    1.2.2 Python Packages ...................... 6
    1.2.3 Document Setup ...................... 6

2 Examples .......................... 7
  2.1 Reference Position ...................... 7
  2.2 Fitting Area .......................... 9
  2.3 Fixed Boundaries ...................... 10
  2.4 Map Without Map Tiles ...................... 11
  2.5 Alignment of the Reference Position ................. 14
  2.6 Flexible Zoom .......................... 15

3 Map Definition and Map Coordinates ...................... 17
  3.1 Option Setting .......................... 17
  3.2 Manual Map Definition ...................... 18
  3.3 Further Map Definition Options ...................... 18
  3.4 Map Coordinate System ...................... 20
  3.5 Named Positions .......................... 22
  3.6 Tests for Points to be inside or outside a Map ................. 23
  3.7 Formatted Coordinate Output ...................... 24

4 Automated Map Definition and Map Tiles ...................... 27
  4.1 Script Activation ...................... 27
  4.2 Map Types .......................... 28
  4.3 Map Supply .......................... 29

1Prof. Dr. Dr. Thomas F. Sturm, Institut für Mathematik und Informatik, Universität der Bundeswehr München, D-85577 Neubiberg, Germany; email: thomas.sturm@unibw.de
1 Introduction

The mercatormap package enables map drawing with the Web Mercator projection. This is done as an extension to TikZ [2] with is complemented by a map coordinate system and many additional commands and options to add elements like markers, geodetic networks, bar scales, routes, orthodrome pieces, distance calculations, etc. Also, the seamless integration of graphics from public map tile servers is provided through a Python script.


With very few exceptions, the package is programmed with the expl3 [3] programming interface for L\(\text{A}\)TEX3 and loads the packages xparse [4] and siunitx [5].

1.1 Quick Start

The package is accompanied with a Python script. You should read Section 1.2 on page 6 for the Python preparations. The package can be used in three ways:

- Completely without the Python script. This is not recommended, because the usage will be quite restricted.

- With Python script, but without map tile download. There is no usage restriction, but you have to create all content yourself. To prevent map tile download, set

\[
\text{\texttt{\textbackslash mermapset\{supply/target=none\}}} \]

- With Python script and map tile download. You need permission and access to a map tile server. Section 4.5 on page 36 lists a selection of servers with free access (some require registration of an API key).

After Python is prepared, you may try to compile mercatormap-example.tex (found in the documentation directory) which contains a map of Bavaria with map tile download. Section 2 on page 7 exhibits further examples which may serve as tutorials what can be done. After the examples you find the reference manual for the package.
1.2 Installation of Python and Packages

A Python 3 script is part of the mercatormap package. The main purpose of this script is to download selected map tiles for the maps of the document. Also, some coordinate system computation is done by this script.

1.2.1 Python 3

Python 3 is a required prerequisite and can be downloaded from https://www.python.org/downloads/
On systems like Linux Python is typically already installed.

To test your installation, type into a command or terminal window:

```
python --version
```

This should give a version number starting with 3. Otherwise, try

```
python3 --version
```

If this is successful, `/mermap/python` has to be adapted to `python3`

1.2.2 Python Packages

The Python packages Pillow (https://pypi.org/project/Pillow/) and requests (https://pypi.org/project/requests/) have to be present. With some luck, they are already installed. With

```
pip3 list
```

and/or

```
pip3 list --user
```

the installed packages are listed. If Pillow and requests are not among these package, they have to be added by

```
pip3 install --user Pillow
pip3 install --user requests
```

or

```
pip3 install Pillow
pip3 install requests
```

The second choice needs administrative rights and may give conflicts with package managers. Pythonians know furthers installation methods.

1.2.3 Document Setup

For your map document you need the following:

- Add `\mrcactivatescript` to the document preamble. Without this command, the script is not active.

- Compile the map document with the `--shell-escape` compiler option. This allows to execute external programs like the Python script.

  Be aware that `--shell-escape` should only be used with trusted documents. Note that external programs can do anything!
2 Examples

The following map examples may be used as tutorials and starting point for own applications. Also see mercatormap-example.tex for a compilable full example. Note to do all preparations documented in Section 1.2 on page 6.

2.1 Reference Position

With /mermap/supply/type=reference a map with a reference position is constructed. Here, Munich is taken as reference position and center of the map. Since the position is used more than once, it is stored with \mrcNPdef for further reference. With /mermap/supply/flex reference scale=1:6000000 the scale is set to 1:6000000. For the background map tiles, a /mermap/supply/source is selected for download. This setup is done by \mrcmap while \mrcdrawmap draws the downloaded map tiles.

\begin{tikzpicture}
\sffamily
\mrcNPdef{munich}{48.137222}{11.575556}
\mrcNPdef{vienna}{48.208333}{16.373056}
\mrcNPdef{cologne}{50.938056}{6.956944}
\mrcNPdef{milano}{45.4625}{9.186389}
\mrcmap[type=reference, named position=munich, flex reference scale=6 000 000, source=topplusopen web, tex width=14cm, tex height=14cm]{examples_reference}
\path[draw=yellow!50!gray,fill=yellow!20] ([xshift=-2mm,yshift=-5mm]mrcmap.south west) rectangle ([xshift=2mm,yshift=15mm]mrcmap.north east);
\mrcdrawmap
\node[below,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south) {\mrcmapattribution};
\path[draw=yellow!50!gray] (mrcmap.south west) rectangle (mrcmap.north east);
\mrcNPdraworthodrome{munich}{milano}
\path[blue,very thick] (mrcNPcs{munich}) -- node[red,fill=white,sloped,below] {\mrcNPprettyloxodistance{munich}{milano}} (mrcNPcs{milano});
\mermapsetmarker{type=pin, draw=red, fill=red!10, font=\sffamily\small}
\mrcmarker{named position=munich, contents={M"unchen}}
\mrcmarker{named position=vienna, contents={Wien}}
\mrcmarker{named position=cologne, contents={K"oln}}
\mrcmarker{named position=milano, contents={Milano}}
\node[above left=5mm,font=\Large\bfseries] at (mrcmap.north east) {Munich};
\node[above right] at ([yshift=10mm]mrcmap.north west) {Scale \mrcprettymapscale};
\mrcdrawscalebar[width-in-km=300,partitions=6,north-west-outside=0mm;5mm, single, height=1mm, major style={yellow!50!gray!50!black}]
\path[\every node/.style={above,inner sep=0.5mm,font=\sffamily\tiny}] (mrcscalebar.north west) -- (mrcscalebar.north east)
node[pos=0]{0} node[pos=0.3333]{100} node[pos=0.6667]{200}
node[pos=1] {300 km};
\end{tikzpicture}
2.2 Fitting Area

With \texttt{/mermap/supply/type^{P.29}=areafit} a map is constructed where a given area is fitted in. The following example lists some US-American cities and constructs an \texttt{/mermap/supply/area^{P.30}} which contains all of them. With \texttt{/mermap/supply/flex area fit^{P.34}=15mm} a border region is added.

\begin{tikzpicture}
\mrcNPdef{honolulu}{21.305225}{-157.867}
\mrcNPdef{fairbanks}{64.8379435}{-147.7192214}
\mrcNPdef{sandiego}{32.7146781}{-117.1640995}
\mrcNPdef{miami}{25.7599333}{-80.1951257}
\mrcNPdef{boston}{42.359744}{-71.061322}
\mrcNPdef{denver}{39.7372435}{-104.997378}
\mrcmap[type=areafit, 
area={honolulu,fairbanks,sandiego,miami,boston,denver}, 
source=stamen terrain, 
tex width=15cm, tex height=11cm, 
flex area fit=15mm, ]{examples_fitting_area}
\mrcdrawmap \mrcdrawnetwork
\node[below,font={\fontsize{7pt}{7pt}\sffamily}] at (mrcmap.south) {\mrcmapattribution};
\node[below left,fill=white,opacity=0.8,text opacity=1] at (mrcmap.north east) {Scale \mrcprettymapscale};
\path[draw] (mrcmap.south west) rectangle (mrcmap.north east);
\foreach \city in {honolulu,fairbanks,sandiego,miami,boston,denver}
{\mrcmarker[type=ringx, draw=red, fill=red!20, named position=\city}}
\end{tikzpicture}

Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
2.3 Fixed Boundaries

With `mermap/supply/type`\textsuperscript{P.29}=\texttt{boundaries} a map is constructed with fixed boundaries. In contrast to the other map types, the document map size cannot be given directly but derives from the map setup. This bears the risk of too large maps. The following example is a map with exact boundaries 45° S to 10° S and 110° E to 155° E. A decent `mermap/supply/zoom`\textsuperscript{P.30} is 5 (every zoom step doubles the map size in each direction).

\begin{tikzpicture}
\mrcmap[type=boundaries,
west=110,east=155,south=-45,north=-10,
zoom=5,
source=thunderforest outdoors,
]{examples_boundaries}
\mrcdrawmap
\node[below,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south) {
\mrcmapattribution};
\mrcdrawnetwork
\mrcdrawscalebar[width-in-km=1000,partitions=4,south-west-inside=5mm,
major style={blue!50!gray!50!black}]
\path[every node/.style={below,inner sep=0.5mm,font=\sffamily\tiny}]
(mrcscalebar.south west) -- (mrcscalebar.south east)
node[pos=0]{} node[pos=0.5]{} node[pos=1]{} node[pos=1,right,yshift=1mm]{km};
\mrcmarker[lat=-35.3,lon=149.116667,type=pictodropring,
draw=red,fill=red!10]
\end{tikzpicture}
2.4 Map Without Map Tiles

There is no coercion to use downloaded map tiles, if they are not needed or wanted. With \verb+/mermap/supply/target=no/+ none no map tiles are downloaded. The following example draws a rough polygon shape of Germany using \verb+/mrcroute/+.

\begin{tikzpicture}
\mrcNPdef{munich}{48.137222}{11.575556}
\mrcNPdef{cologne}{50.938056}{6.956944}
\mrcNPdef{hamburg}{53.550556}{9.993333}
\mrcNPdef{berlin}{52.518611}{13.408333}
\mrcmap[type=areafit, west=5,east=15,south=47,north=55, target=none, tex width=14cm, tex height=14cm, flex area fit=5mm ]{examples_routemap}
\mrcclipmap
\path\[draw,fill=yellow!5\] (mrcmap.south west) rectangle (mrcmap.north east);
\begin{mrcroute*}[\preaction={fill=black,opacity=.5, transform canvas={xshift=1mm,yshift=-1mm}}, draw = green!50!black, top color = green!50!gray!5, bottom color = green!50!gray!15]
\mrcpoint{47.57268069220318}{8.07968771809688}
\mrcpoint{47.55206513030159}{8.453802263852103}
\mrcpoint{47.6052271644701}{8.58564576447632}
\mrcpoint{47.65002478441761}{8.475977737743394}
\mrcpoint{48.129149100372}{8.621611270810012}
\mrcpoint{47.70405949548734}{8.824753417980977}
\mrcpoint{47.57118357939037}{9.37387060586875}
\mrcpoint{47.57763968687206}{9.783024032450792}
\mrcpoint{47.46598581489455}{10.09335967295321}
\mrcpoint{47.28014625489598}{10.2267829169482}
\mrcpoint{47.32098723021947}{10.40920267292467}
\mrcpoint{47.52413897927237}{10.5389647271817}
\mrcpoint{47.5130331145614}{10.9440341747166}
\mrcpoint{47.39097338968944}{11.0473476381045}
\mrcpoint{47.42252867275465}{11.44170163808758}
\mrcpoint{47.57832219946236}{11.64025530020549}
\mrcpoint{47.65203740659043}{12.2451260343467}
\mrcpoint{47.64005851006706}{12.78135708694006}
\mrcpoint{47.42189293170733}{13.0624216695717}
\mrcpoint{47.68727761120528}{13.13421507093485}
\mrcpoint{47.70570852317097}{13.97737528172392}
\mrcpoint{47.86836520373472}{14.9442906710615}
\mrcpoint{48.06063312315187}{14.76937524145701}
\mrcpoint{48.33220372303314}{14.38497370385176}
\mrcpoint{48.5706035511297}{15.55281830406328}
\mrcpoint{48.51465473490364}{17.8916086481733}
\mrcpoint{48.74889509826911}{18.8448153318651}
\mrcpoint{48.95974762545411}{19.5095252517658}
\mrcpoint{49.37381433564681}{20.77000417129028}
\mrcpoint{49.78495654813383}{21.44617209199484}
\mrcpoint{49.93155217044134}{22.49159658142377}
\mrcpoint{50.13144236663469}{22.2456848540114}
\mrcpoint{50.28771270517358}{23.12081015066419}
\end{mrcroute*}
\end{tikzpicture}
2.5 Alignment of the Reference Position

With \texttt{/mermap/supply/align} the reference position can be aligned at different map positions.

\begin{tikzpicture}
\mrcmap[type=reference,
  named position=munich,
  flex reference scale=1 000 000,
  source=opentopomap,
  tex width=\linewidth,tex height=3cm,
  align=\a]{examples_alignment_\a}
\mrcdrawmap
\node[below,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south)
{\mrcmapattribution}; \mrcclipmap
\path[draw] (mrcmap.south west) rectangle (mrcmap.north east);
\ifmrcNPinvicinity{munich}{\fill[red] (mrcpos) circle (4pt) node[below] {M"unchen};}{}
\node[text=red,above left] at (mrcmap.south east) {align=\a};
\end{tikzpicture}

\node[below,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south)
{\mrcmapattribution}; \mrcclipmap
\path[draw] (mrcmap.south west) rectangle (mrcmap.north east);
\ifmrcNPinvicinity{munich}{\fill[red] (mrcpos) circle (4pt) node[below] {M"unchen};}{}
\node[text=red,above left] at (mrcmap.south east) {align=\a};
\end{tikzpicture}

Kartendaten: © OpenStreetMap Mitwirkende, SRTM | Kartendarstellung: © OpenTopoMap (CC-BY-SA)
2.6 Flexible Zoom

Map tiles are only provided at fixed zoom levels with natural numbers, but the package allows a \texttt{/mermap/flex zoom} with rational numbers. The flexible zoom is realized by combining a suitable fixed zoom with an adapted document tile scaling, see Section 5.2 on page 51. The following example shows a more or less smooth zoom increase. The same technique is used by all options starting with \texttt{flex}, e.g. \texttt{/mermap/supply/flex reference scale} or \texttt{/mermap/supply/flex area fit} as seen in the examples before.

\begin{tikzpicture}
\mermapset{flex zoom=7}
\mrcmap[type=reference, named position=munich, source=thunderforest outdoors, tex width=\linewidth, tex height=3cm}{examples_flex_zoom_7}
\mrcdrawmap
\mrcclipmap
\tikzset{every node/.style={fill=white,fill opacity=0.5,text opacity=1}}
\node[above,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south) {\mrcmapattribution};
\path[draw] (mrcmap.south west) rectangle (mrcmap.north east);
\node[below left=2mm,align=right] at (mrcmap.north east) {flex zoom=\fpeval{round(7,1)}\scale \mrcprettymapscale};
\ifmrcNPinvicinity{munich}{\fill (mrcpos) circle (2pt) node[below] {M"unchen};}{}
\end{tikzpicture}
3 Map Definition and Map Coordinates

\begin{tikzpicture}
\mrcdefinemap{west=11,east=13,north=49,south=48}
\mrcdrawmap\[draw=path\]
\mrcdrawnetwork
\draw (mrc cs:latitude=48.53475,longitude=12.15087) circle (2mm);
\draw[red] (mrc cs:lat=48.53475,lon=12.15087) circle (3mm);
\draw[blue] (mrc cs:48.53475:12.15087) circle (4mm);
\ifmrcinmap{48.53475}{12.15087}{\draw[yellow] (mrcpos) circle (5mm);}{}
\ifmrcinvicinity{48.53475}{12.15087}{\draw[cyan] (mrcpos) circle (6mm);}{}
\end{tikzpicture}

3.1 Option Setting

\mermapset\{\langle options\rangle\}

Sets \textit{(options)} for all following maps inside the current \LaTeX{} group. All options share the common prefix `/mermap/`, e.g. for setting `/mermap/vicinity` use

\mermapset{vicinity=3cm}

Also see \mrcdefinemap \textsuperscript{*P.18}, \mermapsetsupply \textsuperscript{*P.29}, and \mermapsetmarker \textsuperscript{*P.62}. Note that the options by \mermapset are \texttt{exp13} \textsuperscript{[3]} keys while \LaTeX{} \texttt{2} uses its own key management.
3.2 Manual Map Definition

The following map definition is only relevant, if no script setup is used and maps are generated completely manually. See Section 4 on page 27 for script aided map definitions.

\mrcdefinemap\{⟨options⟩\}

Establishes a map inside a \texttt{tikzpicture} environment following and applying the given \texttt{⟨options⟩}. All options share the common prefix /mermap/mapdef/. After \mrcdefinemap is applied, map drawing and map coordinates can be used.

- \mrcdefinemap can be used directly, if no tile download and no script setup is intended.
- \mrcdefinemap is implicitly used with \mrcapplymap \textsuperscript{P.35} and \mrcmap \textsuperscript{P.35}. In this case, all options are also set implicitly.

\texttt{/mermap/mapdef/north=⟨map north latitude⟩} (no default, initially 50)

Northern latitude degree of the visible map, possibly negative for the southern hemisphere, lower than 90 but always larger than /mermap/mapdef/south. It is accessible as \mrcmapnorth (use read-only).

\texttt{/mermap/mapdef/south=⟨map south latitude⟩} (no default, initially 48)

Southern latitude degree of the visible map, possibly negative for the southern hemisphere, larger than –90 but always lower than /mermap/mapdef/north. It is accessible as \mrcmapsouth (use read-only).

\texttt{/mermap/mapdef/west=⟨map west longitude⟩} (no default, initially 11)

Western longitude degree of the visible map, possibly negative for the western hemisphere, possibly shifted periodically, but always lower than /mermap/mapdef/east. It is accessible as \mrcmapwest (use read-only).

\texttt{/mermap/mapdef/east=⟨map east longitude⟩} (no default, initially 13)

Eastern longitude degree of the visible map, possibly negative for the western hemisphere, possibly shifted periodically, but always larger than /mermap/mapdef/west. It is accessible as \mrcmapeast (use read-only).

3.3 Further Map Definition Options

The following options are typically implicitly set by \mrcapplymap \textsuperscript{P.35} and not manually by \mrcdefinemap. However, some values are computationally used in all cases. They can be ignored as pure technical information.

\texttt{/mermap/mapdef/xmin=⟨map tile x minimum⟩} (no default, initially 271)

Minimal \texttt{x} coordinate of the map tiles.

\texttt{/mermap/mapdef/xmax=⟨map tile x maximum⟩} (no default, initially 275)

Maximal \texttt{x} coordinate of the map tiles.

\texttt{/mermap/mapdef/ymin=⟨map tile y minimum⟩} (no default, initially 173)

Minimal \texttt{y} coordinate of the map tiles.

\texttt{/mermap/mapdef/ymax=⟨map tile y maximum⟩} (no default, initially 177)

Maximal \texttt{y} coordinate of the map tiles.

\texttt{/mermap/mapdef/zoom=⟨map zoom⟩} (no default, initially 9)

Map tile zoom factor alias \texttt{z} coordinate of the map tiles.

\texttt{/mermap/mapdef/pixelwidth=⟨map width in pixels⟩} (no default, initially 100)

Width of the visible map expressed in pixels of the source file(s). It is accessible as \mrcpixelwidth (use read-only).
/mermap/mapdef/pixelheight=(map height in tiles) (no default, initially 100)

Height of the visible map expressed in pixels of the source file(s). It is accessible as \mrcpixelheight (use read-only).

/mermap/mapdef/westoffset=(map tile offset (west)) (no default, initially 0)

Distance of the visible map from the western edge of the most western tile expressed in tiles (range from 0 to 1).

/mermap/mapdef/northoffset=(map tile offset (north)) (no default, initially 0)

Distance of the visible map from the northern edge of the most northern tile expressed in tiles (range from 0 to 1).

/mermap/mapdef/southoffset=(map tile offset (south)) (no default, initially 0)

Distance of the visible map from the southern edge of the most southern tile expressed in tiles (range from 0 to 1).

/mermap/mapdef/basename=(map tile base name) (no default, initially tiles/tile)

File base name for the tiles.

/mermap/mapdef/attribution=(attribution text) (no default, initially empty)

Attribution text for the map source. Typically, it acknowledges the copyright of the map data provider. It may contain hyperlinks. It is accessible as \mrcmapattribution (use read-only).

/mermap/mapdef/attribution print=(attribution text) (no default, initially empty)

Attribution text for the map source. In contrast to /mermap/mapdef/attribution it is intended for media that does not support hyperlinks like printed posters, books, etc. It is accessible as \mrcmapattributionprint (use read-only).

/mermap/mapdef/resource=(map resource) (no default, initially none)

Available map resource with following feasible values:
- none: No tiles and no merged map.
- tiles: Map tiles locally available.
- mergedmap: Single map picture file merged from tiles locally available.
- wmsmap: Single map picture file locally available.

/mermap/mapdef/tile size=(length) (no default, initially 32.512mm)

Typically set computationally. It is identical to /mermap/tile size •P.51 which is the recommended user option for manual setup.
### 3.4 Map Coordinate System

After a map is defined inside a `tikzpicture` environment by \texttt{\mrcdefinemap}\textsuperscript{P.18}, \texttt{\mrcapplymap}\textsuperscript{P.35}, or \texttt{\mrcmap}\textsuperscript{P.35}, a Mercator map coordinate system can be used. The border of the visible map is denoted by a TikZ node \texttt{mrcmap}.

```
\begin{tikzpicture}
  \mrcNPdef{nuremberg}{49.45522}{11.07631}
  \mermapset\{tile size=2cm\}
  \mrcdefinemap\{west=5,east=15,south=47,north=55,zoom=7\}
  \path\[draw,fill=green!10\] (mrcmap.south west) rectangle (mrcmap.north east);
  \mrcdrawnetwork
  \fill \texttt{mrc cs:latitude=48.137222,longitude=11.575556} circle (2pt) node\[below\] \{M"unchen\};
  \fill \texttt{mrc cs:lat=53.550556,lon=9.993333} circle (2pt) node\[above\] \{Hamburg\};
  \fill \texttt{mrcq cs:52.518611:13.408333} circle (2pt) node\[left\] \{Berlin\};
  \fill \texttt{\mrcNPcs{nuremberg}} circle (2pt) node\[above\] \{N"urnberg\};
  \ifmrcinmap\{50.938056}{6.956944}\{
    \fill \texttt{mrcpos} circle (1cm) node\[right\] \{K"oln\};\}\}
  \ifmrcNPinmap\{nuremberg\}\{
    \draw\[red\] \texttt{mrcpos} circle (1cm);\}\}
\end{tikzpicture}
```

The \texttt{mrc cs} coordinate system defines a map point by \texttt{/mermap/cs/latitude} and \texttt{/mermap/cs/longitude}.

\texttt{/mermap/cs/latitude=(latitude)} \hspace{4cm} (no default)

Sets the \texttt{(latitude)} of a map point.

\texttt{/mermap/cs/longitude=(longitude)} \hspace{4cm} (no default)

Sets the \texttt{(longitude)} of a map point.

```
\fill \texttt{mrc cs:latitude=48.137222,longitude=11.575556} circle (2pt);
```
A map point can also be defined by shorter variants `/mermap/cs/lat` and `/mermap/cs/lon`.

`/mermap/cs/lat=(latitude)` (no default)

Sets the (latitude) of a map point.

`/mermap/cs/lon=(longitude)` (no default)

Sets the (longitude) of a map point.

\begin{verbatim}
\fill (mrc cs:lat=48.137222,lon=11.575556) circle (2pt);
\end{verbatim}

A map point can be defined even quicker by `(mrcq cs:(latitude):(longitude))`.

\begin{verbatim}
\fill (mrcq cs:48.137222:11.575556) circle (2pt);
\end{verbatim}

\texttt{\textbackslash mrcpgfpoint\{(latitude)\}\{(longitude)\}}

Yields a low level \texttt{pgf} point location given by (latitude) and (longitude). This can be used like \texttt{\pgfpoint}.

\begin{verbatim}
\pgfpathcircle{\texttt{\textbackslash mrcpgfpoint\{49.45522\}\{11.07631\}}}{2pt}
\pgfusepath{fill}
\end{verbatim}
3.5 Named Positions

\mrcNPdef\{\langle name\rangle\}\{\langle latitude\rangle\}\{\langle longitude\rangle\}

A coordinate pair of \langle latitude\rangle and \langle longitude\rangle can be saved as named position (NP) to a \langle name\rangle for later use. The named position just stores the given values as evaluated floating points but without coordinate system processing. Therefore, a named position can be used outside a map definition or tikzpicture environment, even as a preset for the whole document. Note that this saving is not global but only effective inside the current \TeX group.

\mrcNPdef\{nuremberg\}\{49.45522\}\{11.07631\}

\mrcNPfrompoint\{\langle name\rangle\}\{\langle TikZ point\rangle\}

Latitude and longitude of a given \langle TikZ point\rangle are calculated and saved as named position (NP) with given \langle name\rangle. \mrcNPfrompoint can only be used after a valid a map definition inside a tikzpicture environment.

\mrcNPfrompoint\{mapcenter\}\{mrcmap.center\}
\mrcNPfrompoint\{mytest\}\{[xshift=1cm,yshift=1cm]mrcmap.south west\}

\mrcNPcs\{\langle name\rangle\}

A map point definition from the \langle name\rangle of a previously saved named position (NP).

\fill (\mrcNPcs\{nuremberg\}) circle (2pt) node[above] {N"urnberg};

\mrcNPlat\{\langle name\rangle\}

Inserts the latitude of a named position with given \langle name\rangle. \mrcNPlat is expandable and may be used in floating point expressions.

\mrcNPdef\{nuremberg\}\{49.45522\}\{11.07631\}
Latitude: \mrcNPlat\{nuremberg\}
Longitude: \mrcNPlon\{nuremberg\}

Latitude: 49.45522
Longitude: 11.07631

\mrcNPlon\{\langle name\rangle\}

Inserts the longitude of a named position with given \langle name\rangle. \mrcNPlon is expandable and may be used in floating point expressions.
3.6 Tests for Points to be inside or outside a Map

When a map is drawn, \texttt{\textbackslash mrclipmap} \textsuperscript{p.50} can be used to set up a \LaTeX{} clip environment which automatically removes all content which is not inside the defined map. However, the \LaTeX{} position of a geographic point has to be computed first to decide, if this point is to be drawn. Since \TeX{} length registers do not allow large dimensions, compiler errors are possible to happen.

The following tests check given geographic coordinates before they are transformed to \TeX{} dimensions and avoid such compiler errors.

\texttt{\textbackslash ifmrcinmap{\langle latitude\rangle}{\langle longitude\rangle}{\langle true\rangle}{\langle false\rangle}}

If the given \texttt{\langle latitude\rangle} and \texttt{\langle longitude\rangle} describes a point inside the visible map, the \texttt{\langle true\rangle} code is executed, otherwise the \texttt{\langle false\rangle} code.

Inside the \texttt{\langle true\rangle} code a \LaTeX{} coordinate \texttt{mrcpos} describes the given point. Also, \texttt{mrclastpos} denotes the last position before.

\begin{verbatim}
\ifmrcinmap{48.137222}{11.575556}{\fill (mrcpos) circle (2pt);}{}
\end{verbatim}

\texttt{\textbackslash ifmrcNPinmap{(name)}{\langle true\rangle}{\langle false\rangle}}

If the given \texttt{name} \texttt{(NP)} \texttt{(name)} describes a point inside the visible map, the \texttt{\langle true\rangle} code is executed, otherwise the \texttt{\langle false\rangle} code.

\begin{verbatim}
\mrcNPdef{munich}{48.137222}{11.575556}
\ifmrcNPinmap{munich}{\fill (mrcpos) circle (2pt);}{}
\end{verbatim}

Very similar to \texttt{\textbackslash ifmrcinmap} is \texttt{\textbackslash ifmrcinvicinity}.

\texttt{\textbackslash ifmrcinvicinity{\langle latitude\rangle}{\langle longitude\rangle}{\langle true\rangle}{\langle false\rangle}}

If the given \texttt{\langle latitude\rangle} and \texttt{\langle longitude\rangle} describes a point inside a vicinity of the visible map, i.e. the map plus a margin of \texttt{/mermap/vicinity}, the \texttt{\langle true\rangle} code is executed, otherwise the \texttt{\langle false\rangle} code.

Inside the \texttt{\langle true\rangle} code a \LaTeX{} coordinate \texttt{mrcpos} describes the given point. Also, \texttt{mrclastpos} denotes the last position before.

\texttt{\textbackslash ifmrcinvicinity} may be used for objects of a certain size like markers which could be partly visible even when their reference point is outside the visible map (but nearby).

\begin{verbatim}
\ifmrcinvicinity{48.137222}{11.575556}{\fill (mrcpos) circle (2pt);}{}
\end{verbatim}

\texttt{\textbackslash ifmrcNPinvicinity{(name)}{\langle true\rangle}{\langle false\rangle}}

If the given \texttt{name} \texttt{(NP)} \texttt{(name)} describes a point inside a vicinity of the visible map, the \texttt{\langle true\rangle} code is executed, otherwise the \texttt{\langle false\rangle} code, see \texttt{\textbackslash ifmrcinvicinity}.

\begin{verbatim}
\mrcNPdef{munich}{48.137222}{11.575556}
\ifmrcNPinvicinity{munich}{\fill (mrcpos) circle (2pt);}{}
\end{verbatim}

\texttt{/mermap/vicinity=\langle width\rangle} \hspace{1cm} (no default, initially 2cm)

The vicinity of the map is the given map plus a border in all directions with the given \texttt{\langle width\rangle}.
### 3.7 Formatted Coordinate Output

\texttt{\textbackslash mrcformlat\{\langle options\rangle\}\{\langle latitude\rangle\}}

Formatted output for a given \langle latitude\rangle following given \langle options\rangle. Formatting \langle options\rangle are described in the following.

<table>
<thead>
<tr>
<th>\textbackslash mrcformlat{-24.29} to \mrcformlat{12.3456789}.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latitude from \mrcformlat{-24.29} to \mrcformlat{12.3456789}.</td>
</tr>
<tr>
<td>Latitude from 24.2900° S to 12.3457° N.</td>
</tr>
</tbody>
</table>

\texttt{\textbackslash mrcformlon\{\langle options\rangle\}\{\langle longitude\rangle\}}

Formatted output for a given \langle longitude\rangle following given \langle options\rangle. Formatting \langle options\rangle are described in the following.

<table>
<thead>
<tr>
<th>\textbackslash mrcformlon{-24.29} to \mrcformlon{12.3456789}.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Longitude from \mrcformlon{-24.29} to \mrcformlon{12.3456789}.</td>
</tr>
<tr>
<td>Longitude from 24.2900° W to 12.3457° E.</td>
</tr>
</tbody>
</table>

\texttt{/mermap/format angle=(type)}

(no default, initially \texttt{decimal-4})

The \texttt{\textbackslash ang} macro from package \texttt{siunitx} \cite{siunitx} is used which can be controlled by further settings of \texttt{siunitx} like digit grouping or changing the decimal marker.

Feasible values for \texttt{\langle type\rangle} are

- \texttt{decimal}: decimal output without rounding.

\[
\texttt{\mermapset{format angle=decimal}}
\texttt{Longitude from \mrcformlon{-24.29} to \mrcformlon\{12.3456789\}.}
\]

Longitude from 24.29° W to 12.3456789° E.

- \texttt{decimal-0}: decimal output with rounding to full degrees.

\[
\texttt{\mermapset{format angle=decimal-0}}
\texttt{Longitude from \mrcformlon{-24.29} to \mrcformlon\{12.3456789\}.}
\]

Longitude from 24° W to 12° E.

- \texttt{decimal-1}: decimal output with rounding to one place.

\[
\texttt{\mermapset{format angle=decimal-1}}
\texttt{Longitude from \mrcformlon{-24.29} to \mrcformlon\{12.3456789\}.}
\]

Longitude from 24.3° W to 12.3° E.

- \texttt{decimal-2}: decimal output with rounding to two places.

\[
\texttt{\mermapset{format angle=decimal-2}}
\texttt{Longitude from \mrcformlon{-24.29} to \mrcformlon\{12.3456789\}.}
\]

Longitude from 24.29° W to 12.35° E.
- **decimal-3**: decimal output with rounding to three places.

  \begin{verbatim}
  \mermapset{format angle=decimal-3}
  Longitude from \mrcformlon{-24.29} to \mrcformlon{12.3456789}.
  \end{verbatim}

  Longitude from 24.290° W to 12.346° E.

- **decimal-4**: decimal output with rounding to four places.

  \begin{verbatim}
  \mermapset{format angle=decimal-4}
  Longitude from \mrcformlon{-24.29} to \mrcformlon{12.3456789}.
  \end{verbatim}

  Longitude from 24.2900° W to 12.3457° E.

- **degree**: output with rounding to full degrees. This is an alias for **decimal-0**.

  \begin{verbatim}
  \mermapset{format angle=degree}
  Longitude from \mrcformlon{-24.29} to \mrcformlon{12.3456789}.
  \end{verbatim}

  Longitude from 24° W to 12° E.

- **minute**: output with rounding to degrees and full minutes.

  \begin{verbatim}
  \mermapset{format angle=minute}
  Longitude from \mrcformlon{-24.29} to \mrcformlon{12.3456789}.
  \end{verbatim}

  Longitude from 24°17′ W to 12°21′ E.

- **second**: output with rounding to degrees, minutes, and full seconds.

  \begin{verbatim}
  \mermapset{format angle=second}
  Longitude from \mrcformlon{-24.29} to \mrcformlon{12.3456789}.
  \end{verbatim}

  Longitude from 24°17′24″ W to 12°20′44″ E.

/mermap/format south=⟨code⟩ (no default, initially #1\,S)  
Defines the format ⟨code⟩ for a negative latitude. Use #1 to place the number (without sign).

\begin{verbatim}
\mermapset{format south={$-#1$}}
Latitude \mrcformlat{-24.29}.
\end{verbatim}

Latitude –24.2900°.

/mermap/format north=⟨code⟩ (no default, initially #1\,N)  
Defines the format ⟨code⟩ for a non-negative latitude. Use #1 to place the number.

\begin{verbatim}
\mermapset{format north={#1 North}}
Latitude \mrcformlat{12.3456789}.
\end{verbatim}

Latitude 12.3457° North.
/mermap/format east=⟨code⟩

(no default, initially #1\,E)

Defines the format ⟨code⟩ for a positive longitude. Use #1 to place the number (without sign).

\mermapset{format east={#1\,O}}
\sisetup{output-decimal-marker={,}}
L"angengrad \mrcformlon{12.3456789}.

Längengrad 12,3457° O.

/mermap/format west=⟨code⟩

(no default, initially #1\,W)

Defines the format ⟨code⟩ for a negative longitude. Use #1 to place the number.

\mermapset{format west={West: #1}}
Longitude \mrcformlon{-24.29}.

Longitude West: 24.2900°.

/mermap/format NEWS numeric

(no value)

Defines the format for north, east, west, and south as numeric value without N, E, W, S.

\mermapset{format NEWS numeric}
Longitude \mrcformlon{-24.29} and \mrcformlon{12.3456789}.\
Latitude \mrcformlat{-24.29} and \mrcformlat{12.3456789}.

Longitude −24.2900° and 12.3457°.
Latitude −24.2900° and 12.3457°.

/mermap/format NEWS absolute

(no value)

Defines the format for north, east, west, and south as absolute value without N, E, W, S and without algebraic sign.

\mermapset{format NEWS absolute}
Longitude \mrcformlon{-24.29} and \mrcformlon{12.3456789}.\
Latitude \mrcformlat{-24.29} and \mrcformlat{12.3456789}.

Longitude 24.2900° and 12.3457°.
Latitude 24.2900° and 12.3457°.
4 Automated Map Definition and Map Tiles

As illustrated above, the script aided map definition is a process with several stages.

- **Map Supply**: \texttt{\textbackslash mrcsupplymap} \textsuperscript{P.29} is the replacement of the manual setup by \texttt{\textbackslash mrcdefinemap} \textsuperscript{P.18}. Actually, it is quite similar to \texttt{\textbackslash mrcdefinemap} \textsuperscript{P.18}. With \texttt{\textbackslash mrcsupplymap} \textsuperscript{P.29} directions for the following Python script are formulated.

- **Python Script**: The script is executed by \texttt{\textbackslash mrcsupplymap} \textsuperscript{P.29} during compilation. It does some coordinate system computations and downloads map tiles from a Web server. Finally, it writes a map definition into a file \texttt{(id).def}.

- **Map Apply**: \texttt{\textbackslash mrcapplymap} \textsuperscript{P.35} reads and applies the map definition from \texttt{(id).def}.

- **Map Drawing**: Afterwards, the map can be drawn by \texttt{\textbackslash mrcdrawmap} \textsuperscript{P.50} and other commands.

A map can be applied more than once, e.g. reused later in the document. If this is not needed, map supply and map apply can be combined by \texttt{\textbackslash mrcmap} \textsuperscript{P.35}.

4.1 Script Activation

Remember to install Python beforehand, see Section 1.2 on page 6.

\texttt{\textbackslash mrcactivatescript}

Use this inside the preamble of your document to activate the accompanying Python script. Without this command, the script is not executed! If the document is final (or the maps are final), this line could be removed and the document should be compilable without script.

\texttt{\textbackslash mermap/python=(python)} \hspace{1cm} (no default, initially \texttt{python})

Names the Python 3 interpreter as \texttt{(python)}. If your Python 3 interpreter is not called \texttt{python}, but e.g. \texttt{python3}, then use

\texttt{\textbackslash mermapset\{python=python3\}}
4.2 Map Types

Currently, there are three methods provided on how a map is computed by the accompanying Python script. The technical background is documented in [1, Section 5].

1. `/mermap/supply/type`\textsuperscript{\textcopyright}P. 29 = reference:
   
   The default method determines the map dimensions from a reference position and given document map dimensions. Also, a zoom level `/mermap/supply/zoom`\textsuperscript{\textcopyright}P. 30 is required which relates to the Web Mercator map tile covering of the Earth. A higher zoom level gives a growing number of smaller map tiles. Alternative to the zoom level, a \textit{(scale denominator)} can be provided by `/mermap/supply/flex area scale`\textsuperscript{\textcopyright}P. 33, `/mermap/flex scale`\textsuperscript{\textcopyright}P. 52 or `/mermap/supply/flex reference scale`\textsuperscript{\textcopyright}P. 33 which determines the zoom level implicitly. As default, the reference position is the center of the map, but can be aligned at the map borders. This method is quite safe to use and could be the preferred one for many applications like showing the neighborhood of a route target. Finding the best reference point for depicting a certain area could be more tricky.

2. `/mermap/supply/type`\textsuperscript{\textcopyright}P. 29 = area fit:
   
   The map dimensions are determined by an area with latitude and longitude boundaries which is fitted into given document map dimensions. The zoom level is computed accordingly for a fixed document tile size or by `/mermap/supply/flex area fit`\textsuperscript{\textcopyright}P. 34. In any case, the map contains the target area plus some protrusion. This method is also quite safe to use and may be the preferred one for many applications like showing a map which contains a bunch of markers (e.g. for locations of universities). If the map borders are required to exactly meet the boundaries, the third method can be regarded.

3. `/mermap/supply/type`\textsuperscript{\textcopyright}P. 29 = boundaries:
   
   The most obvious method determines the map dimensions from latitude and longitude boundaries. For this, a corresponding zoom level `/mermap/supply/zoom`\textsuperscript{\textcopyright}P. 30 is required which relates to the Web Mercator map tile covering of the Earth. Alternative to the zoom level, a \textit{(scale denominator)} can be provided by `/mermap/supply/flex area scale`\textsuperscript{\textcopyright}P. 33 or `/mermap/supply/flex reference scale`\textsuperscript{\textcopyright}P. 33 which determines the zoom level implicitly. Note that a too high zoom level imposes the risk of downloading an unwanted high quantity of map tiles resulting in a much too large document map. Therefore, this most obvious method is \textit{not recommended} for the beginner and may be explored after some experience.
4.3 Map Supply

\texttt{\textbackslash mrcsupplymap\{\langle options\}\{\langle definition\}\}}

The \langle options\rangle provide parameters for the Python 3 script to supply all materials for a map. All options share the common prefix /mermap/supply/.

The map is identified by
\langle id\rangle=/mermap/definition prefix+\langle definition\rangle

for later drawing. This identifier \langle id\rangle has to be unique for the document. It corresponds to generated files \langle id\rangle\.def, \langle id\rangle\.md5, and possibly \langle id\rangle\.png. Do not use spaces or special characters like umlauts for \langle definition\rangle.

If \texttt{\textbackslash mrcactivatescript} is used inside the preamble, \texttt{\textbackslash mrcsupplymap} executes the Python 3 script, otherwise nothing happens.

\texttt{\textbackslash mermapsetsupply\{\langle options\}\}}

Sets \langle options\rangle for all following maps inside the current \TeX\ group. All options share the common prefix /mermap/supply/, e.g. for setting /mermap/supply/type use

\texttt{\textbackslash mermapsetsupply\{type=reference\}}

Also see \texttt{\textbackslash mermapset} and \texttt{\textbackslash mermapsetmarker}.

\texttt{\textbackslash mermap/definition prefix=\langle definition prefix\rangle} (no default, initially maps/)

Prefix for map identifiers and generated map files, see \texttt{\textbackslash mrcsupplymap} and \texttt{\textbackslash mrcapplymap}. Note that /mermap/definition prefix is not to be used inside the option list for \texttt{\textbackslash mrcsupplymap}.

\texttt{\textbackslash mermap/supply/type=\langle type\rangle} (no default, initially reference)

The \langle type\rangle defines the basic computation for the map. Feasible values are

- \textbf{reference}: «map with reference position»
  - The map is constructed from a given reference position /mermap/supply/latitude, /mermap/supply/longitude, a zoom level /mermap/supply/zoom, map dimensions /mermap/supply/width, /mermap/supply/height, and alignment /mermap/supply/align.

- \textbf{areafit}: «map fitting an area»
  - The map is constructed from a given area boundaries /mermap/supply/west, /mermap/supply/east, /mermap/supply/north, /mermap/supply/south, map dimensions /mermap/supply/width, /mermap/supply/height, and alignment /mermap/supply/align.

- \textbf{boundaries}: «map with boundaries»
  - The map is constructed from given boundaries /mermap/supply/west, /mermap/supply/east, /mermap/supply/north, /mermap/supply/south, and zoom level /mermap/supply/zoom.
/mermap/supply/zoom={setup zoom}  (no default, initially 9)

Map tile zoom factor alias $z$ coordinate of the map tiles. Used for map types boundaries and reference.

/mermap/supply/north={setup north latitude}  (no default, initially 50)

Northern latitude degree, possibly negative for the southern hemisphere, lower than 90 but always larger than /mermap/supply/south. Used for map types boundaries and areafit.

/mermap/supply/south={setup south latitude}  (no default, initially 48)

Southern latitude degree, possibly negative for the southern hemisphere, larger than $-90$ but always lower than /mermap/supply/north. Used for map types boundaries and areafit.

/mermap/supply/west={setup west longitude}  (no default, initially 11)

Western longitude degree, possibly negative for the western hemisphere, possibly shifted periodically, but always lower than /mermap/supply/east. Used for map types boundaries and areafit.

/mermap/supply/east={setup east longitude}  (no default, initially 13)

Eastern longitude degree, possibly negative for the western hemisphere, possibly shifted periodically, but always larger than /mermap/supply/west. Used for map types boundaries and areafit.

/mermap/supply/area={{comma separated list of named positions}}  (no default)

/mermap/supply/add area={{comma separated list of named positions}}  (no default)

Sets /mermap/supply/north, /mermap/supply/south, /mermap/supply/west, /mermap/supply/east according to the given "comma separated list of named positions", i.e. the described area contains all these positions.

/mermap/supply/area resets the current area which requires at least two points inside the list.

/mermap/supply/add area does not reset the current area, i.e. the positions are added to the current area which possibly grows to fit all positions.

Also note to take special care, if the international dateline is on your resulting map, see Section 10.2 on page 81. Used for map types boundaries and areafit or in combination with /mermap/supply/area to reference also for for map type reference.

/mermap/supply/area from marker input={{file name}}  (no default)

/mermap/supply/add area from marker input={{file name}}  (no default)

Sets /mermap/supply/north, /mermap/supply/south, /mermap/supply/west, /mermap/supply/east according to the given \texttt{\textbackslash mrc\textbackslash marker} positions contained in a file with the given \texttt{(file name)}.

/mermap/supply/area from marker input resets the current area which requires at least two marker positions inside the file.

/mermap/supply/add area from marker input does not reset the current area, i.e. the positions are added to the current area which possibly grows to fit all positions.

Also note to take special care, if the international dateline is on your resulting map, see Section 10.2 on page 81. Used for map types boundaries and areafit or in combination with /mermap/supply/area to reference also for for map type reference.

/mermap/supply/area to reference  (no value, initially unset)

The map settings /mermap/supply/north, /mermap/supply/south, /mermap/supply/west, /mermap/supply/east are taken to compute the map center. This center position is saved to /mermap/supply/latitude \textsuperscript{\texttt{P.31}} and /mermap/supply/longitude \textsuperscript{\texttt{P.31}}. Used for map type reference.
/mermap/supply/latitude={setup latitude}  
Latitude degree of a reference point, possibly negative for the southern hemisphere. Used for map type reference.

/mermap/supply/longitude={setup longitude}  
Longitude degree of a reference point, possibly negative for the western hemisphere. Used for map type reference.

/mermap/supply/position={setup latitude}:{setup longitude}  
Latitude degree and longitude of a reference point. Used for map type reference.

/mermap/supply/named position={name}  
The named position given by {name} describes a reference point, see Section 3.5 on page 22. Used for map type reference.

/mermap/supply/width={setup width in tiles}  
Width of the map as multiplicity of map tiles. Used for map types reference and areafit.

/mermap/supply/tex width={width}  
Width of the map as \( \text{TeX} \) dimension. This is a style to compute \(/\text{mermap/supply/width}\) according to the current \(/\text{mermap/tile size}\) \( \text{\textsuperscript{P.51}} \).

/mermap/supply/height={setup height in tiles}  
Height of the map as multiplicity of map tiles. Used for map types reference and areafit.

/mermap/supply/tex height={width}  
Height of the map as \( \text{TeX} \) dimension. This is a style to compute \(/\text{mermap/supply/height}\) according to the current \(/\text{mermap/tile size}\) \( \text{\textsuperscript{P.51}} \).

/mermap/supply/align={setup alignment}  
Alignment of reference point or area for map types reference and areafit. Feasible values are northwest, north, northeast, west, center, east, southwest, south, southeast.

/mermap/supply/target={setup target}  
Defines the type of output for the Python 3 script. Feasible values are:
- none: No tiles are downloaded and no merged map is generated, just map computation. This is the fastest method and needs no tile supplier.
- tiles: Download map tiles from a tile map service (TMS) \(/\text{mermap/supply/url}\) \( \text{\textsuperscript{P.32}} \). Compilation of a document with map tile takes longer than compilation with a merged map and transparency should not be used with tiles, but the resulting document is smaller than a document with merged maps.
- mergedmap: Download map tiles from a tile map service (TMS) \(/\text{mermap/supply/url}\) \( \text{\textsuperscript{P.32}} \) and merge them into a single map picture. This speeds compilation and allows transparency effects, but the resulting document is possibly larger than a document with map tiles, because map tiles often are optimized 8-bit image files while the merged image is a 24-bit PNG file. Additionally, synergy effects of using the same map tiles for different maps are lost. Also, since the pixel map is clipped to full pixels, the resulting map may differ (shift/size) from the more accurate tile representation by one pixel.
- wmsmap: Download a single map from a web map service (WMS) \(/\text{mermap/supply/url}\) \( \text{\textsuperscript{P.32}} \). Internally, the package treats a WMS like a tile map service including all tile calculations. Actually, a single file is downloaded.

/mermap/fail on missing resource=true|false  
If set to true, compilation stops with an error, if \(/\text{mermap/supply/target}\) and \(/\text{mermap/mapdef/resource}\) \( \text{\textsuperscript{P.19}} \) are different. Typically, this means that something went wrong while trying to download map tiles. Set this option temporarily to false, if the map tile service or the internet connection is expected to be unavailable only temporarily.
Here, the url format with placeholder \{z\}\{x\}\{y\} for map tile download is defined. Be sure that you have the permission to download, save, and use the map tiles from that URL. Illegal downloads are not endorsed in any way.

\url{https://abc.efg.hij/{z}/{x}/{y}.png?apikey=12345678},

See Section 4.5 on page 36 for predefined URLs.

This is an alternative version of \mermap/supply/url. The URL is constructed from some fixed \langle prefix \rangle and \langle postfix \rangle with an API key in between. The API key is retrieved by \langle name \rangle from a repository filled by \mrcsetapikey.

\url{https://abc.efg.hij/{z}/{x}/{y}.png?apikey=}{myservice}{},

See Section 4.5 on page 36 for predefined URLs.

\mrcsetapikey{\langle name \rangle}{\langle value \rangle}

Stores an API key \langle value \rangle for access with the given \langle name \rangle. Typically, \langle value \rangle is a received ID from a map tile service provider after personal registration. \langle name \rangle is a placeholder which is used inside \mermap/supply/url with api key to mark the insertion point for the API key.

\mrcsetapikey{myservice}{....K942XY....}

\mermap/supply/attribution=\langle attribution text \rangle

Attribution text for the map data provider. It may contain hyperlinks. It is used to set up \mermap/mapdef/attribution afterwards and it is accessible as \mrcmapattribution (use read-only).

For technical reasons, do not use ". \mrcumlaut may be used for masking umlauts, e.g. use \mrcumlaut{u} instead of \"{u}, but umlauts can also be used directly, e.g. as UTF-8 coded characters.

\mermap/supply/attribution print=\langle attribution text \rangle

Attribution text for the map source. In contrast to \mermap/supply/attribution it is intended for media that does not support hyperlinks like printed posters, books, etc. It is used to set up \mermap/mapdef/attribution print afterwards and it is accessible as \mrcmapattritionprint (use read-only).

\mermap/supply/basename=\langle setup tile base name \rangle

Prefix for local tile files, e.g. 'tiles/map' for 'tiles/map_10_10_10.png'.
/mermap/supply/flex reference scale=\{scale denominator\} (no default)

With the given \{scale denominator\}, an appropriate /mermap/supply/zoom\textsuperscript{P.30} and /mermap/tile size\textsuperscript{P.51} is computed. Note that the \{scale denominator\} always applies to the current /mermap/supply/latitude\textsuperscript{P.31} and is used for map type boundaries and reference. For example, if the reference point is on the north side of the map, also the \{scale denominator\} applies to the most northern latitude.

Note to take special care to the order of the options.

- The reference point has to be set before /mermap/supply/flex reference scale, e.g. by /mermap/supply/latitude\textsuperscript{P.31}, /mermap/supply/position\textsuperscript{P.31}, /mermap/supply/named position\textsuperscript{P.31}.
- /mermap/supply/tex height\textsuperscript{P.31}, /mermap/supply/tex width\textsuperscript{P.31} (only for map type reference) have to be set after /mermap/supply/flex reference scale, because the /mermap/tile size\textsuperscript{P.51} is adapted.

Also see /mermap/flex tile size\textsuperscript{P.51}, /mermap/flex zoom\textsuperscript{P.51}, and /mermap/flex scale\textsuperscript{P.52}.

\begin{tikzpicture}
\mrcmap[type=reference,latitude=48.14,longitude=11.57, flex reference scale=250000, source=opentopomap, tex width=\linewidth,tex height=5cm]
\mrcdrawmap
\node[below,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south) {\mrcmapattribution};
\mrcclipmap
\path[draw] (mrcmap.south west) rectangle (mrcmap.north east);
\node[below left=2mm,align=right,fill=white,fill opacity=0.5, text opacity=1] at (mrcmap.north east) {scale \mrcprettymapscale};
\end{tikzpicture}

Kartendaten: © OpenStreetMap Mitwirkende, SRTM | Kartendarstellung: © OpenTopoMap (CC-BY-SA)

/mermap/supply/flex area scale=\{scale denominator\} (no default)

This is a shortcut for setting /mermap/supply/area to reference\textsuperscript{P.30} and /mermap/supply/flex reference scale=\{scale denominator\}. Used for map type boundaries and reference.

Note to take special care to the order of the options.

- The reference point has to be set before /mermap/supply/flex area scale.
- /mermap/supply/tex height\textsuperscript{P.31}, /mermap/supply/tex width\textsuperscript{P.31} (only for map type reference) have to be set after /mermap/supply/flex reference scale.
This key can be used for map type \textit{areafit} as final option after all other options. It applies a fine tuning to \texttt{/mermap/tile size}~\textsuperscript{P. 51}, \texttt{/mermap/supply/width}~\textsuperscript{P. 31}, and \texttt{/mermap/supply/height}~\textsuperscript{P. 31} such that the defined area fits exactly into the map region. If a \texttt{(size)} is specified, width and height are reduced for the calculation by this \texttt{(size)}, e.g. \texttt{(size)=1cm} ensures a border of 5mm on each side. Also see \texttt{/mermap/flex tile size}~\textsuperscript{P. 51} and \texttt{/mermap/flex zoom}~\textsuperscript{P. 51}.

\begin{tikzpicture}
\mrcNPdef{munich}{48.14}{11.58}
\mrcNPdef{rio}{-22.91}{-43.20}
\mrcNPdef{newyork}{40.71}{-74.01}
\mrcmap[ type = areafit, area = {munich,rio,newyork},
 source=topplusopen web grau,
 tex width=\texttt{\linewidth}, tex height=7cm,
 flex area fit=1cm ]{}
\mrcdrawmap
\node[below,font=\texttt{\fontsize{7pt}{7pt}\sffamily}] at (mrcmap.south) \{ \mrcmapattribution \} ;
\draw[yellow] ([xshift=5mm,yshift=5mm]mrcmap.south west) rectangle ([xshift=-5mm,yshift=-5mm]mrcmap.north east);
\draw[red,fill=red!50!gray!50!white,fill opacity=0.25] (\mrcNPcs{newyork}) -- (\mrcNPcs{rio}) -- (\mrcNPcs{munich}) -- cycle;
\end{tikzpicture}
/mermap/supply/pixel=(setup pixel size) (no default, initially 256)

Pixel width (and height) of a tile. It is especially needed for target mergedmap and also wmsmap. For wmsmap, it is multiplied with a pseudo tile calculation to compute the actual picture size to download.

/mermap/supply/dpi=(dpi value) (style, no default)

This style sets /mermap/supply/pixel such that the given (dpi value) is resulting (approximately).

- Note that this only applies for WMS Servers and not for TMS (Tile Map Service) Servers, because there /mermap/supply/pixel is a fixed number depending on Server settings and cannot be chosen arbitrarily.
- A high (dpi value) results in large downloaded map files. If the server does not provide a high resolution map, you will get unnecessary large files with blurred content.
- The (dpi value) for TMS data can be changed by adapting /mermap/tile size\(^*\text{P.51}\) or /mermap/flex tile size\(^*\text{P.51}\).

\begin{tikzpicture}
\mrcmap[type=reference,latitude=48.14,longitude=11.57, flex reference scale=250000, source=topplusopen p250, target=wmsmap, dpi=300, tex width=\linewidth, tex height=5cm]{dpi_value}
\mrcdrawmap
\node[below,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south) {\mrcmapattribution};
\mrcclipmap\path[draw] (mrcmap.south west) rectangle (mrcmap.north east);
\end{tikzpicture}

4.4 Map Apply

\mrcapplymap{(definition)}

A map which is supplied by \mrcsupplymap\(^*\text{P.29}\) is applied inside a tikzpicture environment by \mrcapplymap where (definition) identifies the map. \mrcapplymap replaces a manual setup by \mrcdefinemap\(^*\text{P.18}\). The same map can be applied more than once inside a document. Note that applying a map does not mean to draw the map, but to prepare everything for drawing.

\mrcmap{(options)}{(definition)}

This is a combination of \mrcsupplymap\(^*\text{P.29}\) with the given (options) followed immediately by \mrcapplymap. If (definition) is left empty, an automated unique identifier is inserted. If a map is to be used just once, \mrcmap may be preferred.

© Bundesamt für Kartographie und Geodäsie 2020, Datenquellen
4.5 Map Tile Server

To use map tiles with this package you obviously need access to a map tile server. Thanks to all the many contributors to OpenStreetMap, map data is free for everyone to use. But, map tile servers based on OpenStreetMap are not necessarily free.

A list of online raster tile servers based on OpenStreetMap data is found here: https://wiki.openstreetmap.org/wiki/Tile_servers

- I do not run a map tile server.
- I do not and cannot grant any permission to access a map tile server.
- I do not and cannot grant any permission to use map tiles in private, academic, free, or commercial publications.
- All operators of map tile servers require to mention an attribution for their maps.

The following option allows easy usage of very few selected tile servers. The tile server of OpenStreetMap is not included because of its Tile Usage Policy. I am aware that the following list could be enlarged much more, but I do not want to add more to avoid any legal uncertainties.

- I will remove an entry immediately, if the tile server operator asks for it.
- If YOU operate a tile server and you want an entry here, I would be glad to add it to the following list.

/mermap/supply/source=(source) (style, no default)

This style sets /mermap/supply/url \cite{P.32}, /mermap/supply/attribution \cite{P.32}, /mermap/supply/attribution print \cite{P.32} and /mermap/supply/basename \cite{P.32}. Feasible values for (source) are:

- **dummy**: Dummy tile server at loopback 127.0.0.1 for test purposes.
- **opentopomap**: Tile server (TMS) of OpenTopoMap.
  - Usage (German language): https://opentopomap.org/about#verwendung
  - Required attribution (\mrcmapattribution):
    Kartendaten: © OpenStreetMap Mitwirkende, SRTM | Kartendarstellung: © OpenTopoMap (CC-BY-SA)
  - Required attribution for media without hyperlinks (\mrcmapattributionprint):
    Kartendaten: © www.openstreetmap.org/copyright, SRTM | Kartendarstellung: © opentopomap.org (CC-BY-SA)
• **stamen terrain:**
Tile server of Stamen Design.
Usage: http://maps.stamen.com/#terrain
  - Required attribution (**\mrcmapattribution**):
    Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
  - Required attribution for media without hyperlinks (**\mrcmapattributionprint**):
    Map tiles by Stamen Design (stamen.com), under CC BY 3.0. Data by OpenStreetMap, under ODbL (www.openstreetmap.org/copyright).

• **stamen terrain-background:**
Tile server of Stamen Design.
Usage: http://maps.stamen.com/#terrain-background
  - Required attribution (**\mrcmapattribution**):
    Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
  - Required attribution for media without hyperlinks (**\mrcmapattributionprint**):
    Map tiles by Stamen Design (stamen.com), under CC BY 3.0. Data by OpenStreetMap, under ODbL (www.openstreetmap.org/copyright).

• **stamen terrain-labels:**
Tile server of Stamen Design.
Usage: http://maps.stamen.com/#terrain-labels
  - Required attribution (**\mrcmapattribution**):
    Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
  - Required attribution for media without hyperlinks (**\mrcmapattributionprint**):
    Map tiles by Stamen Design (stamen.com), under CC BY 3.0. Data by OpenStreetMap, under ODbL (www.openstreetmap.org/copyright).
• **stamen terrain-lines:**
  Tile server of Stamen Design.
  Usage: http://maps.stamen.com/#terrain-lines
  – Required attribution (\mrcmapattribution):
    Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
  – Required attribution for media without hyperlinks (\mrcmapattributionprint):
    Map tiles by Stamen Design (stamen.com), under CC BY 3.0. Data by OpenStreetMap, under ODbL (www.openstreetmap.org/copyright).

• **stamen toner:**
  Tile server of Stamen Design.
  Usage: http://maps.stamen.com/#toner
  – Required attribution (\mrcmapattribution):
    Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
  – Required attribution for media without hyperlinks (\mrcmapattributionprint):
    Map tiles by Stamen Design (stamen.com), under CC BY 3.0. Data by OpenStreetMap, under ODbL (www.openstreetmap.org/copyright).

• **stamen toner-lite:**
  Tile server of Stamen Design.
  Usage: http://maps.stamen.com/#toner-lite
  – Required attribution (\mrcmapattribution):
    Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
  – Required attribution for media without hyperlinks (\mrcmapattributionprint):
    Map tiles by Stamen Design (stamen.com), under CC BY 3.0. Data by OpenStreetMap, under ODbL (www.openstreetmap.org/copyright).
• **stamen toner-hybrid:**
  Tile server of Stamen Design.
  Usage: http://maps.stamen.com/#toner-hybrid
  - Required attribution (**\mrcmapattribution**):
    Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
  - Required attribution for media without hyperlinks (**\mrcmapattributionprint**):
    Map tiles by Stamen Design (stamen.com), under CC BY 3.0. Data by OpenStreetMap, under ODbL (www.openstreetmap.org/copyright).

• **stamen toner-background:**
  Tile server of Stamen Design.
  Usage: http://maps.stamen.com/#toner-background
  - Required attribution (**\mrcmapattribution**):
    Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
  - Required attribution for media without hyperlinks (**\mrcmapattributionprint**):
    Map tiles by Stamen Design (stamen.com), under CC BY 3.0. Data by OpenStreetMap, under ODbL (www.openstreetmap.org/copyright).

• **stamen toner-labels:**
  Tile server of Stamen Design.
  Usage: http://maps.stamen.com/#toner-labels
  - Required attribution (**\mrcmapattribution**):
    Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
  - Required attribution for media without hyperlinks (**\mrcmapattributionprint**):
    Map tiles by Stamen Design (stamen.com), under CC BY 3.0. Data by OpenStreetMap, under ODbL (www.openstreetmap.org/copyright).
• **stamen toner-lines:**
  Tile server of Stamen Design.
  Usage: [http://maps.stamen.com/#toner-lines](http://maps.stamen.com/#toner-lines)
  
  - Required attribution (**\mrcmapattribution**):
    Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
  
  - Required attribution for media without hyperlinks (**\mrcmapattributionprint**):
    Map tiles by Stamen Design (stamen.com), under CC BY 3.0. Data by OpenStreetMap, under ODbL (www.openstreetmap.org/copyright).

• **stamen watercolor:**
  Tile server of Stamen Design.
  Usage: [http://maps.stamen.com/#watercolor](http://maps.stamen.com/#watercolor)
  
  - Required attribution (**\mrcmapattribution**):
    Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under CC BY SA.
  
  - Required attribution for media without hyperlinks (**\mrcmapattributionprint**):
    Map tiles by Stamen Design (stamen.com), under CC BY 3.0. Data by OpenStreetMap, under CC BY SA.

• **thunderforest opencyclemap:**
  Tile server of Thunderforest.
  A registered ⟨api-key⟩ is needed (free plan available) which is applied by **\mrcsetapikey**.
  
  - Usage: [https://www.thunderforest.com/terms](https://www.thunderforest.com/terms)
  - API documentation: [https://www.thunderforest.com/maps/opencyclemap](https://www.thunderforest.com/maps/opencyclemap)
  
  - Required attribution (**\mrcmapattribution**):
    Maps © Thunderforest, Data © OpenStreetMap contributors
  
  - Required attribution for media without hyperlinks (**\mrcmapattributionprint**):
    Maps © www.thunderforest.com, Data © www.osm.org/copyright
• **thunderforest transport:**
  Tile server of Thunderforest.
  A registered ⟨api-key⟩ is needed (free plan available) which is applied by \mrcsetapikey \P. 32 \{thunderforest\}{⟨api-key⟩}.
  – Usage: https://www.thunderforest.com/terms
  – API documentation: https://www.thunderforest.com/maps/transport
  – Required attribution (\mrcmapattribution):
    Maps © Thunderforest, Data © OpenStreetMap contributors
  – Required attribution for media without hyperlinks (\mrcmapattributionprint):
    Maps © www.thunderforest.com, Data © www.osm.org/copyright

• **thunderforest landscape:**
  Tile server of Thunderforest.
  A registered ⟨api-key⟩ is needed (free plan available) which is applied by \mrcsetapikey \P. 32 \{thunderforest\}{⟨api-key⟩}.
  – Usage: https://www.thunderforest.com/terms
  – API documentation: https://www.thunderforest.com/maps/landscape
  – Required attribution (\mrcmapattribution):
    Maps © Thunderforest, Data © OpenStreetMap contributors
  – Required attribution for media without hyperlinks (\mrcmapattributionprint):
    Maps © www.thunderforest.com, Data © www.osm.org/copyright

• **thunderforest outdoors:**
  Tile server of Thunderforest.
  A registered ⟨api-key⟩ is needed (free plan available) which is applied by \mrcsetapikey \P. 32 \{thunderforest\}{⟨api-key⟩}.
  – Usage: https://www.thunderforest.com/terms
  – API documentation: https://www.thunderforest.com/maps/outdoors
  – Required attribution (\mrcmapattribution):
    Maps © Thunderforest, Data © OpenStreetMap contributors
  – Required attribution for media without hyperlinks (\mrcmapattributionprint):
    Maps © www.thunderforest.com, Data © www.osm.org/copyright
• **thunderforest transport-dark:**
  Tile server of Thunderforest.
  A registered ⟨api-key⟩ is needed (free plan available) which is applied by \mrcsetapikey\ P. 32 \{thunderforest\}(⟨api-key⟩).
  – Usage: https://www.thunderforest.com/terms
  – API documentation: https://www.thunderforest.com/maps/transport-dark
  – Required attribution (\mrcmapattribution):
    Maps © Thunderforest, Data © OpenStreetMap contributors
  – Required attribution for media without hyperlinks (\mrcmapattributionprint):
    Maps © www.thunderforest.com, Data © www.osm.org/copyright

• **thunderforest spinal-map:**
  Tile server of Thunderforest.
  A registered ⟨api-key⟩ is needed (free plan available) which is applied by \mrcsetapikey\ P. 32 \{thunderforest\}(⟨api-key⟩).
  – Usage: https://www.thunderforest.com/terms
  – API documentation: https://www.thunderforest.com/maps/spinal-map
  – Required attribution (\mrcmapattribution):
    Maps © Thunderforest, Data © OpenStreetMap contributors
  – Required attribution for media without hyperlinks (\mrcmapattributionprint):
    Maps © www.thunderforest.com, Data © www.osm.org/copyright

• **thunderforest pioneer:**
  Tile server of Thunderforest.
  A registered ⟨api-key⟩ is needed (free plan available) which is applied by \mrcsetapikey\ P. 32 \{thunderforest\}(⟨api-key⟩).
  – Usage: https://www.thunderforest.com/terms
  – API documentation: https://www.thunderforest.com/maps/pioneer
  – Required attribution (\mrcmapattribution):
    Maps © Thunderforest, Data © OpenStreetMap contributors
  – Required attribution for media without hyperlinks (\mrcmapattributionprint):
    Maps © www.thunderforest.com, Data © www.osm.org/copyright
• **thunderforest mobile-atlas:**
  Tile server of Thunderforest.
  A registered \(\text{api-key}\) is needed (free plan available) which is applied by \mrcsetapikey\(\text{\{api-key\}}\).
  - Usage: https://www.thunderforest.com/terms
  - API documentation: https://www.thunderforest.com/maps/mobile-atlas
  - Required attribution (\mrcmapattribution):
    Maps © Thunderforest, Data © OpenStreetMap contributors
  - Required attribution for media without hyperlinks (\mrcmapattributionprint):
    Maps © www.thunderforest.com, Data © www.osm.org/copyright

• **thunderforest neighbourhood:**
  Tile server of Thunderforest.
  A registered \(\text{api-key}\) is needed (free plan available) which is applied by \mrcsetapikey\(\text{\{api-key\}}\).
  - Usage: https://www.thunderforest.com/terms
  - API documentation: https://www.thunderforest.com/maps/neighbourhood
  - Required attribution (\mrcmapattribution):
    Maps © Thunderforest, Data © OpenStreetMap contributors
  - Required attribution for media without hyperlinks (\mrcmapattributionprint):
    Maps © www.thunderforest.com, Data © www.osm.org/copyright
- **topplusopen web:**
  Tile server (TMS) of Bundesamt für Kartographie und Geodäsie.
  - Covers world / Europe / Germany depending on zoom level
  - Required attribution (**\mrcmaptribution\**):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen
  - Required attribution for media without hyperlinks (**\mrcmaptributionprint\**):

- **topplusopen web grau:**
  Tile server (TMS) of Bundesamt für Kartographie und Geodäsie.
  - Covers world / Europe / Germany depending on zoom level
  - Required attribution (**\mrcmaptribution\**):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen
  - Required attribution for media without hyperlinks (**\mrcmaptributionprint\**):
The following sources are Web map server (WMS). Therefore, `mermap/supply/target` can only be set to `wmsmap` or `none`.

- **topplusopen p5**: 
  Web map server (WMS) of Bundesamt für Kartographie und Geodäsie. 
  - 1:5000, covers Europe / Germany depending on zoom level 
  - Required attribution (`\mrcmapattribution`): 
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen 
  - Required attribution for media without hyperlinks (`\mrcmapattributionprint`): 

- **topplusopen p5 grau**: 
  Web map server (WMS) of Bundesamt für Kartographie und Geodäsie. 
  - 1:5000, covers Europe / Germany depending on zoom level 
  - Required attribution (`\mrcmapattribution`): 
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen 
  - Required attribution for media without hyperlinks (`\mrcmapattributionprint`): 

- **topplusopen p10**: 
  Web map server (WMS) of Bundesamt für Kartographie und Geodäsie. 
  - 1:10000, covers Europe / Germany depending on zoom level 
  - Required attribution (`\mrcmapattribution`): 
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen 
  - Required attribution for media without hyperlinks (`\mrcmapattributionprint`): 
- **topplusopen p10 grau:**
  Web map server (WMS) of Bundesamt für Kartographie und Geodäsie.
  - 1:10000, covers Europe / Germany depending on zoom level
  - Required attribution (\mrcmaptribution):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen
  - Required attribution for media without hyperlinks (\mrcmaptributionprint):

- **topplusopen p17.5:**
  Web map server (WMS) of Bundesamt für Kartographie und Geodäsie.
  - 1:17500, covers Europe / Germany depending on zoom level
  - Required attribution (\mrcmaptribution):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen
  - Required attribution for media without hyperlinks (\mrcmaptributionprint):

- **topplusopen p17.5 grau:**
  Web map server (WMS) of Bundesamt für Kartographie und Geodäsie.
  - 1:17500, covers Europe / Germany depending on zoom level
  - Required attribution (\mrcmaptribution):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen
  - Required attribution for media without hyperlinks (\mrcmaptributionprint):
• **topplusopen p25:**
Web map server (WMS) of Bundesamt für Kartographie und Geodäsie.
  - 1:25000, covers Europe / Germany depending on zoom level
  - Required attribution (\mrcmapattribution):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen
  - Required attribution for media without hyperlinks (\mrcmapattributionprint):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen:

• **topplusopen p25 grau:**
Web map server (WMS) of Bundesamt für Kartographie und Geodäsie.
  - 1:25000, covers Europe / Germany depending on zoom level
  - Required attribution (\mrcmapattribution):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen
  - Required attribution for media without hyperlinks (\mrcmapattributionprint):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen:

• **topplusopen p50:**
Web map server (WMS) of Bundesamt für Kartographie und Geodäsie.
  - 1:50000, covers Europe / Germany depending on zoom level
  - Required attribution (\mrcmapattribution):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen
  - Required attribution for media without hyperlinks (\mrcmapattributionprint):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen:
- **topplusopen p50 grau:**
  Web map server (WMS) of Bundesamt für Kartographie und Geodäsie.
  - 1:50000, covers Europe / Germany depending on zoom level
  - Required attribution (mrcmapattribution):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen
  - Required attribution for media without hyperlinks (mrcmapattributionprint):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen:

- **topplusopen p100:**
  Web map server (WMS) of Bundesamt für Kartographie und Geodäsie.
  - 1:100000, covers Europe / Germany depending on zoom level
  - Required attribution (mrcmapattribution):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen
  - Required attribution for media without hyperlinks (mrcmapattributionprint):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen:

- **topplusopen p100 grau:**
  Web map server (WMS) of Bundesamt für Kartographie und Geodäsie.
  - 1:100000, covers Europe / Germany depending on zoom level
  - Required attribution (mrcmapattribution):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen
  - Required attribution for media without hyperlinks (mrcmapattributionprint):
    © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen:
• **topplusopen p250:**
  Web map server (WMS) of Bundesamt für Kartographie und Geodäsie.
  - 1:250000, covers Europe / Germany depending on zoom level
  - Required attribution (`\mrcmapattribution`): © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen

• **topplusopen p250 grau:**
  Web map server (WMS) of Bundesamt für Kartographie und Geodäsie.
  - 1:250000, covers Europe / Germany depending on zoom level
  - Required attribution (`\mrcmapattribution`): © Bundesamt für Kartographie und Geodäsie 2020, Datenquellen

\mrcnewsupplysource{⟨source⟩}{⟨options⟩}

Adds a new ⟨source⟩ value to `/mermap/supply/source` P. 36. The ⟨options⟩ should set the keys `/mermap/supply/url` P. 32, `/mermap/supply/attribute` P. 32, `/mermap/supply/attribute print` P. 32 and `/mermap/supply/basename` P. 32. It is recommend to use x ... for ⟨source⟩ to avoid conflicts with future official additions to `/mermap/supply/source` P. 36.

\mrcnewsupplysource{x example}{}

```{mermap}
url = http://127.0.0.1/dummy/{z}/{x}/{y}.png,
attribute = {Dummy tile server},
attribute print = {Dummy tile server},
basename = tiles/dummy,
```
5 Map Drawing

5.1 Principal Drawing

\mrcreademap[⟨options⟩]

Inside a \texttt{tikzpicture} environment, \mrcreademap draws a map prepared by \mrcredefinemap \textsuperscript{P.18}, \mrcreapplymap \textsuperscript{P.35}, or \mrcreapemap \textsuperscript{P.35}. All \langle options \rangle share the common prefix /mermap/. This is the principal macro to draw a prepared map respectively the background of the map. The background consists of downloaded map tiles or just a color rectangle.

/mermap/\texttt{draw}=⟨tile draw⟩ (no default, initially \texttt{auto})

- \texttt{auto}: Draws the map according to /mermap/\texttt{mapdef/resource} \textsuperscript{P.19}, i.e. downloaded maps or tiles are used, if available.
- \texttt{path}: Draws the map according to the style given by /mermap/\texttt{map path}. Existing map tiles or merged maps are ignored
- \texttt{tiles}: Draws the map with downloaded map tiles, if available.
- \texttt{mergedmap}: Draws the map with a merged picture, if available.
- \texttt{wmsmap}: Draws the map with a downloaded WMS picture, if available.

/mermap/\texttt{map path}=⟨options⟩ (no default, initially upper left=\texttt{green!50}, upper right=\texttt{green!25}, lower left=\texttt{green!50!black!50}, lower right=\texttt{green!25})

Defines a Ti\textit{k}Z style for drawing the map without tiles. \langle options \rangle are feasible Ti\textit{k}Z path options.

/mermap/\texttt{map clip}=⟨code⟩ (no default, initially \texttt{\mrcclipmap})

Clipping options for the map. By default, the defined map is clipped with the full map rectangle. Use this option only, if you not want to clip the map to its specified size. \langle code \rangle is some Ti\textit{k}Z clipping code.

/mermap/\texttt{map scope}=⟨options⟩ (no default, initially empty)

\mrcreademap uses a \texttt{scope} environment inside which takes the given Ti\textit{k}Z \langle options \rangle.

\mrcclipmap

Clips all subsequent drawings against the applied map.
This is a shortcut macro identical to

\texttt{\path[clip] (mrcmap.south west) rectangle (mrcmap.north east);} 

\mrcboundmap

Sets the picture bounding box according to the applied map.
This is a shortcut macro identical to

\texttt{\path[use as bounding box] (mrcmap.south west) rectangle (mrcmap.north east);}
5.2 Flexible Tile Size

Typically, the pixel size of a map tile is fixed and a map tile is a pixel graphics file. The actual size of such an included picture inside the document is freely selectable. Note that a very small /mermap/tile size results in very small map lettering, while a very large /mermap/tile size results in very blurred images.

The general idea of a flexible tile size is to specify an aspired tile size called /mermap/flex tile size and to give \LaTeX{} the freedom to select /mermap/tile size in about the same size as /mermap/flex tile size. This freedom is used to achieve a pseudo zoom called /mermap/flex zoom which is a nearly arbitrary rational number instead of /mermap/supply/zoom which is a natural number.

This pseudo zoom is applied by several options which share flex in their names, e.g. /mermap/flex scale, /mermap/named flex scale, /mermap/supply/flex reference scale, /mermap/supply/flex area scale, /mermap/supply/flex area fit.

/mermap/tile size=⟨length⟩ (no default, initially 32.512mm)

Width and height of a drawn tile picture are set to ⟨length⟩. For standard tiles with 256 times 256 pixels a tile size of 32.512 mm = 1.28 in results in an approximate 200 dpi output for the document. For a beamer document, consider to use a /mermap/tile size of 21.333 333 mm to get approximate 1:1 pixel input and output (depending on beamer settings and used hardware). Also see /mermap/mapdef/tile size.

/mermap/flex tile size=⟨length⟩ (no default, initially 32.512mm)

Aspired width and height of a tile picture are set to ⟨length⟩. This value is used while applying /mermap/flex zoom.

/mermap/flex zoom=⟨pseudo zoom⟩ (style, no default)

This style sets /mermap/supply/zoom and /mermap/tile size in combination.

- If ⟨pseudo zoom⟩ is a natural number, /mermap/supply/zoom is set to ⟨pseudo zoom⟩ and /mermap/tile size is set to /mermap/flex tile size.
- Otherwise, /mermap/supply/zoom is set to the natural number closest to ⟨pseudo zoom⟩ and /mermap/tile size is such enlarged or reduced that the ⟨pseudo zoom⟩ value is simulated, i.e. the impression of a rational zoom factor is given.

Note that /mermap/flex zoom has to be used before \mr_supplymap or \mr_map, because the zoom setup is adapted.
/mermap/flex scale=⟨scale denominator⟩:⟨latitude⟩ (style, no default)

For different latitude scopes, an identical zoom factor produces maps of different scale. With /mermap/flex scale, a /mermap/flex zoom\textsuperscript{P.51} is computed to achieve the given ⟨scale denominator⟩ at a given ⟨latitude⟩. Note that this only applies to the center of a map. If the produced map is not centered at ⟨latitude⟩, the produced scale may differ from the intended one. Also see /mermap/supply/flex reference scale\textsuperscript{P.33}.

\begin{tikzpicture}
\mermapset{flex scale=250000:48.14}
\mrcmap[type=reference,latitude=48.14,longitude=11.57, 
source=opentopomap, 
tex width=\linewidth,tex height=5cm}{}
\mrcdrawmap 
\node[below,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south) 
{\mrcmapattribution};
\mrcclipmap 
\path[draw] (mrcmap.south west) rectangle (mrcmap.north east); 
\node[below left=2mm, align=right, fill=white, fill opacity=0.5, 
text opacity=1] 
at (mrcmap.north east) {scale \mrcprettymapscale};
\end{tikzpicture}

/mermap/named flex scale=⟨scale denominator⟩:⟨name⟩ (style, no default)

Identical to /mermap/flex scale, but used the named position ⟨name⟩ to provide a latitude, see Section 3.5 on page 22.

\mrcNPdef{munich}{48.137222}{11.575556}
\mermapset{named flex scale=250000:munich} % identical to the following
\mermapset{flex scale=250000:\mrcNPlat{munich}}
5.3 Geodetic Network

\mrcdrawnetwork[(options)]

Draws a geodetic network with meridians and parallels. All (options) share the common prefix /mermap/. The displayed lines are selected automatically according to some tuning parameters. The map is sliced in about maximal /mermap/network pieces in each direction. Meridians and parallels share a minimal distance of about /mermap/network distance. The algorithm is allowed to violate these conditions somewhat.

Note that oversized maps are not supported, i.e. maps which are wider than 360° in longitude. Here, meridians are expected to be missing or misplaced.

\% \mrcsetapikey{thunderforest}{YOUR-API-KEY} \% registered key
\begin{tikzpicture}
\mrcmap[ type = boundaries,
west = -20, east = 40, south = 36, north = 65,
source=thunderforest landscape,
flex area scale=40 000 000 ]{}
\mrcdrawmap
\node[below,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south)
{\mrcmapattribution};
\mrcclipmap
\mrcdrawnetwork
\path[pattern=mesh] (mrcmap.south west) rectangle (mrcmap.north east);
\end{tikzpicture}

/mermap/network pieces=(number) (no default, initially 8)

The map is sliced in about maximal (number) pieces in each direction. (number) may be exceeded somewhat. It is underrun to comply with /mermap/network distance.

/mermap/network distance=(mesh width) (no default, initially 2cm)

Meridians and parallels share a minimal distance of about (mesh width). (mesh width) may be underrun somewhat. It is exceeded to comply with /mermap/network pieces. For parallels on small scale maps, it refers to an averaged mesh width.

/mermap/network font=(text) (no default, initially \fontsize{4pt}{4pt}\sffamily)

(text) is some font setting for the latitude and longitude display.
5.4 Graphical Debug Overlay

\mrcdrawinfo

Draws some map information overlay for debugging purposes only.

\begin{tikzpicture}
\mrcNPdef{munich}{48.137222}{11.575556}
\mrcmap[type=reference, named position=munich, source=opentopomap, flex reference scale=50000, tex width=\linewidth, tex height=10cm]{}
\mrcdrawmap
\node[below,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south) {\mrcmapattribution};
\mrcclipmap
\mrcdrawinfo
\end{tikzpicture}
6 Scales and Sizes

Inside a defined map several size values and scaling options are available. Please note that due to the nature of the Mercator projection and several simplifying assumptions all specifications for map scale, map width, map height, etc. are imprecise in the best case and even misleading in the worst case. They are suited for representative diagrams, but not for critical navigation purposes etc.

6.1 Map Sizes and Document Sizes

\mrctexwidth
\TeX length denoting the document width of the current map.

\mrctexheight
\TeX length denoting the document width of the current map.

\mrcscale
Scaling factor between map and real world. A \TeX length given in \texttt{pt}, but stripped from that unit, multiplied by \mrcscale corresponds to a real world length given in kilometers. Note that this is \textit{not} the map scale. Actually, it is reciprocal proportional to the map scale and proportional to the map scale denominator, see \mrcmapscaledenominator.

\mrctextokm{⟨length⟩}
Computes a given \TeX ⟨length⟩ (with unit) into the corresponding real world length in kilometers (without unit).

\mrctextomile{⟨length⟩}
Computes a given \TeX ⟨length⟩ (with unit) into the corresponding real world length in miles (without unit).

\mrckmtotex{⟨number⟩}
Computes real world length ⟨number⟩ in kilometers (without unit) to a \TeX length (with unit).

% draw a circle with radius 20km
\draw (mrcpos) circle (\mrckmtotex{20});

\mrcmiletotex{⟨number⟩}
Computes real world length ⟨number⟩ in miles (without unit) to a \TeX length (with unit).

% draw a circle with radius 20 miles
\draw (mrcpos) circle (\mrcmiletotex{20});

\mrcmapscaledenominator
Approximate map scale denominator. 1 cm on the map corresponds approximately to \mrcmapscaledenominator \cdot 1 cm in the real world. Do not confuse with \mrcscale.
6.2 Pretty Size Output

\mrcprettymapscale

Approximate map scale given with three valid digits with a representation like 1:1000.

Map scale: \mrcprettymapscale

\mrcprettymapwidth

Approximate map width in kilometers (or meters) with three valid digits:

Map width: \mrcprettymapwidth

To create a pretty printing to your own liking, you can do like the following:

\newcommand{\myprettymapwidth}{\SI[round-mode=figures,round-precision=3]\{\mrctextokm{\mrctexwidth}\}\{km\}}

\mrcprettymapheight

Approximate map height in kilometers (or meters) with three valid digits:

Map height: \mrcprettymapheight

\mrcprettymapresolution

Approximate map resolution in dpi (dots per inch):

Map resolution: \mrcprettymapresolution

\mrcprettytilesizetx

Approximate tile size inside the document (\TeX{} size) in millimeters:

$\TeX{}$ tile size: \mrcprettytilesizetx
6.3 Scale Bars

\mrcdrawscalebar\((\text{options})\)

Draws a scale bar according to the given \textit{options}. All \textit{options} share the common prefix \texttt{/mermap/scalebar/}. The most essential option is the \textit{width} of the scale bar.

\begin{tikzpicture}
\mrcNPdef{munich}{48.137222}{11.575556}
\mrcNPdef{vienna}{48.208333}{16.373056}
\mrcmap[text width=\linewidth, tex height=4cm, target=none, type=area fit, area={munich,vienna}, flex area fit=2cm]{}
\mrcdrawmap
\mrcmarker[named position=munich, contents={M"unchen}]
\mrcmarker[named position=vienna, contents={Wien}]
\mrcdrawscalebar[width-in-km=100, solid, at={([xshift=-10mm,yshift=5mm]mrcmap.south east)}, placement={above left}, ]
\end{tikzpicture}

The size, position, and appearance of the scale bar can be customized by setting the various \textit{options}. The shape of the scale bar is denoted by a Ti\textsc{k}Z node \texttt{mrcscalebar} which can be used for lettering.

\begin{tikzpicture}
\mrcNPdef{munich}{48.137222}{11.575556}
\mrcNPdef{vienna}{48.208333}{16.373056}
\mrcmap[text width=\linewidth, tex height=4cm, target=none, type=area fit, area={munich,vienna}, flex area fit=2cm]{}
\mrcdrawmap
\mrcmarker[named position=munich, contents={M"unchen}]
\mrcmarker[named position=vienna, contents={Wien}]
\mrcdrawscalebar[width-in-km=100,solid,south-west-inside=10mm;3mm ]
\path[every node/.style={above,inner sep=0.5mm,font=\sffamily\tiny}](mrcscalebar.north west) -- (mrcscalebar.north east)
node[pos=0]{} node[pos=0.2]{20} node[pos=0.4]{40} node[pos=0.6]{60}
node[pos=0.8]{80} node[pos=1]{100} node[pos=1, right,yshift=-1mm]{km};
\end{tikzpicture}
Sets the width of the scale bar to match the real word length \langle \textit{number} \rangle in kilometers.

\textbf{/mermap/scalebar/width-in-kilometer=\langle \textit{number} \rangle}  
(no default, initially 0)

Sets the width of the scale bar to match the real word length \langle \textit{number} \rangle in meters.

\textbf{/mermap/scalebar/width-in-meter=\langle \textit{number} \rangle}  
(no default, initially 0)

Sets the width of the scale bar to match the real word length \langle \textit{number} \rangle in miles.

\textbf{/mermap/scalebar/width-in-mile=\langle \textit{number} \rangle}  
(no default, initially 0)

Sets the width of the scale bar to match the real word length \langle \textit{number} \rangle in yards.

\textbf{/mermap/scalebar/partitions=\langle \textit{number} \rangle}  
(no default, initially 5)

Determines the \langle \textit{number} \rangle of partitions for the scale bar. If \langle \textit{number} \rangle is set to 1, there is no partitioning.

\textbf{/mermap/scalebar/height=\langle \textit{length} \rangle}  
(no default, initially 2mm)

Sets the height of the scale bar to the given \TeX \langle \textit{length} \rangle.

\textbf{/mermap/scalebar/at=\{\langle \textit{TikZ coordinate} \rangle \}}  
(no default, initially \{(0,0)\})

The scale bar is positioned at the given \langle \textit{TikZ coordinate} \rangle. The placement is done with the \textbf{/mermap/scalebar/placement} \footnote{P.\ 60} option. Both option correspond to the \textit{TikZ} options for positioning nodes. The scale bar can be positioned outside the map (e.g. below), but remember to use \texttt{\textbackslash mrcclipmap} \footnote{P.\ 50} after the scale bar in this case, if needed.

\begin{tikzpicture}
\mrcNPdef{munich}{48.137222}{11.575556}
\mrcNPdef{vienna}{48.208333}{16.373056}
\mrcmap[tex width=\linewidth, tex height=4cm, target=none, type=areafit, area={munich,vienna}, flex area fit=2cm]{}
\mrcdrawmap
\mrcmarker[named position=munich, contents=\{M"unchen\}]
\mrcmarker[named position=vienna, contents=\{Wien\}]
\mrcdrawscalebar[width-in-km=200, partitions=8, at={([xshift=5mm,yshift=-2mm]mrcmap.south west)}, placement=below right]
\path[every node/.style={below, inner sep=0.5mm, font=\sffamily\tiny}]
(mrcscalebar.south west) -- (mrcscalebar.south east)
node[pos=0]{0} node[ pos=0.25]{50} node[ pos=0.5]{100} node[ pos=0.75]{150}
node[ pos=1]{200} node[ pos=1, right, yshift=1mm]{km};
\end{tikzpicture}
/mermap/scalebar/placement=(TikZ positioning) (no default, initially empty)

(TikZ positioning) of a scale bar in combination with /mermap/scalebar/at. All TikZ placement options for nodes can be used, e.g. above left or anchor=mid west, etc. Actually, any node option could be applied here, but the intended use is for placement options only.

/mermap/scalebar/south-east-inside=(x shift);(y shift) (default 0pt;0pt, initially unset)
Shortcut for placing the scale bar at the south east corner of the map. The optional (x shift) and (y shift) denote the absolute shift values in each direction, i.e. the algebraic sign is automatically complemented. If only (x shift) is given, then (y shift) is set to the same value. /mermap/scalebar/at and /mermap/scalebar/placement are set by this option.

/mermap/scalebar/south-east-outside=(x shift);(y shift) (default 0pt;0pt, initially unset)
Shortcut for placing the scale bar below the south east corner of the map.

/mermap/scalebar/south-west-inside=(x shift);(y shift) (default 0pt;0pt, initially unset)
Shortcut for placing the scale bar at the south west corner of the map.

/mermap/scalebar/south-west-outside=(x shift);(y shift) (default 0pt;0pt, initially unset)
Shortcut for placing the scale bar below the south west corner of the map.

/mermap/scalebar/north-west-inside=(x shift);(y shift) (default 0pt;0pt, initially unset)
Shortcut for placing the scale bar at the north west corner of the map.

/mermap/scalebar/north-west-outside=(x shift);(y shift) (default 0pt;0pt, initially unset)
Shortcut for placing the scale bar above the north west corner of the map.

/mermap/scalebar/north-east-inside=(x shift);(y shift) (default 0pt;0pt, initially unset)
Shortcut for placing the scale bar at the north east corner of the map.

/mermap/scalebar/north-east-outside=(x shift);(y shift) (default 0pt;0pt, initially unset)
Shortcut for placing the scale bar above the north east corner of the map.

/mermap/scalebar/major style=⟨TikZ options⟩ (no default, initially empty)
The major part of the scale bar is a single TikZ path object which can be customized by the given ⟨TikZ options⟩. The major part consists of the black area in the default case.

\begin{tikzpicture}
\mrcdrawscalebar[scale=2000000, width-in-km=100, major style={left color=red,right color=blue} ]
\end{tikzpicture}

/mermap/scalebar/minor style=⟨TikZ options⟩ (no default, initially empty)
The minor part of the scale bar is a single TikZ path object which can be customized by the given ⟨TikZ options⟩. The minor part is seen as holes in the default case. /mermap/scalebar/minor style has only an effect, if the minor part is drawn /mermap/scalebar/solid.

\begin{tikzpicture}
\mrcdrawscalebar[scale=2000000, width-in-km=100, solid, minor style={yellow} ]
\end{tikzpicture}
/mermap/scalebar/double=true|false  (default true, initially true)
If set to true, the scale bar is drawn as a double ruler.

/mermap/scalebar/single=true|false  (default true, initially false)
If set to true, the scale bar is drawn as a single ruler. /mermap/scalebar/single is inverse to /mermap/scalebar/double.

\begin{tikzpicture}
  \mrcdrawscalebar[\text{scale}=2000000, \text{width-in-km}=100, \text{single}, \text{height}=1mm]
\end{tikzpicture}

/mermap/scalebar/transparent=true|false  (default true, initially true)
If set to true, the minor part of the scale bar is drawn transparent, i.e. as holes inside the ruler.

/mermap/scalebar/solid=true|false  (default true, initially false)
If set to true, the minor part of the scale bar is drawn opaque. It is drawn white or according to /mermap/scalebar/minor_style. /mermap/scalebar/solid is inverse to /mermap/scalebar/transparent.

/mermap/scalebar/scale=⟨scale denominator⟩  (no default, initially unset)
Sets or overwrites the ⟨scale denominator⟩ setting. Using this key is not needed and may even lead to erroneous displays inside a tikzpicture with a defined map setting. This key is helpful, if a scale bar is used without a defined map.
7 Markers

As described before, a map can be amended by arbitrary TikZ code using map coordinates. For highlighting places or adding markers, the `{\mrcmarker}` macro may be helpful which provides some predefined marker types.

7.1 Marker Settings

`{\mrcmarker}{⟨options⟩}`

Places a marker according to the given ⟨options⟩ on the map. All ⟨options⟩ share the common prefix `/mermap/marker/`. Different `/mermap/marker/type`\textsuperscript{P.66} settings are available which are more or less customizable.

```latex
% \mrcsetapikey{thunderforest}{YOUR-API-KEY} % registered key
\begin{tikzpicture}
  \mrcmap[type=reference, position=48.15:11.6, flex reference scale=500000, source=thunderforest neighbourhood, tex width=\linewidth, tex height=4cm]{}/
  \mrcdrawmap\node[below,font=\fontsize{7pt}{7pt}\sffamily\footnotesize] at (mrcmap.south) {\mrcmapattribution};
  \mrcdrawnetwork\mermapsetmarker{draw=red, fill=red!20!white, font=\sffamily\footnotesize}\mrcmarker[type=classic, position=48.15:11.2, contents={A}, radius=0.5mm}\mrcmarker[type=pin, position=48.15:11.3, contents={B}]\mrcmarker[type=pinflip, position=48.15:11.4, contents={C}]\mrcmarker[type=drop, position=48.15:11.5, contents={D}]\mrcmarker[type=knob, position=48.15:11.6, contents={E}]\mrcmarker[type=pictodropring, position=48.15:11.7, contents={F}]\mrcmarker[type=pictoknobring, position=48.15:11.8, contents={G}]\mrcmarker[type=ringx, position=48.15:11.9, contents={H}]\mrcmarker[type=markx, position=48.15:12, contents={I}]\end{tikzpicture}
```

Maps © Thunderforest, Data © OpenStreetMap contributors

`{\mermapsetmarker}{⟨options⟩}`

Sets ⟨options⟩ for all following markers inside the current \TeX{} group. All options share the common prefix `/mermap/marker/`, e.g. for setting `/mermap/marker/type`\textsuperscript{P.66} use

`{\mermapsetmarker}{type=pin}`

Also see `{\mermapset}`\textsuperscript{P.17} and `{\mermapsetsupply}`\textsuperscript{P.29}.
The given list of \emph{options} is used inside every \texttt{\mrcmarker-P.62} before the options of \texttt{\mrcmarker-P.62}.

The given list of \emph{options} is used inside every \texttt{\mrcmarker-P.62} after the options of \texttt{\mrcmarker-P.62}.

Latitude degree of the place marker. It is accessible as \texttt{\mrcmarkerlatitude} (use read-only).

Alias for /mermap/marker/latitude.

Longitude degree of the place marker. It is accessible as \texttt{\mrcmarkerlongitude} (use read-only).

Alias for /mermap/marker/longitude.

Sets the latitude degree and the longitude degree of the place marker.

Sets the latitude degree and the longitude degree of the place marker to the named position denoted by \emph{name}, see Section 3.5 on page 22.

The place marker is used or ignored according to its belonging inside the given \emph{area}.

Feasible values for \emph{area} are:
- \texttt{map}: Use inside map.
- \texttt{vicinity}: Use inside map plus vicinity, see /mermap/vicinity \texttt{P.23}.

Sets \emph{text} for displaying inside the marker, if the marker type supports such a thing. It is accessible as \texttt{\mrcmarkercontents} (use read-only).

Sets Ti\LaTeX \emph{code} for displaying inside the marker, if the marker type supports such a thing. It is accessible as \texttt{\mrcmarkerpictocontents} (use read-only).

The position of the marker is available as Ti\LaTeX coordinate by the given \emph{text}, e.g. to draw to or from the marker.

Sets a \emph{uuid} for unique identification of markers. It is accessible as \texttt{\mrcmarkeruuid} (use read-only) and \texttt{l_mermap_marker_uuid_tl} (use read-only). The \emph{uuid} is provided for user applications.

Sets \emph{text} as generic content for user applications. It is accessible as \texttt{\mrcmarkergeneric} (use read-only) and \texttt{l_mermap_marker_generic_tl} (use read-only).

Sets a \emph{category} to group markers. It is accessible as \texttt{\mrcmarkercategory} (use read-only).
/mermap/marker/show=true|false (default true, initially true)

If set to true, the marker is shown, if it lies inside the map (or vicinity). Otherwise, the place marker is not used.

/mermap/marker/hide=true|false (default true, initially false)

If set to true, the marker is not used. /mermap/marker/hide is inverse to /mermap/marker/show.

/mermap/marker/show category=⟨category⟩ (style, no default)

Sets /mermap/marker/show to true, if /mermap/marker/category = P.63 equals ⟨category⟩. Otherwise, nothing happens. /mermap/marker/category = P.63 has to be set before.

/mermap/marker/hide category=⟨category⟩ (style, no default)

Sets /mermap/marker/show to true, if /mermap/marker/category = P.63 equals ⟨category⟩. Otherwise, nothing happens. /mermap/marker/category = P.63 has to be set before.

/mermap/marker/hide all but category=⟨category⟩ (style, no default)

Sets /mermap/marker/show to false, if /mermap/marker/category = P.63 equals ⟨category⟩. Otherwise, nothing happens. /mermap/marker/category = P.63 has to be set before.

/mermap/marker/hide all but category=⟨category⟩ (style, no default)

Sets /mermap/marker/show to false, if /mermap/marker/category = P.63 does not equal ⟨category⟩. Otherwise, nothing happens. /mermap/marker/category = P.63 has to be set before.

\begin{tikzpicture}
\mrcmap[tex width=4cm, tex height=4cm, 
        latitude=48.14, longitude=11.57, target=none\{]
\mrcdrawmap[draw=path]
\mermapsetmarker{last options={hide all but category=A}}
\mrcmarker{type=knob,position=48.00:11.43,fill=blue,category=A}
\mrcmarker{type=knob,position=48.28:11.43,fill=red,category=B}
\mrcmarker{type=knob,position=48.00:11.71,fill=blue,category=A}
\mrcmarker{type=knob,position=48.28:11.71,fill=red,category=B}
\end{tikzpicture}
The following options require the package `hyperref` to be loaded. If `hyperref` is not loaded, 
`/mermap/marker/url` and `/mermap/marker/link` are ignored.

Note that if `/mermap/marker/url` and `/mermap/marker/link` are applied concurrently, then
`/mermap/marker/link` will win.

N 2020-05-04 /mermap/marker/url=(URL)  
(no default, initially empty)

Sets ⟨URL⟩ as external link of the marker. The package is required for this option. For an
own marker type, it is applied as TikZ option `mrchyperpath`.

\begin{tikzpicture}
\mrcNPdef{unibwm}{48.0802826}{11.6381048}
\mrcmap[type=reference, named position=unibwm, 
flex reference scale=100000, 
source=topplusopen web grau, 
tex width=\linewidth, tex height=4cm]{}
\mrcdrawmap 
\mrcclipmap 
\mrcmarker{ named position = unibwm, 
type = pin, 
draw = red, 
fill = red!20!white, 
url = {https://www.unibw.de}, 
contents = {Universität der Bundeswehr München},
}
\end{tikzpicture}

N 2020-05-04 /mermap/marker/link=(name)  
(no default, initially empty)

Sets an internal link of the marker to a document target with the given ⟨name⟩. The package `hyperref` is required for this option. For an own marker type, it is applied as TikZ option `mrchyperpath`.

N 2020-05-04 /mermap/marker/use urls=true|false  
(default true, initially true)

N 2020-05-04 /mermap/marker/ignore urls=true|false  
(default true, initially false)

If `/mermap/marker/use urls` is set to false, external links with `/mermap/marker/url` are ignored.

`/mermap/marker/use urls` is inverse to `/mermap/marker/use urls`.

N 2020-05-04 /mermap/marker/use links=true|false  
(default true, initially true)

N 2020-05-04 /mermap/marker/ignore links=true|false  
(default true, initially false)

If `/mermap/marker/use links` is set to false, internal links with `/mermap/marker/link` are ignored.

`/mermap/marker/ignore links` is inverse to `/mermap/marker/use links`.

65
7.2 Marker Types

/mermap/marker/type=(type)  (no default, initially classic)

Decides about the basic shape and style of the marker. Feasible values for (type) are listed in the following. More values can be defined by \mrcnewmarkertype \textsuperscript{P. 72}.

- **classic**:

```latex
\begin{tikzpicture}
\mrcmap[\textwidth=8cm, \textheight=4cm, zoom=10, 
            latitude=48.14, longitude=11.48, target=none]{}
\mrcdrawmap[draw=path]
\mermapsetmarker{type=classic}
\mrcmarker{position=48.14:11.36, contents={Germering}}
\mermapsetmarker{type=classic, draw=blue, fill=blue!20!white, 
                 text=blue, font=\sffamily\footnotesize}
\mrcmarker{position=48.14:11.57, contents={M"unchen}, alias={munich}, angle=-30}
\end{tikzpicture}
```

![Diagram of classic marker type with examples of Germering and München]

- **pin**:

```latex
\begin{tikzpicture}
\mrcmap[\textwidth=8cm, \textheight=4cm, zoom=10, 
            latitude=48.14, longitude=11.48, target=none]{}
\mrcdrawmap[draw=path]
\mermapsetmarker{type=pin}
\mrcmarker{position=48.14:11.36, contents={Germering}}
\mermapsetmarker{type=pin, draw=blue, fill=blue!20!white, 
                 font=\sffamily\footnotesize}
\mrcmarker{position=48.14:11.57, contents={M"unchen}, alias={munich}}
\end{tikzpicture}
```

![Diagram of pin marker type with examples of Germering and München]
• **pinflip:**

```latex
\begin{tikzpicture}
  \mrcmap[text width=8cm, tex height=4cm, zoom=10, latent=48.14, longitude=11.48, target=none]
  \mrcdrawmap[draw=path]
  \mermapsetmarker{type=pinflip}
  \mrcmarker{position=48.14:11.36, contents={Germering}}
  \mermapsetmarker{type=pinflip, draw=red, fill=red!20!white, font=\sffamily\footnotesize}
  \mrcmarker{position=48.14:11.57, contents={M"unchen}, shift=4mm}
\end{tikzpicture}
```

![Image of map with pinflips for Germering and München]

• **drop:**

```latex
\begin{tikzpicture}
  \mrcmap[text width=8cm, tex height=4cm, zoom=10, latent=48.14, longitude=11.48, target=none]
  \mrcdrawmap[draw=path]
  \mermapsetmarker{type=drop}
  \mrcmarker{position=48.14:11.36, contents={Germering}}
  \mermapsetmarker{type=drop, fill=blue, draw=blue!20!white, text=white, font=\sffamily\small\bfseries, path style={line join=round, thin, draw=mrcmarkerfill, double=mrcmarkerdraw, double distance=0.6pt} }
  \mrcmarker{position=48.14:11.57, contents={M}, alias={munich}}
\end{tikzpicture}
```

![Image of map with drops for Germering and München]
- **pictodrop**:

\begin{tikzpicture}
\mrcmap[\text width=8cm, \text height=4cm, zoom=10,\n latitude=48.14, longitude=11.48, target=none]{}
\mrcdrawmap[\text=\text]{\draw[\text=\text]{
\mermapsetmarker{type=pictodrop, pictocontents={
\draw[\text=\text]{(-0.2,-0.2)--(0.2,-0.2)--(0.2,0.1)--(0,0.2)--(-0.2,0.1) -- cycle;
}}\mrcmarker{position=48.14:11.36, contents={Germering}}\mrcmarker{position=48.14:11.57, pictocontents={\node {\includegraphics[width=6mm]{alertmessage-warning.png}};}}\end{tikzpicture}

- **pictodropring**:

\begin{tikzpicture}
\mrcmap[\text width=8cm, \text height=4cm, zoom=10,\n latitude=48.14, longitude=11.48, target=none]{}
\mrcdrawmap[\text=\text]{\draw[\text=\text]{\mermapsetmarker{type=pictodropring}\mrcmarker{position=48.14:11.36, contents={Germering}}\mrcmarker{position=48.14:11.57, pictocontents={\node {\includegraphics[width=6mm]{alertmessage-warning.png}};}}\end{tikzpicture}
- **knob:**

```latex
\begin{tikzpicture}
\mrcmap[tex width=8cm, tex height=4cm, zoom=10, 
latitude=48.14, longitude=11.48, target=none]{}
\mrcdrawmap[draw=path]
\mermapsetmarker{type=knob}
\mermapsetmarker{position=48.14:11.36, contents={Germering}}
\mermapsetmarker{type=knob, radius=2mm, 
fill=blue, draw=blue!20!white, 
text=white, path style={draw=mrcmarkerfill, double=mrcmarkerdraw, 
double distance=0.6pt} }
\mermapsetmarker{position=48.14:11.57, contents={M}}
\end{tikzpicture}
```

- **pictoknob:**

```latex
\begin{tikzpicture}
\mrcmap[tex width=8cm, tex height=4cm, zoom=10, 
latitude=48.14, longitude=11.48, target=none]{}
\mrcdrawmap[draw=path]
\mermapsetmarker{type=pictoknob}
\mermapsetmarker{position=48.14:11.36, contents={Germering}}
\mermapsetmarker{type=pictoknob, radius=2mm, 
fill=blue, draw=blue!20!white, 
path style={draw=mrcmarkerfill, double=mrcmarkerdraw, 
double distance=0.6pt} }
\mermapsetmarker{position=48.14:11.57, alias={munich}}
\end{tikzpicture}
```
- **pictoknobring:**

```latex
\begin{tikzpicture}
\mrcmap[draw=path,\textwidth=8cm,\textheight=4cm,\zoom=10,
    \latitude=48.14,\longitude=11.48,\target=none]\{
\mermapsetmarker[type=pictoknobring]\{
\mer marker[position=48.14:11.36,contents={Germering}]\{
\mermapsetmarker[type=pictoknobring,radius=2mm,inner radius=1mm,draw=blue,fill=blue!20!white,pictocontents={\node[above] at (0,\mrcmarkerradius)\
{\mrcmarkercontents};}}\}
\mermarker[position=48.14:11.57,contents={M"unchen}]\end{tikzpicture}
```

- **ringx:**

```latex
\begin{tikzpicture}
\mrcmap[draw=path,\textwidth=8cm,\textheight=4cm,\zoom=10,
    \latitude=48.14,\longitude=11.48,\target=none]\{
\mermapsetmarker[type=ringx]\{
\mermarker[position=48.14:11.36,contents={Germering}]\{
\mermapsetmarker[type=ringx,\radius=4mm,inner radius=3mm,draw=red,fill=red!50!white,path style={draw=mrcmarkerdraw,very thin,double=white,double distance=0.6pt}}\}
\mermarker[position=48.14:11.57,alias=munich]\end{tikzpicture}
```
The different marker types can be customized by some additional options:

\begin{itemize}
  \item /mermap/marker/font\(=\langle\text{font}\rangle\)  
    (no default, initially \text{sffamily\small})
    Font of the marker text (\text{classic, pin, pinflip, drop, knob}). It is accessible as \mrcmarkerfont (use read-only).
  \item /mermap/marker/text\(=\langle\text{color}\rangle\)  
    (no default, initially \text{black})
    Color of the marker text (\text{classic, pin, pinflip, drop, knob}). It is accessible as mrcmarkertext (use read-only).
  \item /mermap/marker/draw\(=\langle\text{color}\rangle\)  
    (no default, initially \text{gray})
    Color of the marker frame (\text{pin, pinflip, drop, pictodrop, pictodropring, knob, pictoknob, pictoknobring, ringx, markx}). It is accessible as mrcmarkerdraw (use read-only).
  \item /mermap/marker/fill\(=\langle\text{color}\rangle\)  
    (no default, initially \text{gray!20})
    Color of the marker interior (\text{pin, pinflip, drop, pictodrop, pictodropring, knob, pictoknob, pictoknobring, ringx, markx}). It is accessible as mrcmarkerfill (use read-only).
  \item /mermap/marker/angle\(=\langle\text{angle}\rangle\)  
    (no default, initially \text{90})
    Angle of the marker (\text{classic}). It is accessible as \mrcmarkerangle (use read-only).
  \item /mermap/marker/shift\(=\langle\text{length}\rangle\)  
    (no default, initially \text{0pt})
    Shift of the marker text (\text{pin, pinflip}). It also shifts the drop center (\text{drop, pictodrop, pictodropring}). It is accessible as \mrcmarkershift (use read-only).
  \item /mermap/marker/distance\(=\langle\text{length}\rangle\)  
    (no default, initially \text{5mm})
    Length of the marker needle (\text{pin, pinflip}). It is accessible as \mrcmarkerdistance (use read-only).
\end{itemize}
Radius of the marker (classic, drop, pictodrop, pictodropring, knob, pictoknob, pictoknobring, ringx, markx). It is accessible as \mrcmarkerradius (use read-only).

Inner radius of the marker (pictodropring, pictoknobring, ringx). It is accessible as \mrcmarkerinnerradius (use read-only).

Ti\kZ \langle options \rangle which are added to (some) path elements of of the marker (pin, pinflip, drop, pictodrop, pictodropring, knob, pictoknob, pictoknobring, ringx, markx). It is accessible as Ti\kZ option mrcpathstyle (use read-only).

Ti\kZ \langle options \rangle which are added to the node element of of the marker (classic, pin, pinflip, drop, knob). It is accessible as Ti\kZ option mrcnodestyle (use read-only).

7.3 New Marker Types

\mrcnewmarkertype{⟨type name⟩}{⟨Ti\kZ code⟩}

Creates a new /mermap/marker/type value ⟨type name⟩ using the given ⟨Ti\kZ code⟩ for drawing a place marker. To avoid future name clashes, you should start a private ⟨type name⟩ with letter x. For ⟨Ti\kZ code⟩ settings like \mrcmarkercontents, \mrcmarkerfont, or mrcmarkerrfill may be used or ignored.

\mrcnewmarkertype{xmark}{
\path[draw=mrcmarkerdraw,thick] circle (\mrcmarkerradius);
\path[draw=mrcmarkerdraw,thick]
  (45:2*\mrcmarkerradius)--(225:2*\mrcmarkerradius)
  (135:2*\mrcmarkerradius)--(315:2*\mrcmarkerradius);
}
\begin{tikzpicture}
\mrcmap[tex width=4cm, tex height=4cm, latitude=48.14, longitude=11.57, target=none]{}
\mrcdrawmap[draw=path]
\mrcmarker[type=xmark, position=48.14:11.57,alias=munich, draw=red]
\end{tikzpicture}
7.4 New Marker Styles

\mrcnewmarkerstyle\{\langle style name\rangle\}\{\langle options\rangle\}

Creates a new /mermap/marker/style value \langle style name\rangle using the given \langle options\rangle for
drawing a place marker. All \langle options\rangle share the common prefix /mermap/marker/. Here, a
\textit{style} has the same concept as a TikZ style.

\begin{tikzpicture}
\mrcmap\[\text{tex width=8cm, tex height=4cm, zoom=10,}
\text{latitude=48.14, longitude=11.48, target=none}\}\{\}
\mrcdrawmap\[\text{draw=path}\]
\mrcnewmarkerstyle\{\text{one}\}\{\text{type=pin, draw=blue, fill=blue!20!white,}
\text{font=\sffamily\footnotesize}\}
\mrcnewmarkerstyle\{\text{two}\}\{\text{type=pinflip, draw=red, fill=red!20!white,}
\text{font=\sffamily\footnotesize}\}
\mrcmarker\[\text{style=one, position=48.14:11.36, contents=\text{Germering}}\]
\mrcmarker\[\text{style=two, position=48.14:11.57, contents=\text{M"unchen}}\]
\end{tikzpicture}

A \textit{style} can have one parameter, but note the small difference in applying this parameter
compared to TikZ:

\begin{tikzpicture}
\mrcmap\[\text{tex width=8cm, tex height=4cm, zoom=10,}
\text{latitude=48.14, longitude=11.48, target=none}\]\{\}
\mrcdrawmap\[\text{draw=path}\]
\mrcnewmarkerstyle\{\text{color}\}\{\text{type=pin, draw=#1, fill=#1!20!white,}
\text{font=\sffamily\footnotesize}\}
\mrcmarker\[\text{style/color=blue, position=48.14:11.36, contents=\text{Germering}}\]
\mrcmarker\[\text{style/color=red, position=48.14:11.57, contents=\text{M"unchen}}\]
\end{tikzpicture}

/mermap/marker/style=\langle style name\rangle

(no default, initially unset)

Applies a given \langle style name\rangle, i.e. all options which were stored by \mrcnewmarkerstyle
under this name.
8 Routes

Routes are TikZ path objects which can be drawn with the appropriate TikZ macros and the coordinate system documented in Section 3 on page 17. Nevertheless, in the following some alternatives are described which allow to specify a path by a sequence of points (\texttt{\textbackslash mrcpoint}). The idea is that such a sequence of points is exported by a third-party application for inclusion in a \LaTeX map, see Section 8.2 on page 76.

8.1 Route Path Definition

\begin{mrcroute}(options)
  (environment content)
\end{mrcroute}

This environment creates a TikZ path with given TikZ (options). The (environment content) is a sequence of points made by \texttt{\textbackslash mrcpoint}.

\begin{verbatim}
\begin{mrcroute}[red, very thick]
  \mrcpoint{48.137222}{11.575556}
  \mrcpoint{49.019479}{12.0976942}
  ...
\end{mrcroute}
\end{verbatim}

Note that a map definition by \texttt{\textbackslash mrcdefinemap} \textsuperscript{P.18}, \texttt{\textbackslash mrcmap} \textsuperscript{P.35}, or \texttt{\textbackslash mrcapplymap} \textsuperscript{P.35} is needed before a route path can be drawn.

\begin{mrcroute*}(options)
  (environment content)
\end{mrcroute*}

Identical to \texttt{mrcroute}, but the created TikZ path is closed.

\begin{mrcrouteinput}(options)\{filename\}
  (environment content)
\end{mrcrouteinput}

Identical to \texttt{mrcroute}, but the sequence of points is included from a file with the given (filename).

\begin{mrcrouteinput*}(options)\{filename\}
  (environment content)
\end{mrcrouteinput*}

Identical to \texttt{mrcrouteinput}, but the created TikZ path is closed.

\texttt{\textbackslash mrcpoint}(latitude)(longitude)

Specifies a single coordinate point with given (latitude) and (longitude) as part of sequence inside \texttt{mrcroute}.

This is a wrapper for \texttt{\textbackslash pgfpathmoveto} respectively \texttt{\textbackslash pgfpathlineto}.

\texttt{/mermap/every route=(options)}

(no default, initially empty)

Sets TikZ (options) which are applied to every \texttt{mrcroute} and \texttt{mrcrouteinput}.

74
\begin{tikzpicture}
\mrcmap[type=areafit, west=5, east=15, south=47, north=55, source=thunderforest neighbourhood, tex width=14cm, tex height=14cm, flex area fit=5mm]
\mrcmap(\texttt{routes_example})
\mrcdrawmap
\node[below,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south) {\mrcmapattribution};
\mrcclipmap
\path[draw] (mrcmap.south west) rectangle (mrcmap.north east);
\begin{mrcroute}[blue,line width=0.4mm,line cap=round, line join=round,double=blue!5!white, double distance=0.4mm]
\mrcpoint{48.137222}{11.575556}
\mrcpoint{49.019479}{12.0976942}
\mrcpoint{49.45522}{11.07631}
\mrcpoint{50.978056}{11.029167}
\mrcpoint{52.518611}{13.408333}
\end{mrcroute}
\end{tikzpicture}
8.2 Example Python Conversion Scripts for gpX

The following scripts are examples for conversions from a standard routes and tracks of gpX files to \LaTeXX include files (assuming a single route/track per file).

```
import xmltodict

with open('route.gpx', encoding='utf-8') as gpx:
    doc = xmltodict.parse(gpx.read())
    with open('route.inc', 'w', encoding='utf-8') as inc:
        for rtept in doc['gpx']['rte']['rtept']:
            lat = rtept['@lat']
            lon = rtept['@lon']
            inc.write(f'mrcpoint{{{lat}}}{{{lon}}}n')
```

```
import xmltodict

with open('track.gpx', encoding='utf-8') as gpx:
    doc = xmltodict.parse(gpx.read())
    with open('track.inc', 'w', encoding='utf-8') as inc:
        for trkpt in doc['gpx']['trk']['trkseg']['trkpt']:
            lat = trkpt['@lat']
            lon = trkpt['@lon']
            inc.write(f'mrcpoint{{{lat}}}{{{lon}}}n')
```


9 Orthodromes and Loxodromes

A loxodrome is a curve which crosses all meridians with a constant angle. On Mercator maps, loxodromes are depicted as straight lines and can be drawn by simple TikZ path elements.

On a sphere, the shortest path from one point to another runs along an orthodrome where an orthodrome is a great-circle.

The mathematical background and further information are found in [1].

9.1 Orthodrome Drawing

\mrcdraworthodrome[(options)]{(lat1)}{(lon1)}{(lat2)}{(lon2)}

Draws an orthodrome curve from a point with latitude \(\langle \text{lat1} \rangle\) and longitude \(\langle \text{lon1} \rangle\) to a point with latitude \(\langle \text{lat2} \rangle\) and longitude \(\langle \text{lon2} \rangle\). This is a TikZ path object where \(\langle \text{options} \rangle\) are TikZ settings for this path. There are two orthodrome pieces connecting two positions (forming a great-circle). \mrcdraworthodrome does not necessarily choose the shorter one, see [1]. The drawn orthodrome is a spherical approximation instead of an ellipsoidal one.
\mrcNPdraworthodrome[(options)]{⟨name1⟩}{⟨name2⟩}

Identical to \mrcdraworthodrome^P.77, but the start and end point are described by named positions ⟨name1⟩ and ⟨name2⟩.

\begin{tikzpicture}
\mrcNPdef{munich}{48.14}{11.58}
\mrcNPdef{newyork}{40.71}{-74.01}
\mrcmap[type=areafit, area={munich,newyork},
  source=topplusopen web,
  tex width=\linewidth,tex height=6cm,]
\node at (mrcmap.south) {ortho_orthodrome2};
\mrcclipmap
\draw (mrcmap.south west) rectangle (mrcmap.north east);
\mrcmarker[type=pin,named position=munich,contents={M"unchen}]
\mrcmarker[type=pinflip,shift=5mm,named position=newyork,contents={New York City}]
\mrcNPdraworthodrome[red,very thick] {munich}{newyork}
\draw[blue,very thick] (\mrcNPcs{munich}) -- (\mrcNPcs{newyork});
\node[red,fill=white] at ([above=1.3cm]\mrcmap) {
  \mrcNPprettyorthodistance{munich}{newyork} 
};
\node[blue,fill=white] at ([below=5mm]\mrcmap) {
  \mrcNPprettyloxodistance{munich}{newyork} 
};
\end{tikzpicture}
9.2 Orthodromic and Loxodromic Distances

\texttt{\textbackslash mrcprettyorthodistance\{\langle lat1\rangle\}\{\langle lon1\rangle\}\{\langle lat2\rangle\}\{\langle lon2\rangle\}}

Approximate orthodromic distance between two points with latitude \langle lat1\rangle, longitude \langle lon1\rangle and latitude \langle lat2\rangle, longitude \langle lon2\rangle with three valid digits.

\texttt{\mrcprettyorthodistance\{48.14\}\{11.58\}\{40.71\}\{-74.01\}}

6490 km

\texttt{\textbackslash mrcNPprettyorthodistance\{\langle name1\rangle\}\{\langle name2\rangle\}}

Approximate orthodromic distance between two named positions \langle name1\rangle and \langle name2\rangle with three valid digits.

\texttt{\mrcNPdef\{munich\}\{48.14\}\{11.58\}}
\texttt{\mrcNPdef\{newyork\}\{40.71\}\{-74.01\}}
\texttt{\mrcNPprettyorthodistance\{munich\}\{newyork\}}

6490 km

\texttt{\textbackslash mrcstoreorthodistance\{\langle macro\rangle\}\{\langle lat1\rangle\}\{\langle lon1\rangle\}\{\langle lat2\rangle\}\{\langle lon2\rangle\}}

Stores the approximate orthodromic distance (in kilometers) between two points with latitude \langle lat1\rangle, longitude \langle lon1\rangle and latitude \langle lat2\rangle, longitude \langle lon2\rangle to a given \langle macro\rangle.

\texttt{\mrcstoreorthodistance\mydist\{48.14\}\{11.58\}\{40.71\}\{-74.01\}}
\texttt{\mydist}

6488.72407474055 km

\texttt{\textbackslash mrcprettyloxodistance\{\langle lat1\rangle\}\{\langle lon1\rangle\}\{\langle lat2\rangle\}\{\langle lon2\rangle\}}

Approximate loxodromic distance between two points with latitude \langle lat1\rangle, longitude \langle lon1\rangle and latitude \langle lat2\rangle, longitude \langle lon2\rangle with three valid digits.

\texttt{\mrcprettyloxodistance\{48.14\}\{11.58\}\{40.71\}\{-74.01\}}

6830 km

\texttt{\textbackslash mrcNPprettyloxodistance\{\langle name1\rangle\}\{\langle name2\rangle\}}

Approximate loxodromic distance between two named positions \langle name1\rangle and \langle name2\rangle with three valid digits.

\texttt{\mrcNPdef\{munich\}\{48.14\}\{11.58\}}
\texttt{\mrcNPdef\{newyork\}\{40.71\}\{-74.01\}}
\texttt{\mrcNPprettyloxodistance\{munich\}\{newyork\}}

6830 km

\texttt{\textbackslash mrcstoreloxodistance\{\langle macro\rangle\}\{\langle lat1\rangle\}\{\langle lon1\rangle\}\{\langle lat2\rangle\}\{\langle lon2\rangle\}}

Stores the approximate loxodromic distance (in kilometers) between two points with latitude \langle lat1\rangle, longitude \langle lon1\rangle and latitude \langle lat2\rangle, longitude \langle lon2\rangle to a given \langle macro\rangle.

\texttt{\mrcstoreloxodistance\mydist\{48.14\}\{11.58\}\{40.71\}\{-74.01\}}
\texttt{\mydist}

6833.046494937649 km
10 Limitations and Caveats

10.1 No Polar Regions

The standard Mercator projection is not suited for north or south polar regions. The Web Mercator projections only covers positions between 85.0511° S and 85.0511° N with map tiles. Equally, the mercatormap package does only provide support for this area.

If your map overlaps to south of 85.0511° S or to north of 85.0511° N, compiler errors are possible because of internal computation limitations. In any case, there are no map tiles.
10.2 International Dateline (180th Meridian)

If your map includes the 180th meridian (more or less the international dateline), remember that the eastern hemisphere lies west and the western hemisphere lies east for your map. Consider the following example displaying Vladivostok and Anchorage. Here, `/mermap/supply/west` is set to 130°E (130) and `/mermap/supply/east` is set to 149°W (211 instead of -149).

There is some automatic correction for positions to fit inside the defined map (see Anchorage in the example below), but west should be lower than east.
10.3 Very small Scale Maps

If a map has a very small scale, a coordinate point may appear more than once on the map. But, the map coordinates of this package will only pilot to a single point:

\begin{tikzpicture}
\mrcmap[type=reference,latitude=48.14,longitude=11.57,
source=opentopomap,\text width=\linewidth,\text height=4cm,zoom=1]{}
\mrcdrawmap
\node[below,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south)
{\mrcmapattribution} \mrcclipmap
\path[draw] (mrcmap.south west) rectangle (mrcmap.north east);
\mrcmarker{type=pin,position=61.22:-149.88,contents={Anchorage}}
\mrcmarker{type=pin,position=43.12:131.9,contents={Vladivostok}}
\end{tikzpicture}

You cannot trust too much in figures from \mrcprettymapscalet\textsuperscript{+P. 56}, \mrcprettymapwidtht\textsuperscript{+P. 56}, \mrcprettymapheightt\textsuperscript{+P. 56}, etc for large and medium scale maps. For small scale maps, these figures are worse and even misleading. Better do not use them:

\begin{tikzpicture}
\mrcmap[type=reference,latitude=48.14,longitude=11.57,\text width=\linewidth,\text height=6cm,zoom=2]{}
\mrcdrawmap
\node[below,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south)
{\mrcmapattribution}; \mrcclipmap \mrcdrawinfo
\end{tikzpicture}
10.4 Zoom is not Scale

Using the same zoom value for different latitudes can give completely different scales. The following example for Oslo and Rio de Janeiro at zoom 12 scales to 1:151000 and otherwise 1:277000.

Same example again, but with \texttt{\texttt{/mermap/\texttt{flex scale}}} \textsuperscript{P.52} which gives 1:150000 for both cities and comparable maps.
10.5 Distances

There are at least three kinds of distance measures between two points on the map:

- Measure the distance with a ruler on the printed (or displayed) map and multiply with the \texttt{\mrcmapscaledenominator} \textsuperscript{P.55}. Note that the map scale denominator is only (approximately) correct for the map center.

- Compute the loxodromic distance, e.g. with \texttt{\mrcstoreloxodistance} \textsuperscript{P.79} which gives the (approximated) distance following a loxodrome. On our Mercator map, this would be equal to the ruler method from above, if the map scale would be constant.

- Compute the orthodromic distance, e.g. with \texttt{\mrcstoreorthodistance} \textsuperscript{P.79} which gives the (approximated) shortest distance following an orthodrome.

Apart from the approximate nature of the implementation, these distances are expected to be quite identical for large scale maps, but not for (very) small scale maps.
\begin{tikzpicture}
\mrcNPdef{munich}{48.137222}{11.575556}
\mrcNPdef{fraunberg}{48.3685075}{11.9894209}
\mrcmap[type=reference, named position=munich,  
   flex reference scale=1000000,  
   source=topplusopen web,  
   tex width=\linewidth, tex height=10cm]{}
\mrcdrawmap
\node[below left,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south east)  
   {\mrcmapattribution};  
\node[below right,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south west)  
   {\mrcprettymapscale};  
\mrcclipmap  
\path[draw] (\mrcNPcs{munich}) circle (\mrckmtotex{40});  
\node[above,fill=white,opacity=0.85,text opacity=1] at ([yshift=\mrckmtotex{40}]\mrcNPcs{munich}) {\SI{40}{km}};  
\draw[blue,very thick] (\mrcNPcs{munich}) -- (\mrcNPcs{fraunberg});  
\mrcNPdraworthodrome[red,very thick] {munich}{fraunberg}  
\path (\mrcNPcs{munich}) --  
   node[sloped,above=3mm,red,fill=white,opacity=0.85,text opacity=1]  
   {\mrcNPprettyorthodistance{munich}{fraunberg}}  
   node[sloped,below=3mm,blue,fill=white,opacity=0.85,text opacity=1]  
   {\mrcNPprettyloxodistance{munich}{fraunberg}}  
(\mrcNPcs{fraunberg});  
\mrcmarker[type=pin,named position=munich,contents={M"unchen}]  
\mrcmarker[type=pin,named position=fraunberg,contents={Fraunberg}]  
\end{tikzpicture}
The next example gives notable different distances. The aberration is dependent from the center angle.

\begin{tikzpicture}
\mrcNPdef{munich}{48.137222}{11.575556}
\mrcNPdef{barcelona}{41.3947688}{2.0787285}
\mrcmap[type=reference, named position=munich, flex reference scale=25000000, source=topplusopen web, tex width=\linewidth, tex height=10cm]{}
\mrcdrawmap
\node[below left,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south east) {\mrcmapattribution};
\node[below right,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south west) {\mrcprettymapscale};
\mrcclipmap
\path\draw\[\mrckmtotex{1000}\] \mrcNPcs{munich} circle {\mrckmtotex{1000}};
\node\[\mrckmtotex{1000}\] \mrcNPcs{munich} \node\[\mrckmtotex{1000}\] \mrcNPcs{barcelona};
\mrcNPdraworthodrome\[\mrckmtotex{1000}\] \mrcNPdraworthodrome{munich}{barcelona};
\path\draw\[\mrckmtotex{1000}\] \mrcNPcs{munich} \node\[\mrckmtotex{1000}\] \mrcNPcs{barcelona} \node\[\mrckmtotex{1000}\] \mrcNPcs{barcelona};
\mrcmarker[type=pinnl, named position=munich, contents={M"unchen}]
\mrcmarker[type=pinnlflip, named position=barcelona, contents={Barcelona}]
\end{tikzpicture}
The following small scale example contains unacceptable deviations from the displayed circle radius of 3000 km. The aberration is highly dependent from the center angle. For such small scale maps, displaying a distance circle should be avoided.

\begin{tikzpicture}
  \mrcNPdef{munich}{48.137222}{11.575556}
  \mrcNPdef{neskaup}{65.1446431}{-13.7420082}
  \mrcNPdef{atyrau}{47.0971204}{51.866263}
  \mrcNPdef{dakhla}{24.2026691}{-15.4883971}
  \mrcmap[type=reference, named position=munich,
  flex reference scale=75000000,
  source=topplusopen web,
  tex width=\linewidth, tex height=10cm]{}
  \mrcdrawmap
  \node[below left,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south east)
  {\mrcmapattribution};
  \node[below right,font=\fontsize{7pt}{7pt}\sffamily] at (mrcmap.south west)
  {\mrcprettymapscale};
  \mrcclipmap
  \path[draw] (\mrcNPcs{munich}) circle (\mrckmtotex{3000});
  \node[above,fill=white,opacity=0.85,text opacity=1]
  at ([yshift=\mrckmtotex{3000}]\mrcNPcs{munich}) {\SI{3000}{km}};
  \draw[blue,very thick] (\mrcNPcs{munich}) -- (\mrcNPcs{neskaup});
  \mrcNPdraworthodrome[red,very thick] {munich}{neskaup}
  \path (\mrcNPcs{munich}) --
  node[sloped,above=3mm,red,fill=white,opacity=0.85,text opacity=1]
  {\mrcNPprettyorthodistance{munich}{neskaup}}
  \node[sloped,below=3mm,blue,fill=white,opacity=0.85,text opacity=1]
  {\mrcNPprettyloxodistance{munich}{neskaup}}
  (\mrcNPcs{neskaup});
  \draw[blue,very thick] (\mrcNPcs{munich}) -- (\mrcNPcs{atyrau});
  \mrcNPdraworthodrome[red,very thick] {munich}{atyrau}
  \path (\mrcNPcs{munich}) --
  node[sloped,above=3mm,red,fill=white,opacity=0.85,text opacity=1]
  {\mrcNPprettyorthodistance{munich}{atyrau}}
  \node[sloped,below=3mm,blue,fill=white,opacity=0.85,text opacity=1]
  {\mrcNPprettyloxodistance{munich}{atyrau}}
  (\mrcNPcs{atyrau});
  \draw[blue,very thick] (\mrcNPcs{munich}) -- (\mrcNPcs{dakhla});
  \mrcNPdraworthodrome[red,very thick] {munich}{dakhla}
  \path (\mrcNPcs{munich}) --
  node[sloped,above=3mm,red,fill=white,opacity=0.85,text opacity=1]
  {\mrcNPprettyorthodistance{munich}{dakhla}}
  \node[sloped,below=3mm,blue,fill=white,opacity=0.85,text opacity=1]
  {\mrcNPprettyloxodistance{munich}{dakhla}}
  (\mrcNPcs{dakhla});
  \mrcmarker[type=pinflip,named position=munich,contents={M"unchen})
  \mrcmarker[type=pin,named position=neskaup,contents={Neskaupsta\dh ur})
  \mrcmarker[type=pin,named position=atyrau,contents={Atyrau})
  \mrcmarker[type=pinflip,named position=dakhla,contents={Ad-Dakhla})
\end{tikzpicture}
11 Version History

v1.00 (2020/04/20)

- Initial public release.

v1.01 (2020/05/05)

- Sorting of index key entries fixed for the documentation.
- User messages for failed Python script calls are changed from warnings to errors to fail fast and to provide more information and hints about possible reasons.
- New general marker option `/mermap/marker/generic` with corresponding macros `\mrcmarkergeneric` and `l_mermap_marker_generic_tl`. Also, the marker uuid is made exp13 accessible as `l_mermap_marker_uuid_tl`.
- New hyper marker options `/mermap/marker/url` and `/mermap/marker/link` which allow to fit a map to a given external list of marker positions.
- New option `/mermap/fail on missing resource` to control compilation behavior for missing resource files.

v1.02 (2020/08/06)

- New options `/mermap/supply/area from marker input` and `/mermap/supply/add area from marker input` which allow to fit a map to a given external list of marker positions.
- New option `/mermap/fail on missing resource` to control compilation behavior for missing resource files.
- Openrouteservice has discontinued mapsurfer tiles since June 2020. Therefore, `/mermap/supply/source` = `openrouteservice mapsurfer` is deprecated now. It is not documented any more, but for some time it is still available for old documents with cached map tiles.
- New marker option `/mermap/marker/distance` with corresponding macro `\mrcmarkerdistance`. 
References


Index

add area key, 30
add area from marker input key, 30
alias key, 63
align key, 31
angle key, 71
area key, 30
area from marker input key, 30
area to reference key, 30
areafit value, 9, 28–31, 34
at key, 59
attribution key, 19, 32
attribution print key, 19, 32
auto value, 50
basename key, 19, 32
boundaries value, 10, 28–30, 33
category key, 63
center value, 31
classic value, 66, 71, 72
Colors
  mrcmarkerdraw, 71
  mrcmarkerfill, 71, 72
  mrcmarkertext, 71
contents key, 63
decimal value, 24
decimal-0 value, 24, 25
decimal-1 value, 24
decimal-2 value, 24
decimal-3 value, 25
decimal-4 value, 25
definition prefix key, 29
degree value, 25
distance key, 71
double key, 61
dpi key, 35
draw key, 50, 71
drop value, 67, 71, 72
dummy value, 36
east key, 18, 30
east value, 31
Environments
  mrcroute, 74
  mrcroute*, 74
every route key, 74
fail on missing resource key, 31
fill key, 71
first options key, 63
flex area fit key, 34
flex area scale key, 33
flex reference scale key, 33
flex scale key, 52
flex tile size key, 51
flex zoom key, 51
font key, 71
format angle key, 24
format east key, 26
format NEWS absolute key, 26
format NEWS numeric key, 26
format north key, 25
format south key, 25
format west key, 26
generic key, 63
height key, 31, 59
hide key, 64
hide all but category key, 64
hide category key, 64
\ifmrcinmap, 23
\ifmrcinvicinity, 23
\ifmrcNPinmap, 23
\ifmrcNPinvicinity, 23
ignore links key, 65
ignore urls key, 65
inner radius key, 72

Keys

/memmap/
  definition prefix, 29
draw, 50
every route, 74
fail on missing resource, 31
flex scale, 52
flex tile size, 51
flex zoom, 51
format angle, 24
format east, 26
format NEWS absolute, 26
format NEWS numeric, 26
format north, 25
format south, 25
format west, 26
map clip, 50
map path, 50
map scope, 50
named flex scale, 52
network distance, 53
network font, 53
network pieces, 53
python, 27
samples, 77
tile size, 51
vicinity, 23
/memmap/cs/
  lat, 21
  latitude, 20
  lon, 21
  longitude, 20
/memmap/mapdef/
east, 31
knob, 69, 71, 72
map, 63
mark, 71, 72
mergedmap, 19, 31, 35, 50
minute, 25
mrchyperpath, 65
mrclastpos, 23
mrcmap, 20
mrcnodestyle, 72
mrcpathstyle, 72
mrcpos, 23
mrcscalebar, 58
none, 19, 31, 45
north, 31
northeast, 31
northwest, 31
openrouteservice mapsurfer, 89
opentopomap, 36
path, 50
pictodrop, 68, 71, 72
pictodropring, 68, 71, 72
pictoknob, 69, 71, 72
pictoknobreng, 70–72
pin, 66, 71, 72
pinflip, 67, 71, 72
reference, 7, 28–31, 33
ringx, 70–72
second, 25
south, 31
southeast, 31
southwest, 31
stamen terrain, 37
stamen terrain-background, 37
stamen terrain-labels, 37
stamen terrain-lines, 38
stamen toner, 38
stamen toner-background, 39
stamen toner-hybrid, 39
stamen toner-labels, 39
stamen toner-lines, 40
stamen toner-lite, 38
stamen watercolor, 40
thunderforest, 40–43
thunderforest landscape, 41
thunderforest mobile-atlas, 43
thunderforest neighbourhood, 43
thunderforest opencyclemap, 40
thunderforest outdoors, 41
thunderforest pioneer, 42
thunderforest spinal-map, 42
thunderforest transport, 41
thunderforest transport-dark, 42
tiles, 19, 31, 50
topplusopen p10, 45
topplusopen p100, 48
topplusopen p100 grau, 48
topplusopen p10 grau, 46
topplusopen p17.5, 46
topplusopen p25, 47
topplusopen p250, 49
topplusopen p250 grau, 49
topplusopen p25 grau, 47
topplusopen p5, 45
topplusopen p50, 47
topplusopen p50 grau, 48
topplusopen p5 grau, 45
topplusopen web, 44
topplusopen web grau, 44
vicinity, 63
west, 31
wmsmap, 19, 31, 35, 45, 50
vicinity key, 23
vicinity value, 63
west key, 18, 30
west value, 31
westoffset key, 19
width key, 31
width-in-kilometer key, 59
width-in-km key, 59
width-in-meter key, 59
width-in-mile key, 59
width-in-yard key, 59
wmsmap value, 19, 31, 35, 45, 50
xmax key, 18
xmin key, 18
ymax key, 18
ymin key, 18
zoom key, 18, 30