zTool 接口文档

Eureka

总目录

ZTEX 宏集已独立实现了一个 ztool 宏包, 此宏包中包含原来已被废弃的 l3sys-shell 中的所有命令. 除此之外, ztool 提供了 box 操作, 文件 IO 以及基本图形绘制相关的函数. 在 ztool 的协助下, ZTEX 能够避免或减少命令行 -shell-escape 参数或其它相关宏包的调用 (如 robust-externalize 宏包).

1 l3sys-shell

本部分主要介绍 ztool 中实现的原始 l3sys-shell 宏包中的命令. 所以使用本部分的命令时需在编译 LaTeX 文档时启用 -shell-escape 参数, 否则此系列命令将不会执行任何操作.

WARNING: 请谨慎使用此部分的命令, 部分不当操作可能会导致一些无法挽救的后果.

\ztool_shell_escape:n
\ztool_shell_escape:e

Updated: 2024-12-05

\ztool_shell_escape:n {⟨command⟩}

当 -shell-escape 参数启用时, 此命令会在 shell 中执行 ⟨command⟩, 如果 -shell-escape 参数未启用, 此命令将不会执行任何操作.

\ztool_shell_mkdir:n
\ztool_shell_mkdir:e

Updated: 2024-12-05

\ztool_shell_mkdir:n {⟨dir⟩}

当 -shell-escape 参数启用时, 此命令会创建一个目录 ⟨dir⟩, 如果 -shell-escape 参数未启用, 此命令将不会执行任何操作.

\ztool_shell_cp:nn
\ztool_shell_cp:(ee|ne|en)

Updated: 2024-12-05

\ztool_shell_cp:nn {⟨source⟩}{⟨target⟩}

当 -shell-escape 参数启用时, 此命令将把文件 ⟨source⟩ 复制为文件 ⟨target⟩, 如果 -shell-escape 参数未启用, 此命令将不会执行任何操作.

\ztool_shell_mv:nn
\ztool_shell_mv:(ee|ne|en)

Updated: 2024-12-05

\ztool_shell_mv:nn {⟨source⟩}{⟨target⟩}

当 -shell-escape 参数启用时, 此命令将把文件 ⟨source⟩ 移动到目录 ⟨target⟩, 如果 -shell-escape 参数未启用, 此命令将不会执行任何操作.

\ztool_shell_rm:n
\ztool_shell_rm:e

Updated: 2024-12-05

\ztool_shell_rm:n {⟨file⟩}

当 -shell-escape 参数启用时, 此命令将删除文件 ⟨file⟩, 如果 -shell-escape 参数未启用, 此命令将不会执行任何操作.

\ztool_shell_rmdir:n
\ztool_shell_rmdir:e

Updated: 2024-12-05

\ztool_shell_rmdir:n {⟨dir⟩}

当 -shell-escape 参数启用时, 此命令将删除目录 ⟨dir⟩, 如果 -shell-escape 参数未启用, 此命令将不会执行任何操作.

\ztool_get_shell_pwd:N
\ztool_get_shell_pwd:c

Updated: 2024-12-05

\ztool_get_shell_pwd:N ⟨*tl*⟩

当 -shell-escape 参数启用时, 此命令将返回当前的工作目录, 并将其存放在 ⟨*tl*⟩ 中, 如果 -shell-escape 参数未启用, 此命令将不会执行任何操作.

\ztool_shell_split_ls:nN

Updated: 2024-12-05

\ztool_shell_split_ls:nN {⟨*dir*⟩}⟨*tl*⟩

当 -shell-escape 参数启用时, 此命令将返回目录 ⟨*dir*⟩ 下的所有文件名, 并将其存放在 ⟨*tl*⟩ 中, 如果 -shell-escape 参数未启用, 此命令将不会执行任何操作.

2 File IO

本部分主要介绍 ztool 中实现的文件 IO 操作, 包括: 读取文件, 写入文件, 追加文件等操作. 本部分的系列命令均不需要启用 -shell-escape 参数.

\ztool_file_new:nn

Updated: 2024-12-05

`\ztool_file_new:nn {⟨boolean⟩}{⟨file⟩}`

此命令用于创建一个名为 ⟨*file*⟩ 的新文件, 如果 ⟨*file*⟩ 不存在, 则会创建一个名为 ⟨*file*⟩ 的新文件. 若文件已存在, 那么当 ⟨*boolean*⟩ 为 \c_true_bool 时, **会覆盖原文件**, 否则不会进行任何操作.

\ztool_read_file_as_seq:nnN
\ztool_read_file_as_seq:(neN|nnc|nec)

Updated: 2024-12-05

`\ztool_read_file_as_seq:nnN {⟨bool⟩}{⟨file⟩}⟨seq⟩`

此命令用于读取文件 ⟨*file*⟩ 的内容, 并将其存放在 ⟨*seq*⟩ 中, 如果 ⟨*file*⟩ 不存在, 则 ⟨*seq*⟩ 会被置为空. ⟨*bool*⟩ 用于控制是否保留行尾的空格, 可选值有:\c_true_-bool, \c_false_bool, 默认为 \c_true_bool; 如果 ⟨*bool*⟩ 为 \c_true_bool, 则保留行尾的空格, 否则不保留.

注意: 此命令仅在一个组内有效, 且不可嵌套.

\ztool_gread_file_as_seq:nnN
\ztool_gread_file_as_seq:(neN|nnc|nec)

Updated: 2025-01-05

`\ztool_read_file_as_seq:nnN {⟨bool⟩}{⟨file⟩}⟨seq⟩`

此命令用于读取文件 ⟨*file*⟩ 的内容, 并将其存放在 ⟨*seq*⟩ 中, 如果 ⟨*file*⟩ 不存在, 则 ⟨*seq*⟩ 会被置为空. ⟨*bool*⟩ 用于控制是否保留行尾的空格, 可选值有:\c_true_-bool, \c_false_bool, 默认为 \c_true_bool; 如果 ⟨*bool*⟩ 为 \c_true_bool, 则保留行尾的空格, 否则不保留.

注意: 此命令作用于全局, 不可嵌套.

\ztool_append_to_file:nn
\ztool_append_to_file:(no|nf|ee)

Updated: 2025-01-05

`\ztool_append_to_file:nn {⟨file⟩}{⟨content⟩}`

此命令用于将 ⟨*content*⟩ 追加到文件 ⟨*file*⟩ 中, 如果 ⟨*file*⟩ 不存在, 则会创建一个名为 ⟨*file*⟩ 的新文件, 并将 ⟨*content*⟩ 写入其中.

\ztool_replace_file_line:nnn
\ztool_replace_file_line:(enn|ene|eee)

Updated: 2025-01-05

`\ztool_replace_file_line:nnn {⟨file⟩}{⟨line⟩}{⟨content⟩}`

此命令用于将文件 ⟨*file*⟩ 中的第 ⟨*line*⟩ 行替换为 ⟨*content*⟩, 如果 ⟨*file*⟩ 不存在, 则不会进行任何操作.

`\ztool_insert_to_file:nnn`
`\ztool_insert_to_file:(nen|nfn|een)`

Updated: 2025-01-05

`\ztool_insert_to_file:nnn {⟨file⟩}{⟨line⟩}{⟨content⟩}`

此命令用于将 ⟨*content*⟩ 插入到文件 ⟨*file*⟩ 的第 ⟨*line*⟩ 行之前, 如果 ⟨*file*⟩ 不存在, 则不会进行任何操作.

下面一个示例展示了如何使用 ztool 中的几个文件 IO 操作命令:

```
\ExplSyntaxOn                                                              例 1
\ztool_file_new:nn {\c_true_bool}{testIO.txt}
\seq_new:N \l_ztool_tmp_seq \seq_clear:N \l_ztool_tmp_seq
\ztool_append_to_file:nn {testIO.txt} {|APPEND-CONTENT|}
\ztool_insert_to_file:nnn {testIO.txt} {2} {|~~INSERT-CONTENT~~|}
\ztool_gread_file_as_seq:nnN {\c_false_bool} {testIO.txt} \l_ztool_tmp_seq
\seq_use:Nn \l_ztool_tmp_seq {\par}
\ExplSyntaxOff
\inputminted{text}{testIO.txt}
```

|INSERT-CONTENT|

|APPEND-CONTENT|


```
| INSERT-CONTENT |
|APPEND-CONTENT|
```

3 盒子操作

本部分介绍 ztool 中实现的 Box 操作, 包括 box 的测量以及 box 的简单变换.

\ztool_get_ht:Nn
\ztool_get_ht:(Ne|ce)

Updated: 2024-12-05

\ztool_get_ht:Nn ⟨*dim*⟩{⟨*content*⟩}

此命令用于将 ⟨*content*⟩ 的高度保存在 ⟨*dim*⟩ 这一寄存器中.

\ztool_get_ht_plus_dp:Nn
\ztool_get_ht_plus_dp:(Ne|ce)

Updated: 2024-12-05

\ztool_get_ht:Nn ⟨*dim*⟩{⟨*content*⟩}

此命令用于将 ⟨*content*⟩ 的高度和深度的和保存在 ⟨*dim*⟩ 这一寄存器中.

\ztool_get_wd:Nn
\ztool_get_wd:(Ne|ce)

Updated: 2024-12-05

\ztool_get_wd:Nn ⟨*dim*⟩{⟨*content*⟩}

此命令用于将 ⟨*content*⟩ 的宽度保存在 ⟨*dim*⟩ 这一寄存器中.

\ztool_get_dp:Nn
\ztool_get_dp:(Ne|ce)

Updated: 2024-12-05

\ztool_get_dp:Nn ⟨*dim*⟩{⟨*content*⟩}

此命令用于将 ⟨*content*⟩ 的深度保存在 ⟨*dim*⟩ 这一寄存器中.

\ztool_gget_ht:Nn
\ztool_gget_ht:(Ne|ce)

Updated: 2024-12-05

\ztool_gget_ht:Nn ⟨*dim*⟩{⟨*content*⟩}

此命令用于将 ⟨*content*⟩ 的高度保存在 ⟨*dim*⟩ 这一寄存器中, 并且此操作是全局的.

\ztool_gget_wd:Nn
\ztool_gget_wd:(Ne|ce)

Updated: 2024-12-05

\ztool_gget_wd:Nn ⟨*dim*⟩{⟨*content*⟩}

此命令用于将 ⟨*content*⟩ 的宽度保存在 ⟨*dim*⟩ 这一寄存器中, 并且此操作是全局的.

\ztool_gget_dp:Nn
\ztool_gget_dp:(Ne|ce)

Updated: 2024-12-05

\ztool_gget_dp:nn ⟨*dim*⟩{⟨*content*⟩}

此命令用于将 ⟨*content*⟩ 的深度保存在 ⟨*dim*⟩ 这一寄存器中, 并且此操作是全局的.

\ztool_set_to_wd:nn
\ztool_set_to_wd:(en|ne)

Updated: 2024-12-05

\ztool_set_to_wd:nn {⟨*dim*⟩}{⟨*content*⟩}

此命令用于将 ⟨*content*⟩ 的宽度调整为 ⟨*dim*⟩, 然后排版出来.

\ztool_set_to_ht:nn
\ztool_set_to_ht:(en|ne)

Updated: 2024-12-05

\ztool_set_to_ht:nn {⟨*dim*⟩}{⟨*content*⟩}

此命令用于将 ⟨*content*⟩ 的高度调整为 ⟨*dim*⟩, 然后排版出来.

\ztool_autoset_to_wd_and_ht:nnn
\ztool_autoset_to_wd_and_ht:(nne|een|eee)

Updated: 2025-04-29

\ztool_autoset_to_wd_and_ht:nn {⟨width⟩}{⟨height⟩}{⟨content⟩}

此命令用于将 ⟨content⟩ 的宽度调整为 min(⟨width⟩, ⟨height⟩), 然后排版出来.

\ztool_rotate:nn
\ztool_rotate:(en|ne|ee)

New: 2025-04-29

\ztool_rotate:nn {⟨angle⟩}{⟨content⟩}

此命令用于将 ⟨content⟩ 旋转 ⟨angle⟩ 度, 然后排版出来.

\ztool_scale_to_wd:nn
\ztool_scale_to_wd:(en|ne|ee)

New: 2025-04-29

\ztool_scale_to_wd:nn {⟨dim⟩}{⟨content⟩}

此命令用于将 ⟨content⟩ 的宽度调整为 ⟨dim⟩, 但是不对盒子的高度做任何的调整, 然后排版出来.

\ztool_scale_to_ht:nn
\ztool_scale_to_ht:(en|ne|ee)

New: 2025-04-29

\ztool_scale_to_ht:nn {⟨dim⟩}{⟨content⟩}

此命令用于将 ⟨content⟩ 的高度 + 深度整体调整为 ⟨dim⟩, 但是不对盒子的宽度做任何的调整, 然后排版出来.

\ztool_scale_to_wd_and_ht:nnn
\ztool_scale_to_wd_and_ht:(nno|nne|eee)

New: 2025-04-29

\ztool_scale_to_wd_and_ht:nnn {⟨width⟩}{⟨height⟩}{⟨content⟩}

此命令用于将 ⟨content⟩ 的宽度调整为 ⟨width⟩, 高度 + 深度整体调整为 ⟨height⟩, 然后排版出来.

\ztool_box_item_align:Nnnn
\ztool_box_item_align:(cnnn|Nnno|cnno|Nnen|Nnee)

Updated: 2025-05-13

\ztool_box_item_align:Nnnn ⟨cmd⟩{⟨width⟩}{⟨content⟩}{⟨align⟩}

此命令用于将 ⟨content⟩ 的宽度调整为 ⟨width⟩, 然后排版出来, ⟨align⟩ 用于控制对齐方式, 可选值有: left, center, right, scatter. ⟨cmd⟩ 为一个命令, 其接受一个参数, 它将应用到 ⟨content⟩ 的每一个 Token 上. **注意**: 原 ⟨content⟩ 中的空格会被删除.

\ztool_fp_to_rad:n

New: 2025-05-12

\ztool_fp_to_rad:n {⟨angle⟩}

此命令用于将 ⟨angle⟩ 从弧度制转换为角度制.

`\ztool_affine_transformation:Nnnnn` `\ztool_affine_transformation:(Neeee|cnnnn|ceeee)`	`\ztool_affine_transformation:Nnnnn` $\langle box \rangle\{\langle a\rangle\}\{\langle b\rangle\}\{\langle c\rangle\}\{\langle d\rangle\}$

<div align="right">New: 2025-05-12</div>

此命令用于对 $\langle box \rangle$ 进行任意的仿射变换 (线性变换), 具体的使用方法可以参见前述的 `ztoolboxaffine` 命令; 上述参数对应的仿射变换矩阵 Λ 为

$$\Lambda = \begin{bmatrix} a & c \\ b & d \end{bmatrix}.$$

关于上述函数 `\ztool_affine_transformation:Nnnnn` 的一些技术细节: 给定任意一个仿射变换 Λ, 不妨设

$$\Lambda = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}.$$

我们可以做如下的分解 (与 SVD 分解类似), 令 $m = 2x$, 则有:

$$\begin{aligned}
\Lambda &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & m \\ 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \\
&= \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} \cos\omega & -\sin\omega \\ \sin\omega & \cos\omega \end{bmatrix} \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}.
\end{aligned} \quad (3.1)$$

我们给出如下的记号:

- $\mathbf{T}_1(\theta)$: 旋转矩阵, 绕原点逆时针旋转 θ 角;

- $\mathbf{T}_2(x)$: 缩放矩阵, 把 x 轴方向的所有向量变为原来的 x 倍;

- $\mathbf{T}_3(y)$: 缩放矩阵, 把 y 轴方向的所有向量变为原来的 y 倍;

那么我们可以认为 $\{\mathbf{T}_1(\theta), \mathbf{T}_2(x), \mathbf{T}_3(y)\}$ 就是 $A_{2\times 2}$ 的基. 所以我们可以把上面的 方程 (3.1) 写成如下表达式:

$$\Lambda = \mathbf{T}_1(\theta) \cdot \mathbf{T}_1(\phi) \cdot \mathbf{T}_2(S_x) \cdot \mathbf{T}_3(S_y) \cdot \mathbf{T}_1(\omega) \cdot \mathbf{T}_2(s_x) \cdot \mathbf{T}_3(s_y). \quad (3.2)$$

根据矩阵乘法的结果, 我们可以知道上述的 m, s_x, S_x, ϕ 等参数如下:

$$s_x = \sqrt{A_{11}^2 + A_{21}^2}, \qquad \theta = \arctan\left(\frac{A_{21}}{A_{11}}\right).$$

s_y 和 m 的求解结果如下:

$$ms_y = A_{12}\cos\theta + A_{22}\sin\theta, \qquad s_y = \begin{cases} \dfrac{ms_y\cos\theta - A_{12}}{\sin\theta} & \text{如果 } \sin\theta \neq 0, \\ \dfrac{A_{22} - ms_y\sin\theta}{\cos\theta} & \text{如果 } \sin\theta = 0; \end{cases}$$

那么此时很容易知道 $m = ms_y/s_y$. 对 shear matrix 的分解结果如下:

$$S_x = \sqrt{\frac{m^2}{4} + 1} - \frac{m}{2}, \qquad S_y = \sqrt{\frac{m^2}{4} + 1} + \frac{m}{2},$$

$$\phi = -\frac{\pi}{4} - \frac{1}{2}\arctan(\frac{m}{2}), \qquad \omega = \frac{\pi}{4} - \frac{1}{2}\arctan(\frac{m}{2}).$$

最后我们只需要从右到左将这一系列的变换应用到 〈*box*〉 上即可. 从上面也可以看出, 命令 \ztool_affine_transformation:Nnnnn 仅依赖于 LaTeX3 中的 \coffin_scale:Nnn 和 \coffin_rotate:Nn 两个函数. 命令 \ztool_affine_-transformation:Nnnnn 实现过程中相关的参考链接如下:

- https://math.stackexchange.com/a/3521141/1235323;

- https://math.stackexchange.com/a/281087/1235323.

如果原 TeX 引擎提供了 shear transformation 相关的 primitive, 那么上述对 shear matrix 的分解就是不必要的. 部分的引擎中原始提供了仿射变换矩阵这一 primitive, 比如 pdfTeX 中的 \pdfsetmatrix 命令.

下面一个示例展示了如何使用 ztool 中的几个 Box 操作命令:

```
\ExplSyntaxOn                                                           例 2
% get dim of content
\dim_new:N \l_ztool_tmp_H_dim
\dim_new:N \l_ztool_tmp_W_dim
\ztool_get_ht:Nn \l_ztool_tmp_H_dim {Hello,~world!}
\ztool_get_wd:Nn \l_ztool_tmp_W_dim {Hello,~world!}
\dim_use:N \l_ztool_tmp_H_dim \quad \dim_use:N \l_ztool_tmp_W_dim\\
% set content to dim
Hello,~world|
\ztool_set_to_ht:nn {.5cm} {Hello,~world}|
\ztool_set_to_wd:nn {40pt} {Hello,~world}\par
% scale one dimension
\ztool_scale_to_wd:nn {2em}{AA}\par
\ztool_scale_to_wd:nn {2em}{AAAAAA}\par

\ztool_scale_to_ht:nn {2.5em}{\fbox{\vbox{\hbox{A}}}}\quad
\ztool_scale_to_ht:nn {2.5em}{\fbox{\vbox{\hbox{A}\hbox{A}\hbox{A}\hbox{A}}}}
\ExplSyntaxOff
```
- -

7.54619pt   58.58835pt

Hello, world|Hello, world|Hello, world

AA

AAAAAA

A

4 zdraw

这部分主要包含一些图像绘制命令, 这系列的命令并不依赖于 tikz 宏包, 它们的主要依赖项如下:

- LaTeX 2$_\varepsilon$ 内置 picture 环境;

- pict2e : LaTeX 2$_\varepsilon$ 内置 picture 环境的增强版, 提供了更好的绘图功能;

- bxeepic: 可以用于提供 dash line 支持, 目前还未引入该宏包.

---

zpic

New: 2025-05-13

`\begin{zpic}[⟨key-value⟩] ⟨draw commands⟩ \end{zpic}`

此环境基于 LaTeX 2$_\varepsilon$ 内置 picture 环境定义,

---

ztool/draw/picture/**unit**	unit = ⟨长度⟩ ....初始值: `1cm`
ztool/draw/picture/**width**	width = ⟨浮点数⟩ ....初始值: `0`
ztool/draw/picture/**height**	height = ⟨浮点数⟩ ....初始值: `0`
ztool/draw/picture/**xoffset**	xoffset = ⟨浮点数⟩ ....初始值: `0`
ztool/draw/picture/**yoffset**	yoffset = ⟨浮点数⟩ ....初始值: `0`
ztool/draw/picture/**opacity-color**	opacity-color = ⟨颜色⟩ ....初始值: `white`

上述的 ⟨*opacity-color*⟩ 选项用于设置当前 zpic 环境中的 "透明" 色彩, 也就是和当前文档默认背景色相同的色彩; 所以可能会出现 ⟨*opacity-color*⟩ 覆盖到其它 object 上的情况.

\put

New: 2025-05-13

`\put (⟨x, y⟩) {⟨content⟩}`

此命令即为 LaTeX 2$_\varepsilon$ 内置 picture 环境中的 \put 命令. **注意**: 此命令需要在 picture 或 zpic 环境中使用.

\zline

New: 2025-05-13

`\zline [⟨key-value⟩](⟨coor-1⟩)(⟨coor-2⟩)`

此命令用于绘制一条从 ⟨*coor-1*⟩ 到 ⟨*coor-2*⟩ 的线段, ⟨*key-value*⟩ 用于设置线条的属性, 可用选项请参见后续的 ⟨*parent=ztool/draw/picture/line*⟩.

---

ztool/../line/**draw**	draw = ⟨颜色⟩ ....初始值: `black`
ztool/../line/**width**	width = ⟨长度⟩ ....初始值: `.4pt`
ztool/../line/**dash**	dash = ⟨*true*|***false***⟩ ....初始值: `false`

上述 ⟨*width*⟩ 用于设置线条的宽度, ⟨*draw*⟩ 用于设置线条的颜色, ⟨*dash*⟩ 用于设置线条是否为虚线. **注意**: 目前 ⟨*dash*⟩ 选项还未适配, 处于不可用的状态.

\zvector

New: 2025-05-13

`\zvector [⟨key-value⟩](⟨coor-1⟩)(⟨coor-2⟩)`

此命令用于绘制向量, 该向量的起点为 ⟨*coor-1*⟩, 终点为 ⟨*coor-2*⟩; ⟨*key-value*⟩ 用于设置该向量的外观属性, 其继承自 ⟨*parent=ztool/draw/picture/line*⟩, 其余的可用选项请参见后续 ⟨*parent=ztool/draw/picture/line/vector*⟩.

ztool/../vector/>   > = ⟨**latex**|*pst*⟩...初始值: latex

此选项用于控制箭头的样式, 默认为 LATEX 样式, 即 \ltxarrows; ⟨*pst*⟩, 即 PsTricks, 对应于 \pstarrows 命令.

\zdraw   \zdraw [⟨*key-value*⟩](⟨*coor-1*⟩)...(⟨*coor-n*⟩);

New: 2025-05-13   此命令将绘制一条从点 ⟨*coor-1*⟩ 到点 ⟨*coor-n*⟩ 的折线段, ⟨*key-value*⟩ 继承自 ⟨*parent=ztool/draw/picture/line*⟩, 可以用于设置线条的属性, 额外可用的选项 请参见后续的 ⟨*parent=ztool/draw/picture/zdraw*⟩.

   注意: 此命令末尾的 ";" 是不能省略的, 否则会报错.

ztool/../zdraw/vector   vector = ⟨**false**|*true*⟩...初始值: false
ztool/../zdraw/cycle   cycle = ⟨**false**|*true*⟩...初始值: false
ztool/../zdraw/fill   fill   = ⟨**false**|*true*| 颜色⟩.......................................初始值: false
ztool/../zdraw/shift   shift = {⟨浮点数，浮点数⟩}...初始值: {0, 0}

当 ⟨*fill*⟩ 设置为 true 时, ⟨*cycle*⟩ 会自动设置为 true; ⟨*vector*⟩ 用于设置是否将 每一个子线段替换为向量. ⟨*shift*⟩ 分别表示 x 和 y 方向的偏移量. **注意**: ⟨*shift*⟩ 选项中的 {} 不能省略.

\zarc   \zarc[⟨*key-value*⟩](⟨浮点数，浮点数⟩)

New: 2025-05-13   此命令用于绘制一个圆弧, (⟨浮点数，浮点数⟩) 为其圆心, 默认绘制 $\frac{1}{4}$ 圆弧; ⟨*key-value*⟩ 继承自 ⟨*parent=ztool/draw/picture/line*⟩, 可以用于设置线条的 属性, 额外可用的选项请参见后续的 ⟨*parent=ztool/draw/picture/zarc*⟩.

ztool/../zarc/radius   radius = ⟨浮点数⟩..初始值: .5
ztool/../zarc/start   start = ⟨浮点数⟩..初始值: 0
ztool/../zarc/end   end   = ⟨浮点数⟩..初始值: 90
ztool/../zarc/fill   fill   = ⟨**false**|*true*| 颜色⟩.......................................初始值: false

⟨*start*⟩ 按照逆时针旋转到角度 ⟨*end*⟩ 结束; ⟨*radius*⟩ 为圆弧的半径; ⟨*fill*⟩ 用于 设置圆弧的填充颜色.

\zcircle   \zcircle[⟨*key-value*⟩](⟨浮点数，浮点数⟩)

New: 2025-05-13   此命令基于上述的 \zarc 命令, 默认情况下将以 (⟨浮点数，浮点数⟩) 为圆心绘制 一个完整的圆; ⟨*key-value*⟩ 和上述的 \zrac 命令中的 ⟨*key-value*⟩ 选项相同,

\zrectangle   \zrectangle[⟨*key-value*⟩](⟨*coor-1*⟩)(⟨*coor-2*⟩)

New: 2025-05-13   此命令用于绘制矩形, (⟨*coor-1*⟩) 和 (⟨*coor-2*⟩) 为矩形对角线的两个端点坐标; ⟨*key-value*⟩ 继承自 ⟨*parent=ztool/draw/picture/line*⟩, 其余的 ⟨*key-value*⟩ 请 参见后续 ⟨*parent=ztool/draw/picture/zrectangle*⟩.

ztool/../zrectangle/arc   arc = ⟨浮点数⟩...初始值: 0
ztool/../zrectangle/fill fill = ⟨**false**|*true*| 颜色⟩.......................................初始值: false

⟨*fill*⟩ 用于设置矩形的填充颜色, ⟨*arc*⟩ 用于设置矩形圆角对应的半径.

下面给出一些绘图示例, 方便读者理解上述绘图命令的基本使用方法:

```
\mbox{}\vskip2em                                                    例 3
\begin{zpic}[unit=2em]
  \zdraw[fill, cycle] (0, 0)(1, 0)(1, 1)(0, 1);
  \zdraw[cycle, shift={2, 0}] (0, 0)(1, 0)(1, 1)(0, 1);
  \zdraw[fill, shift={4, 0}] (0, 0)(1, 0)(1, 1)(0, 1);
  \zdraw[draw=red, width=1pt, shift={6, 0}] (0, 0)(1, 0)(1, 1)(0, 1);
  \zdraw[vector, shift={8, 0}] (0, 0)(1, 0)(1, 1)(0, 1);
  \zdraw[vector, cycle, shift={10, 0}] (0, 0)(1, 0)(1, 1)(0, 1);
  \zdraw[vector, fill, shift={12, 0}] (0, 0)(1, 0)(1, 1)(0, 1);
  \zdraw[vector, cycle, fill, shift={14, 0}] (0, 0)(1, 0)(1, 1)(0, 1);
\end{zpic}

\vskip2cm
\begin{zpic}[unit=2cm, xoffset=2]
  % 1. rectangle
  \zrectangle[arc=.1, fill=gray!20](0, 0)(2, 1)
  \zrectangle[draw=green, width=1pt](.5, .25)(1.5, .75)
  % 2. line / vecter
  \zline[width=3pt, draw=red](0, .5)(2, .5)
  \zvector[>=pst](0, 0)(1, 1)
  \zvector[draw=blue, width=2pt](1, 1)(2, 0)
  % 3. arc / circle
  \zarc[draw=blue, end=45](0, 0) % fill=<empty>
  \zarc[draw=blue, width=2pt, end=15, fill=, draw=red](0, 0)
  \zcircle[radius=.25, fill, draw=purple](1, .5)
  \zcircle[radius=.25, fill=orange, draw=none](1.5, 1)
  \zcircle[radius=.25, fill=red, draw=](2, .5)
\end{zpic}
```
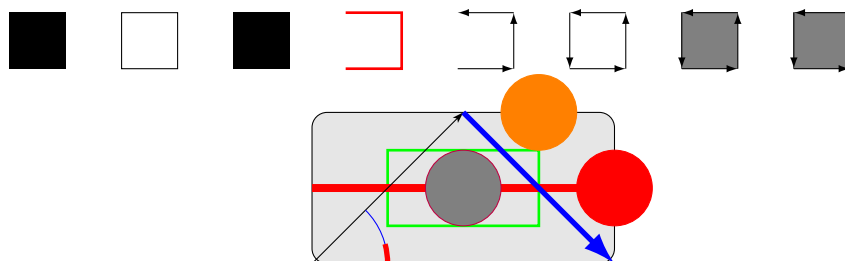
5 TODO

ztool 在将来也许会有改动, 这里列出部分将来可能会完善的功能 (□ – 未完成; ☑ – 已完成; ☒ – 不考虑该功能):

□ 重新实现 xsimverb 宏包中的 \xsim_file_write_start:nn 和 \xsim_file_-
write_stop: 命令, 使其和 ztool 宏包适配.

```
1   \NeedsTeXFormat{LaTeX2e}
2   \RequirePackage{expl3}
3   \ProvidesExplPackage{ztool}{2025/05/15}{1.0.1}{A~pre-release~tool~package~for~LaTeX}
4
5
6   %%%%%    l3keys intial patch begin    %%%%%
7   % 1. https://github.com/latex3/latex3/issues/1738
8   % 2. https://tex.stackexchange.com/q/742604/294585
9   \cs_set_protected:Npn \__keys_initialise:n #1
10    {
11      \exp_after:wN \__keys_find_key_module:wNN
12        \l_keys_path_str \s__keys_stop
13        \l_keys_key_tl \l_keys_key_str
14      \tl_set_eq:NN \l_keys_key_tl \l_keys_key_str
15      \tl_set:Nn \l_keys_value_tl {#1}
16      \cs_if_exist:cTF { \c__keys_code_root_str \l_keys_path_str }
17        {
18          \str_clear:N \l__keys_inherit_str
19          \__keys_execute:nn \l_keys_path_str {#1}
20        }
21        {
22          \cs_if_exist:cT
23            { \c__keys_inherit_root_str \__keys_parent:o \l_keys_path_str }
24            { \__keys_execute_inherit: }
25        }
26    }
27  %%%%%    l3keys intial patch end    %%%%%
28
29
30
31  % ==> l3sys-shell tool
32  % NOTE: Copy of the original 'l3sys-shell' + some modifications
33  % windows path handle
34  \cs_new:Npn \ztool_sys_path_to_win:N #1
35    {
36      \quark_if_nil:NF #1 {
37        \token_if_eq_meaning:NNTF #1 /
38          { \c_backslash_str }
39          {#1}
40        \ztool_sys_path_to_win:N
41      }
42    }
43  \cs_new:Npn \ztool_sys_path_to_win:w #1 ~ #2 \q_stop
44    {
45      \ztool_sys_path_to_win:N #1 \q_nil
46      \tl_if_empty:nF {#2}
47        {
48          \c_space_tl
49          \__sys_path_to_win:w #2 \q_stop
50        }
```

15

```
51        }
52    \cs_new:Npn \ztool_sys_path_to_win:n #1
53      {
54        \exp_after:wN \ztool_sys_path_to_win:w \tl_to_str:n {#1} ~ \q_stop
55      }
56    % respective commands
57    \cs_new_protected:Npn \ztool_shell_escape:n #1
58      {
59        \sys_if_shell_unrestricted:T
60          { \sys_shell_now:n {#1} }
61      }
62    \cs_generate_variant:Nn \ztool_shell_escape:n {e}
63    \cs_new_protected:Npe \ztool_shell_mkdir:n #1
64      {
65        \ztool_shell_escape:e {
66          \sys_if_platform_unix:T
67            {mkdir~-p~\exp_not:N \tl_to_str:n {#1}}
68          \sys_if_platform_windows:T
69            {mkdir~ \exp_not:N \ztool_sys_path_to_win:n {#1}}
70        }
71      }
72    \cs_new_protected:Npe \ztool_shell_cp:nn #1#2
73      {
74        \ztool_shell_escape:e {
75          \sys_if_platform_unix:T
76            {
77              cp~-f~ \exp_not:N \tl_to_str:n {#1} ~
78                \exp_not:N \tl_to_str:n {#2}
79            }
80          \sys_if_platform_windows:T
81            {% can NOT use wildcards in CMD
82              copy~/y~ \exp_not:N \ztool_sys_path_to_win:n {#1} ~
83                \exp_not:N \ztool_sys_path_to_win:n {#2}
84            }
85        }
86      }
87    \cs_new_protected:Npe \ztool_shell_mv:nn #1#2
88      {
89        \ztool_shell_escape:e {
90          \sys_if_platform_unix:T
91            {
92              mv~ \exp_not:N \tl_to_str:n {#1} ~
93                \exp_not:N \tl_to_str:n {#2}
94            }
95          \sys_if_platform_windows:T
96            {
97              copy~/y~ \exp_not:N \ztool_sys_path_to_win:n {#1} ~
98                \exp_not:N \ztool_sys_path_to_win:n {#2}
99                \token_to_str:N & \token_to_str:N &
100               del~/f~/q~\exp_not:N \ztool_sys_path_to_win:n {#1}
101           }
102     }
```

```
103      }
104  \cs_new_protected:Npe \ztool_shell_rm:n #1
105    {
106      \ztool_shell_escape:e {
107        \sys_if_platform_unix:T
108          { rm~-f~ \exp_not:N \tl_to_str:n {#1}  }
109        \sys_if_platform_windows:T
110          { del~/f~/q~ \exp_not:N \ztool_sys_path_to_win:n {#1} }
111      }
112    }
113  \cs_new_protected:Npe \ztool_shell_rmdir:n #1
114    {
115      \ztool_shell_mkdir:n {#1}
116      \ztool_shell_escape:e {
117        \sys_if_platform_unix:T
118          { rm~-rf~ \exp_not:N \tl_to_str:n {#1} }
119        \sys_if_platform_windows:T
120          { rmdir~/s~/q~ \exp_not:N \ztool_sys_path_to_win:n {#1} }
121      }
122    }
123  \tl_new:N \l__ztool_shell_tmp_tl
124  \cs_new_protected:Npe \ztool_get_shell_pwd:N #1
125    {
126      \exp_not:N \sys_get_shell:nnN
127        {
128          \sys_if_platform_unix:T { pwd }
129          \sys_if_platform_windows:T { cd }
130        }{
131          \char_set_catcode_other:N \exp_not:N \\
132          \char_set_catcode_other:N \exp_not:N \#
133          \char_set_catcode_other:N \exp_not:N \~
134          \char_set_catcode_other:N \exp_not:N \%
135          \char_set_catcode_space:N \exp_not:N \ %
136          \tex_endlinechar:D -1 \scan_stop:
137        }
138      \exp_not:N \l__ztool_shell_tmp_tl
139      \str_set:NV #1 \exp_not:N \l__ztool_shell_tmp_tl
140    }
141  \cs_new_protected:Npe \ztool_shell_split_ls:nN #1#2
142    {
143      \exp_not:N \sys_get_shell:nnN
144        {
145          \sys_if_platform_unix:T { ls~-1~ #1 }
146          \sys_if_platform_windows:T { dir~/b~ #1 }
147        }{
148          \ExplSyntaxOff
149          \char_set_catcode_other:N \exp_not:N \\
150          \char_set_catcode_other:N \exp_not:N \#
151          \char_set_catcode_other:N \exp_not:N \~
152          \char_set_catcode_other:N \exp_not:N \%
153          \char_set_catcode_other:n { 13 }
154        }
```

```
155        \exp_not:N \l__ztool_shell_tmp_tl                                    155
156      \str_set:NV \exp_not:N \l__sys_tmp_tl \exp_not:N \l__sys_tmp_tl         156
157      \seq_set_split:NnV #2                                                   157
158        { \char_generate:nn { `\^^M } { 12 } }                               158
159        \exp_not:N \l__ztool_shell_tmp_tl                                    159
160      \seq_pop_right:NN #2 \exp_not:N \l__sys_tmp_tl                          160
161    }                                                                        161
162  \cs_generate_variant:Nn \ztool_shell_mkdir:n {e}                           162
163  \cs_generate_variant:Nn \ztool_shell_cp:nn { ee, ne, en }                  163
164  \cs_generate_variant:Nn \ztool_shell_mv:nn { ee, ne, en }                  164
165  \cs_generate_variant:Nn \ztool_shell_rm:nn { e, f, o }                     165
166  \cs_generate_variant:Nn \ztool_shell_rmdir:nn { e, f, o }                  166
167  \cs_generate_variant:Nn \ztool_get_shell_pwd:N {c}                         167
168  \cs_generate_variant:Nn \ztool_shell_split_ls:nN {nc}                      168
169                                                                             169
170                                                                             170
171                                                                             171
172  % ==> file IO operations                                                   172
173  \ior_new:N \g_ztool_file_read_ior                                          173
174  \ior_new:N \g_ztool_file_append_ior                                        174
175  \iow_new:N \g_ztool_file_append_iow                                        175
176  \tl_new:N \l_ztool_current_line                                            176
177  \str_clear:N \l_ztikz_file_ori_content_str                                 177
178  \seq_new:N \l_ztool_file_seq                                               178
179  \cs_new_protected:Npn \ztool_read_file_as_seq:nnN #1#2#3                   179
180    {% #1: bool(True to keep spaces, False to trim); #2: file name; #3: seq  180
181      \seq_clear:N #3                                                        181
182      \file_if_exist:nT {#2}                                                 182
183        {                                                                    183
184          \ior_open:Nn \g_ztool_file_read_ior {#2}                           184
185          \ior_map_inline:Nn \g_ztool_file_read_ior                          185
186            {                                                                186
187              \bool_if:nTF {#1}                                              187
188                { \seq_put_right:Nn #3 {##1} }                               188
189                { \seq_put_right:Nn #3 {\tl_trim_spaces:n {##1}} }           189
190            }                                                                190
191          \ior_close:N \g_ztool_file_read_ior                                191
192        }                                                                    192
193    }                                                                        193
194  \cs_new_protected:Npn \ztool_gread_file_as_seq:nnN #1#2#3                  194
195    {% #1: bool(True to keep spaces, False to trim); #2: file name; #3: seq  195
196      \seq_gclear:N #3                                                       196
197      \file_if_exist:nT {#2}                                                 197
198        {                                                                    198
199          \ior_open:Nn \g_ztool_file_read_ior {#2}                           199
200          \ior_map_inline:Nn \g_ztool_file_read_ior                          200
201            {                                                                201
202              \bool_if:nTF {#1}                                              202
203                { \seq_gput_right:Nn #3 {##1} }                              203
204                { \seq_gput_right:Ne #3 {\tl_trim_spaces:n {##1}} }          204
205            }                                                                205
206          \ior_close:N \g_ztool_file_read_ior                                206
```

18

```
207          }
208       }
209   \cs_generate_variant:Nn \ztool_read_file_as_seq:nnN { ne, nnc, nec }
210   \cs_generate_variant:Nn \ztool_gread_file_as_seq:nnN { ne, nnc, nec }
211
212   \cs_new_protected:Npn \ztool_file_new:nn #1#2
213     {% #1: \c_true_bool to allow overwrite; #2: file name
214       \bool_if:nT {#1}
215         {
216           \iow_open:Nn \g_ztool_file_append_iow {#2}
217           \iow_close:N \g_ztool_file_append_iow
218         }
219     }
220   \cs_new_protected:Npn \ztool_append_to_file:nn #1#2
221     {% #1: file name; #2: content
222       \seq_clear:N \l_ztool_file_seq
223       \file_if_exist:nF {#1}{ \ztool_file_new:nn {\c_false_bool}{#1} }
224       \ior_open:Nn \g_ztool_file_append_ior {#1}
225       \ior_str_map_inline:Nn \g_ztool_file_append_ior
226         {
227           \seq_put_right:Nn \l_ztool_file_seq
228             { ##1 }
229         }
230       \iow_open:Nn \g_ztool_file_append_iow {#1}
231       \iow_now:Ne \g_ztool_file_append_iow
232         { \seq_use:Nn \l_ztool_file_seq {^^J} }
233       \iow_now:Ne \g_ztool_file_append_iow {#2}
234       \iow_close:N \g_ztool_file_append_iow
235     }
236   \cs_generate_variant:Nn \ztool_append_to_file:nn { no, nf, ne, ee }
237   \cs_new_protected:Npn \ztool_replace_file_line:nnn #1#2#3
238     {% #1:file name; #2:line index; #3:replacement
239       \seq_clear:N \l_ztool_file_seq
240       \file_if_exist:nT {#1}{
241         \ior_open:Nn \g_ztool_file_read_ior {#1}
242         \ior_str_map_inline:Nn \g_ztool_file_read_ior
243           {
244             \seq_put_right:Nn \l_ztool_file_seq {##1}
245           }
246         \ior_close:N \g_ztool_file_read_ior
247         \seq_set_item:Nnn \l_ztool_file_seq {#2}
248           { #3 }
249         \iow_open:Nn \g_ztool_file_append_iow {#1}
250         \iow_now:Ne \g_ztool_file_append_iow
251           { \seq_use:Nn \l_ztool_file_seq {^^J} }
252         \iow_close:N \g_ztool_file_append_iow
253       }
254     }
255   \cs_generate_variant:Nn \seq_set_item:Nnn { Nne }
256   \cs_generate_variant:Nn \ztool_replace_file_line:nnn { e, ene, eee }
257   \cs_new_protected:Npn \ztool_insert_to_file:nnn #1#2#3
258     {% #1:file name; #2:line index; #3:content
```

```
259    \seq_clear:N \l_ztool_file_seq
260    \file_if_exist:nT {#1}{
261      \ior_open:Nn \g_ztool_file_read_ior {#1}
262      \ior_str_map_inline:Nn \g_ztool_file_read_ior
263        {
264          \seq_put_right:Nn \l_ztool_file_seq {##1}
265        }
266      \ior_close:N \g_ztool_file_read_ior
267      \tl_set:No \l_ztool_current_line
268        { \seq_item:Nn \l_ztool_file_seq {#2} }
269      \seq_set_item:Nne \l_ztool_file_seq {#2}
270        { #3^^J\l_ztool_current_line }
271      \iow_open:Nn \g_ztool_file_append_iow {#1}
272      \iow_now:Ne \g_ztool_file_append_iow
273        { \seq_use:Nn \l_ztool_file_seq {^^J} }
274      \iow_close:N \g_ztool_file_append_iow
275    }
276  }
277  \cs_generate_variant:Nn \ztool_insert_to_file:nn { ne, nf, ee }



281  % ==> box manipulation tool
282  \cs_set:Nn \__ztool_leave_vmode:
283    { \ifvmode \leavevmode \fi }
284  % catch box dimension
285  \box_new:N \l_ztool_measure_box
286  \cs_new:Npn \ztool_box_set_to:NNn #1#2#3 {
287    \hbox_set:Nn \l_ztool_measure_box {#3}
288    \dim_set:Nn #2 {#1 \l_ztool_measure_box}
289    \box_set_eq:NN \l_ztool_measure_box \c_empty_box
290  }
291  \cs_new:Npn \ztool_box_gset_to:NNn #1#2#3 {
292    \hbox_set:Nn \l_ztool_measure_box {#3}
293    \dim_gset:Nn #2 {#1 \l_ztool_measure_box}
294    \box_set_eq:NN \l_ztool_measure_box \c_empty_box
295  }
296  \cs_new:Npn \ztool_get_ht:Nn
297    { \ztool_box_set_to:NNn \box_ht:N }
298  \cs_new:Npn \ztool_get_ht_plus_dp:Nn
299    { \ztool_box_set_to:NNn \box_ht_plus_dp:N }
300  \cs_new:Npn \ztool_get_wd:Nn
301    { \ztool_box_set_to:NNn \box_wd:N }
302  \cs_new:Npn \ztool_get_dp:Nn
303    { \ztool_box_set_to:NNn \box_dp:N }
304  \cs_new:Npn \ztool_gget_ht:Nn
305    { \ztool_box_gset_to:NNn \box_ht:N }
306  \cs_new:Npn \ztool_gget_wd:Nn
307    { \ztool_box_gset_to:NNn \box_wd:N }
308  \cs_new:Npn \ztool_gget_dp:Nn
309    { \ztool_box_gset_to:NNn \box_dp:N }
310  \cs_generate_variant:Nn \ztool_get_ht:Nn  { Ne, ce }
```

311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362

```
\cs_generate_variant:Nn \ztool_get_ht_plus_dp:Nn  { Ne, ce }
\cs_generate_variant:Nn \ztool_get_wd:Nn  { Ne, ce }
\cs_generate_variant:Nn \ztool_gget_ht:Nn { Ne, ce }
\cs_generate_variant:Nn \ztool_gget_wd:Nn { Ne, ce }


%% modify box content
% 1. auto scale and rotate (smaller of two)
\cs_new_protected:Npn \ztool_autoset_to_wd_and_ht:nnn #1#2#3
  {% #1:width; #2:height; #3:object
    \hbox_set:Nn \l_tmpa_box {#3}
    \box_autosize_to_wd_and_ht:Nnn \l_tmpa_box {#1}{#2}
    \__ztool_leave_vmode:
    \box_use:N \l_tmpa_box
  }
\cs_new_protected:Npn \ztool_rotate:nn #1#2
  {% #1:angle; #2:object
    \hbox_set:Nn \l_tmpa_box {#2}
    \box_rotate:Nn \l_tmpa_box {#1}
    \__ztool_leave_vmode:
    \box_use:N \l_tmpa_box
  }
\cs_generate_variant:Nn \ztool_rotate:nn { e, ne, ee }
\cs_generate_variant:Nn \ztool_autoset_to_wd_and_ht:nnn { nne, een, eee }

% 2. width/height scale to same time
\cs_new_protected:Npn \ztool_set_to_wd:nn #1#2
  {% #1:width; #2:object
    \hbox_set:Nn \l_tmpa_box {#2}
    \box_resize_to_wd:Nn \l_tmpa_box {#1}
    \__ztool_leave_vmode:
    \box_use:N \l_tmpa_box
  }
\cs_new_protected:Npn \ztool_set_to_ht:nn #1#2
  {% #1:height; #2:object
    \hbox_set:Nn \l_tmpa_box {#2}
    \box_resize_to_ht:Nn \l_tmpa_box {#1}
    \__ztool_leave_vmode:
    \box_use:N \l_tmpa_box
  }
\cs_generate_variant:Nn \ztool_set_to_wd:nn { e, ne, ee }
\cs_generate_variant:Nn \ztool_set_to_ht:nn { e, ne, ee }

% 3. only scale one dimension
% NOTE: if boxwdcontent ≤ given dim, no manipulation
\cs_new_protected:Npn \ztool_scale_to_wd:nn #1#2
  {
    \hbox_set:Nn \l_tmpa_box {#2}
    \dim_set:Nn \l_tmpa_dim { \box_wd:N \l_tmpa_box }
    \fp_set:Nn \l_tmpa_fp
      {
        \fp_eval:n { min(1, \dim_ratio:nn {#1}{\l_tmpa_dim}) }
```

```
363             }
364             \box_scale:Nnn \l_tmpa_box {\l_tmpa_fp}{1}
365             \__ztool_leave_vmode:
366             \box_use:N \l_tmpa_box
367         }
368     \cs_new_protected:Npn \ztool_scale_to_ht:nn #1#2
369         {% take depth into consideration
370             \hbox_set:Nn \l_tmpa_box {#2}
371             \dim_set:Nn \l_tmpa_dim { \box_ht_plus_dp:N \l_tmpa_box }
372             \fp_set:Nn \l_tmpa_fp
373                 {
374                     \fp_eval:n { min(1, \dim_ratio:nn {#1}{\l_tmpa_dim}) }
375                 }
376             \box_scale:Nnn \l_tmpa_box {1}{\l_tmpa_fp}
377             \__ztool_leave_vmode:
378             \box_use:N \l_tmpa_box
379         }
380     \cs_new_protected:Npn \ztool_scale_to_wd_and_ht:nnn #1#2#3
381         {% take depth into consideration
382             \hbox_set:Nn \l_tmpa_box {#3}
383             \dim_set:Nn \l_tmpa_dim { \box_wd:N \l_tmpa_box }
384             \dim_set:Nn \l_tmpb_dim { \box_ht_plus_dp:N \l_tmpa_box }
385             \fp_set:Nn \l_tmpa_fp
386                 {
387                     \fp_eval:n { min(1, \dim_ratio:nn {#1}{\l_tmpa_dim}) }
388                 }
389             \fp_set:Nn \l_tmpb_fp
390                 {
391                     \fp_eval:n { min(1, \dim_ratio:nn {#2}{\l_tmpb_dim}) }
392                 }
393             \box_scale:Nnn \l_tmpa_box {\l_tmpa_fp}{\l_tmpb_fp}
394             \__ztool_leave_vmode:
395             \box_use:N \l_tmpa_box
396         }
397     \cs_generate_variant:Nn \ztool_scale_to_wd:nn { e, ne, ee }
398     \cs_generate_variant:Nn \ztool_scale_to_ht:nn { e, ne, ee }
399     \cs_generate_variant:Nn \ztool_scale_to_wd_and_ht:nnn { nne, nno, eee }
400
401
402     %% box content align
403     \seq_new:N \l__ztool_boxitem_seq
404     \cs_set_protected:Npn \ztool_box_item_align:Nnnn #1#2#3#4
405         {% #1:cmd, #2:width, #3:object, #4:align format(left, right, scatter, center)
406             \hb@xt@#2{
407                 \tl_map_inline:nn {#3}
408                     {
409                         \seq_put_right:No \l__ztool_boxitem_seq {\exp_not:N #1{##1}}
410                     }
411                 \str_case:nnF { #4 }
412                     {
413                         { left }{ \seq_use:Nn \l__ztool_boxitem_seq {}\hfill }
414                         { right }{ \hfill\seq_use:Nn \l__ztool_boxitem_seq {} }
```

```
415              { scatter}{ \seq_use:Nn \l__ztool_boxitem_seq {\hfill} }
416              { center }{ \hfill\seq_use:Nn \l__ztool_boxitem_seq {}\hfill }
417              { tower  }
418                {
419                  \edef\seq@count{\seq_count:N \l__ztool_boxitem_seq}
420                  \seq_map_indexed_inline:Nn \l__ztool_boxitem_seq
421                    {% ##1: index, ##2: content
422                      %% Method II: plain
423                      \edef\item@width{\dim_eval:n {#2/(\seq@count+1)}}
424                      \hskip\item@width\clap{##2}
425                    }\hskip\item@width\hss
426                }
427              { custom }
428                {
429                  \def\total@width{#2}
430                  \def\align@cmd{#1}
431                  \def\align@object{#3}
432                  \def\align@format{#4}
433                  \tl_use:N \l__ztex_boxitem_align_custom_tl
434                }
435            }{\relax}
436        }
437      \seq_clear:N \l__ztool_boxitem_seq
438    }
439  \cs_generate_variant:Nn \ztool_box_item_align:Nnnn { c, Nnno, cnno, Nne, Nnee }
440
441
442  %% affine transformation
443  % REF:
444  % 1. https://math.stackexchange.com/a/3521141/1235323
445  % 2. https://math.stackexchange.com/a/281087/1235323
446  \cs_new:Npn \ztool_fp_to_rad:n #1
447    { \fp_eval:n {#1/pi*180} }
448  \cs_new:Npn \ztool_matrix_det:nnnn #1#2#3#4
449    {
450      \fp_eval:n { #1*#4 - #2*#3 }
451    }
452  % (translation) + x-scale + y-scale + rotate
453  \fp_new:N \g_affine_precision_fp
454  \fp_set:Nn \g_affine_precision_fp {0.0001}
455  \fp_new:N \l__affine_@@_a_fp
456  \fp_new:N \l__affine_@@_b_fp
457  \fp_new:N \l__affine_@@_c_fp
458  \fp_new:N \l__affine_@@_d_fp
459  \msg_set:nnn { ztool }{affine-det-zero}
460    {
461      current~determination~of~the~affine~transformation~
462      matrix~equals~to~zero,~give~up~this~transformation
463    }
464
465  \coffin_new:N \l__affine_trans_coffin
466  \cs_generate_variant:Nn \coffin_typeset:Nnnnn { Nxxxx }
```

```
467  \cs_new:Npn \ztool_affine_transformation:Nnnnn #1#2#3#4#5
468    {% #1:box; #2:a_{11}; #3:a_{21}; #4:a_{12}; #5:a_{22}.
469      \fp_compare:nNnT
470        { abs(\ztool_matrix_det:nnnn {#2}{#3}{#4}{#5}) }
471          < { \g_affine_precision_fp }
472        { \prg_map_break:Nn \l__affine_matrix_det_zero
473          { \msg_warning:nn { ztool }{affine-det-zero} }}
474      \fp_set:Nn \l__affine_@@_a_fp {#2}
475      \fp_set:Nn \l__affine_@@_b_fp {#3}
476      \fp_set:Nn \l__affine_@@_c_fp {#4}
477      \fp_set:Nn \l__affine_@@_d_fp {#5}
478      \__box_affine_transform:N #1
479      \prg_break_point:Nn \l__affine_matrix_det_zero { }
480      \coffin_typeset:Nxxxx \l__affine_trans_coffin
481        { \l__ztool_affine_pole_a_tl }{ \l__ztool_affine_pole_b_tl }
482        { \l__ztool_affine_xoffset_dim }{ \l__ztool_affine_yoffset_dim }
483    }
484  \cs_generate_variant:Nn \ztool_affine_transformation:Nnnnn { Neeee, cnnnn, ceeee }
485  \cs_new:Npn \__box_affine_transform:N #1
486    {
487      % transform debug
488      \bool_if:NT \g_ztool_affine_debug_bool
489        {
490          \noindent\dotfill\[\begin{bmatrix}
491            \fp_use:N \l__affine_@@_a_fp & \fp_use:N \l__affine_@@_c_fp\\
492            \fp_use:N \l__affine_@@_b_fp & \fp_use:N \l__affine_@@_d_fp
493          \end{bmatrix}\]
494        }
495      % get affine parameters
496      \__affine_trans_get_sx:
497      \__affine_trans_get_theta:
498      \__affine_trans_get_sy:
499      \__affine_trans_get_Sx:
500      \__affine_trans_get_Sy:
501      \__affine_trans_get_phi:
502      \__affine_trans_get_omega:
503      % start transform box/coffin
504      \coffin_scale:Nnn #1
505        { \l__box_affine_sx_fp }
506        { \l__box_affine_sy_fp }
507      \coffin_rotate:Nn #1
508        { \ztool_fp_to_rad:n {\l__box_affine_omega_fp} }
509      \coffin_scale:Nnn #1
510        { \l__box_affine_Sx_fp }
511        { \l__box_affine_Sy_fp }
512      \coffin_rotate:Nn #1
513        { \ztool_fp_to_rad:n {\l__box_affine_phi_fp} }
514      \coffin_rotate:Nn #1
515        { \ztool_fp_to_rad:n {\l__box_affine_theta_fp} }
516    }
517  \keys_define:nn { ztool / affine }
518    {
```

```
519      debug   .bool_gset:N = \g_ztool_affine_debug_bool,
520      debug   .initial:n   = false,
521      debug   .default:n   = true,
522      pole-1  .tl_set:N    = \l__ztool_affine_pole_a_tl,
523      pole-2  .tl_set:N    = \l__ztool_affine_pole_b_tl,
524      pole-1  .initial:n   = { l },
525      pole-2  .initial:n   = { b },
526      xoffset .dim_set:N   = \l__ztool_affine_xoffset_dim,
527      yoffset .dim_set:N   = \l__ztool_affine_yoffset_dim,
528      xoffset .initial:n   = { 0pt },
529      yoffset .initial:n   = { 0pt },
530    }
531  \NewDocumentCommand{\ztoolboxaffine}{O{}m>{\SplitList{,}}m}
532    {% #1:key-value; #2:content; #3:matrix.
533      \group_begin:
534        \keys_set:nn { ztool / affine } {#1}
535        \hcoffin_set:Nn \l__affine_trans_coffin {#2}
536        \ztool_affine_transformation:Nnnnn \l__affine_trans_coffin #3
537      \group_end:
538    }
539  % internal affine transform functions
540  \cs_new:Nn \__ztool_affine_debug_fp:N
541    {
542      \bool_if:NTF \g_ztool_affine_debug_bool
543        { \string #1 % \show #1
544          ~=~\fp_use:N #1\\
545        }{ \relax }
546    }
547  \fp_new:N \l__box_affine_sx_fp
548  \cs_new:Nn \__affine_trans_get_sx:
549    {
550      \fp_set:Nn \l__box_affine_sx_fp
551        { \fp_eval:n {sqrt(\l__affine_@@_a_fp^2 + \l__affine_@@_b_fp^2)} }
552      \__ztool_affine_debug_fp:N \l__box_affine_sx_fp
553    }
554  \fp_new:N \l__box_affine_theta_fp
555  \cs_new:Nn \__affine_trans_get_theta:
556    {
557      \fp_set:Nn \l__box_affine_theta_fp
558        { \fp_eval:n {atan(\l__affine_@@_b_fp/\l__affine_@@_a_fp)} }
559      \__ztool_affine_debug_fp:N \l__box_affine_theta_fp
560    }
561  \fp_new:N \l__box_affine_msy_fp
562  \cs_new:Nn \__affine_trans_get_msy:
563    {
564      \fp_set:Nn \l__box_affine_msy_fp
565        { \fp_eval:n {
566          \l__affine_@@_c_fp*cos(\l__box_affine_theta_fp)
567          +
568          \l__affine_@@_d_fp*sin(\l__box_affine_theta_fp)
569        } }
570      \__ztool_affine_debug_fp:N \l__box_affine_msy_fp
```

```
571     }                                                                            571
572   \fp_new:N \l__box_affine_sy_fp                                                572
573   \cs_new:Nn \__affine_trans_get_sy:                                            573
574     {                                                                            574
575       \__affine_trans_get_msy:                                                   575
576       \bool_if:nTF                                                               576
577         {                                                                        577
578           \fp_compare_p:nNn { abs(sin(\l__box_affine_theta_fp)) }                578
579             < {\c_zero_fp + \g_affine_precision_fp}                              579
580         }{                                                                       580
581           \fp_set:Nn \l__box_affine_sy_fp                                        581
582             {                                                                    582
583               ( \l__affine_@@_d_fp -                                         ↙   583
                    \l__box_affine_msy_fp*sin(\l__box_affine_theta_fp) )
584               / cos(\l__box_affine_theta_fp)                                     584
585             }                                                                    585
586         }{                                                                       586
587           \fp_set:Nn \l__box_affine_sy_fp                                        587
588             {                                                                    588
589               ( \l__box_affine_msy_fp*cos(\l__box_affine_theta_fp) -        ↙   589
                    \l__affine_@@_c_fp )
590               / sin(\l__box_affine_theta_fp)                                     590
591             }                                                                    591
592         }                                                                        592
593       \__ztool_affine_debug_fp:N \l__box_affine_sy_fp                            593
594     }                                                                            594
595   \fp_new:N \l__box_affine_m_fp                                                  595
596   \cs_new:Nn \__affine_trans_get_m:                                             596
597     {                                                                            597
598       \fp_set:Nn \l__box_affine_m_fp                                             598
599         { \l__box_affine_msy_fp / \l__box_affine_sy_fp }                         599
600       \__ztool_affine_debug_fp:N \l__box_affine_m_fp                             600
601     }                                                                            601
602   \fp_new:N \l__box_affine_Sx_fp                                                 602
603   \fp_new:N \l__box_affine_Sy_fp                                                 603
604   \cs_new:Nn \__affine_trans_get_Sx:                                            604
605     {                                                                            605
606       \__affine_trans_get_m:                                                     606
607       \fp_set:Nn \l__box_affine_Sx_fp                                            607
608         { sqrt(\l__box_affine_m_fp^2/4 + 1) - \l__box_affine_m_fp/2 }            608
609       \__ztool_affine_debug_fp:N \l__box_affine_Sx_fp                            609
610     }                                                                            610
611   \cs_new:Nn \__affine_trans_get_Sy:                                            611
612     {                                                                            612
613       \fp_set:Nn \l__box_affine_Sy_fp                                            613
614         { sqrt(\l__box_affine_m_fp^2/4 + 1) + \l__box_affine_m_fp/2 }            614
615       \__ztool_affine_debug_fp:N \l__box_affine_Sy_fp                            615
616     }                                                                            616
617   \fp_new:N \l__box_affine_phi_fp                                                617
618   \fp_new:N \l__box_affine_omega_fp                                              618
619   \cs_new:Nn \__affine_trans_get_phi:                                           619
620     {                                                                            620
```

```
621    \fp_set:Nn \l__box_affine_phi_fp
622       { -pi/4 - 1/2*atan(\l__box_affine_m_fp/2) }
623    \__ztool_affine_debug_fp:N \l__box_affine_phi_fp
624    }
625  \cs_new:Nn \__affine_trans_get_omega:
626    {
627    \fp_set:Nn \l__box_affine_omega_fp
628       { pi/4 - 1/2*atan(\l__box_affine_m_fp/2) }
629    \__ztool_affine_debug_fp:N \l__box_affine_omega_fp
630    }



% ==> ztool draw (based on package 'pict2e' and 'picture' env)
634
635  \RequirePackage{pict2e}
636  \cs_new:Npn \_@@_begin_picture:nnnn #1#2#3#4
637    { \begin{picture}
638       (\fp_eval:n {#1}, \fp_eval:n {#2})
639       (\fp_eval:n {-#3}, \fp_eval:n {-#4}) }
640  \cs_new:Nn \_@@_end_picture:
641    { \end{picture} }
642  \cs_new:Npn \__@@_pic_put:nnn #1#2#3
643    { \put(\fp_eval:n {#1}, \fp_eval:n {#2}){ #3 } }
644  \cs_generate_variant:Nn \_@@_begin_picture:nnnn { VVVV, eeee }
645  \cs_generate_variant:Nn \__@@_pic_put:nnn { VVV, een }

% picture environment alias
648  \keys_define:nn { ztool / draw / picture }
649    {
650      unit    .dim_set:N = \l__pic_unit_dim,
651      unit    .initial:n = { 1cm },
652      width   .fp_set:N  = \l__pic_width_fp,
653      width   .initial:n = 0,
654      height  .fp_set:N  = \l__pic_height_fp,
655      height  .initial:n = 0,
656      xoffset .fp_set:N  = \l__pic_xoffset_fp,
657      xoffset .initial:n = 0,
658      yoffset .fp_set:N  = \l__pic_yoffset_fp,
659      yoffset .initial:n = 0,
660      opacity-color .tl_set:N  = \l__pic_opacity_color_tl,
661      opacity-color .initial:n = { white },
662    }
663  \NewDocumentEnvironment{zpic}{O{}}
664    {
665      \group_begin:
666      \keys_set:nn { ztool / draw / picture } {#1}
667      \setlength\unitlength{ \l__pic_unit_dim }
668      \_@@_begin_picture:VVVV
669        \l__pic_width_fp  \l__pic_height_fp
670        \l__pic_xoffset_fp\l__pic_yoffset_fp
671    }{
672      \_@@_end_picture:
```

```
673      \group_end:
674    }
675

676

677  % picture commands alias
678  \cs_new:Npn \__coor_st:n #1
679    { \clist_item:nn {#1}{1} }
680  \cs_new:Npn \__coor_nd:n #1
681    { \clist_item:nn {#1}{2} }
682  \cs_new:Npn \__coor_rd:n #1#2
683    { \clist_item:nn {#1}{3} }
684  \cs_new:Npn \__coor_st_nd:n #1
685    {
686      {\clist_item:nn {#1}{1}}
687      {\clist_item:nn {#1}{2}}
688    }
689  \cs_new:Npn \__coor_st_nd_rd:n #1
690    {
691      {\clist_item:nn {#1}{1}}
692      {\clist_item:nn {#1}{2}}
693      {\clist_item:nn {#1}{3}}
694    }
695  \cs_generate_variant:Nn \__coor_st:n { V, e }
696  \cs_generate_variant:Nn \__coor_nd:n { V, e }
697  \cs_generate_variant:Nn \__coor_rd:n { V, e }
698  \cs_generate_variant:Nn \__coor_st_nd:n { V, e }
699  \cs_generate_variant:Nn \__coor_st_nd_rd:n { V, e }
700

701  \bool_new:N \l__ztool_invalid_color_bool
702  \cs_new:Npn \__color_safe_use:n #1
703    {
704      \__color_if_valid:nT {#1}
705        { \color{#1} }
706    }
707  \prg_new_conditional:Npnn \__color_if_valid:n #1 {p, T, F, TF}
708    {
709      \def\ztool@targer@color{#1}
710      \def\ztool@color@none{none}
711      \bool_if:eTF
712        {
713          \tl_if_empty_p:e {#1} ||
714          \tl_if_eq_p:NN \ztool@color@none \ztool@targer@color
715        }{ \prg_return_false: }
716        { \prg_return_true: }
717    }
718  \prg_generate_conditional_variant:Nnn \__color_if_valid:n
719    { V, e }{ p, T, F, TF }
720  \cs_generate_variant:Nn \__color_safe_use:n { V, e }
721

722

723  % --> line/vector
724  \fp_new:N \l__draw_vector_slope_fp
```

```
725  \fp_new:N \l__draw_vector_normal_fp                                          725
726  \fp_new:N \l__draw_vector_xsep_fp                                            726
727  \cs_new:Npn \__@@_pic_line:nnn #1#2#3                                        727
728    {% #1:x; #2:y; #3:x-distance NOT the length                               728
729      \line(\fp_eval:n {#1}, \fp_eval:n {#2})                                 729
730        { \fp_eval:n {#3} }                                                   730
731    }                                                                          731
732  \cs_new:Npn \__@@_pic_vector:nnn #1#2#3                                      732
733    {% #1:x; #2:y; #3:x-distance NOT the length                               733
734      \vector(\fp_eval:n {#1}, \fp_eval:n {#2})                               734
735        { \fp_eval:n {#3} }                                                   735
736    }                                                                          736
737  \keys_define:nn { ztool / draw / picture / line }                           737
738    {                                                                          738
739      draw   .tl_set:N  = \l__pic_line_draw_color_tl,                         739
740      draw   .initial:n = { black },                                          740
741      % color  .meta:n   =  draw = #1 , % alias for 'draw'                    741
742      width  .dim_set:N = \l__pic_line_width_dim,                            742
743      width  .initial:n = { .4pt },                                           743
744      dash   .bool_set:N = \l__pic_line_dash_bool,                           744
745      dash   .initial:n = { false },                                          745
746    }                                                                          746
747  \cs_new_protected:Nn \__pic_set_line_width:                                 747
748    {                                                                          748
749      \linethickness{ \l__pic_line_width_dim }                               749
750    }                                                                          750
751  \cs_new_protected:Nn \__pic_set_line_color:                                 751
752    {                                                                          752
753      \__color_safe_use:V \l__pic_line_draw_color_tl                         753
754    }                                                                          754
755  \cs_new_protected:Nn \__pic_set_fill_color:                                 755
756    {                                                                          756
757      \__color_safe_use:V \l__pic_region_fill_color_tl                       757
758    }                                                                          758
759  \def\z@pic@vector@style{\ltxarrows}                                         759
760  \keys_define:nn { ztool / draw / picture }                                  760
761    {                                                                          761
762      vector .inherit:n = { ztool/draw/picture/line },                       762
763    }                                                                          763
764  \keys_define:nn { ztool / draw / picture / vector }                         764
765    {                                                                          765
766      >          .choice:,                                                     766
767      > / latex  .code:n = {\def\z@pic@vector@style{\ltxarrows}},             767
768      > / pst    .code:n = {\def\z@pic@vector@style{\pstarrows}},             768
769      > / unknown .code:n =                                                    769
770        {                                                                      770
771          \msg_set:nnn { ztool }{unknown-arrow-style}                         771
772            { Unknown~arrow~style,~use~'latex'~or~'pst'. }                    772
773          \msg_error:nn { ztool }{unknown-arrow-style}                        773
774        }                                                                      774
775    }                                                                          775
776  \cs_new_protected:Npn \ztool_pic_line_vector:nnnn #1#2#3#4                  776
```

```
777    {% #1:line/vector; #2:key-value; #3:start coor; #4:end coor;
778      \group_begin:
779      \keys_set:nn { ztool / draw / picture / #1 }{#2}
780      \fp_set:Nn \l__draw_vector_slope_fp
781        { (\__coor_nd:n {#4} - \__coor_nd:n {#3})
782        / (\__coor_st:n {#4} - \__coor_st:n {#3}) }
783      \fp_set:Nn \l__draw_vector_xsep_fp
784        { abs(\__coor_st:n {#4} - \__coor_st:n {#3}) }
785      \z@pic@vector@style
786      \__pic_set_line_width:
787      \exp_last_unbraced:Ne \__@@_pic_put:nnn {\__coor_st_nd:n {#3}}
788        {
789          \__pic_set_line_color:
790          \cs:w __@@_pic_#1:nnn\cs_end: {1}
791            { \l__draw_vector_slope_fp }
792            { \l__draw_vector_xsep_fp  }
793        }
794      \group_end:
795    }
796  \NewDocumentCommand{\zline}{O{}d()d()}
797    {
798      \ztool_pic_line_vector:nnnn {line}{#1}{#2}{#3}
799    }
800  \NewDocumentCommand{\zvector}{O{}d()d()}
801    {
802      \ztool_pic_line_vector:nnnn {vector}{#1}{#2}{#3}
803    }
804
805
806  % --> \zdraw -- similar to \tikz command in tikz
807  % NOTE: these line/vector commands are identical to
808  %  1. \Line (x_1,y_1)(x_2,y_2),          \Vector(x_1,y_1)(x_2,y_2)
809  %  2. \polyline (x_1,y_1)···(x_n,y_n), \polyvector(x_1,y_1)···(x_n,y_n)
810  %  3. \polygon (x_1,y_1)···(x_n,y_n), when set 'cycle',
811  %     \polygon*(x_1,y_1)···(x_n,y_n), when set 'fill' (auto cycle).
812  %  4. Trim leading space after '\polygon' or '*' to avoid error !!
813  \cs_new:Npn \__@@_pic_Line:nnnn #1#2#3#4
814    { \Line (#1, #2)(#3, #4) }
815  \cs_new:Npn \__@@_pic_Vector:nnnn #1#2#3#4
816    { \Vector (#1, #2)(#3, #4) }
817  \cs_new:Npn \__@@_pic_polyline:n #1
818    {
819      \tl_set:Ne \l_tmpa_tl {\tl_trim_spaces:e {#1}}
820      \exp_last_unbraced:NV \polyline \l_tmpa_tl
821    }
822  \cs_new:Npn \__@@_pic_polyvector:n #1
823    {
824      \tl_set:Ne \l_tmpa_tl {\tl_trim_spaces:e {#1}}
825      \exp_last_unbraced:NV \polyvector \l_tmpa_tl
826    }
827  \cs_new:Npn \__@@_pic_polygon:nn #1#2
828    {
```

```
829    \tl_set:Ne \l_tmpa_tl {\tl_trim_spaces:e {#1}}
830    \tl_set:Ne \l_tmpb_tl {\tl_trim_spaces:e {#2}}
831    \tl_set:Ne \l_tmpa_tl { \l_tmpa_tl\l_tmpb_tl }
832    \exp_last_unbraced:NV \polygon \l_tmpa_tl
833    }
834  \cs_generate_variant:Nn \__@@_pic_polygon:nn { nV, ne }
835  \tl_new:N \l__pic_region_fill_color_tl
836  \bool_new:N \l__pic_region_fill_bool
837  \keys_define:nn { ztool / draw / picture / region }
838    {
839      fill   .choices:nn  = { true, false }{
840        \use:c { bool_set_ \l_keys_choice_tl :N }
841          \l__pic_region_fill_bool
842      },
843      fill   .initial:n  = { false },
844      fill   .default:n  = { true  },
845      fill / unknown .code:n = {
846        \tl_if_empty:eF \l_keys_value_tl
847          { \bool_set_true:N \l__pic_region_fill_bool }
848        \tl_set:Ne \l__pic_region_fill_color_tl { \l_keys_value_tl }
849      },
850    }
851  \keys_define:nn { ztool / draw / picture }
852    {
853      zdraw   .inherit:n  =   {
854        ztool/draw/picture/line,
855        ztool/draw/picture/vector,
856        ztool/draw/picture/region,
857      },
858    }
859  \keys_define:nn { ztool / draw / picture / zdraw }
860    {
861      vector  .bool_set:N = \l__pic_draw_vector_bool,
862      vector  .initial:n  = { false },
863      cycle   .bool_set:N = \l__pic_draw_cycle_bool,
864      cycle   .initial:n  = { false },
865      shift   .tl_set:N   = \l__pic_draw_shift_tl,
866      shift   .initial:n  = { 0, 0 },
867    }
868  \cs_new:Npn \__region_fill_color_miss:n #1
869    {
870      \bool_if:eT {
871        \l__pic_region_fill_bool &&
872        \tl_if_empty_p:N \l__pic_region_fill_color_tl
873      }{ \tl_set:Nn \l__pic_region_fill_color_tl {#1} }
874    }
875  \cs_new_protected:Npn \ztool_pic_draw:nw #1#2;
876  {% #1:key-value; #2:coors list (use ';' to end scan just like tikz)
877    \group_begin:
878    \keys_set:nn { ztool / draw / picture / zdraw }{#1}
879    \__region_fill_color_miss:n { gray }
880    \edef\coors@first
```

```
          {
            \exp_last_unbraced:Ne
              \__coors_list_first:w {\tl_trim_spaces:e {#2}}
                \scan_stop:
          }
        \edef\draw@flag
          {
            \tl_map_function:nN {
              \l__pic_draw_vector_bool
              \l__pic_draw_cycle_bool
              \l__pic_region_fill_bool
            } \int_eval:n
          }
        \__@@_pic_put:nnn
          { \__coor_st:V \coors@first + \__coor_st:V \l__pic_draw_shift_tl }
          { \__coor_nd:V \coors@first + \__coor_nd:V \l__pic_draw_shift_tl }
          {
            \__pic_set_line_width:
            \__pic_set_line_color:
            \exp_after:wN \int_case:nnF \exp_after:wN {
              \exp_after:wN \int_from_bin:n \exp_after:wN
                { \draw@flag }
            }{
              {0}{ \__@@_pic_polyline:n    {#2} }
              {1}{ \__@@_pic_polygon:nn {*}{#2} }
              {2}{ \__@@_pic_polygon:ne { }{#2} }
              {3}{ \__@@_pic_polygon:nn {*}{#2} }
              {4}{ \__@@_pic_polyvector:n   {#2} }
              {5}{
                \__pic_set_fill_color:
                \__@@_pic_polygon:nn {*}{#2}
                \__pic_set_line_color:
                \exp_args:Ne \__@@_pic_polyvector:n {#2(\coors@first)}
              }
              {6}{ \exp_args:Ne \__@@_pic_polyvector:n {#2(\coors@first)} }
              {7}{
                \__pic_set_fill_color:
                \__@@_pic_polygon:nn {*}{#2}
                \__pic_set_line_color:
                \exp_args:Ne \__@@_pic_polyvector:n {#2(\coors@first)}
              }
            }{\relax}
          }
        \group_end:
    }
\cs_new:Npn \__coors_list_first:w (#1)#2\scan_stop:
  { #1 }
\NewDocumentCommand{\zdraw}{O{}}
  { \ztool_pic_draw:nw {#1} }


% --> arc / circle
```

```
933  \cs_new:Npn \__@@_pic_arc:nnnn #1#2#3#4
934    {% #1:fill bool; #2:start angle; #3:end angle; #4:radius
935      \arc #1[\fp_eval:n {#2}, \fp_eval:n {#3}]
936        { \fp_eval:n {#4} }
937    }
938  \cs_new:Npn \__@@_pic_circel:nn #1#2
939    {% #1:fill bool; #2:radius
940      \__@@_pic_arc:nnnn {#1}{0}{360}{#2}
941    }


944  % --> circle
945  \keys_define:nn { ztool / draw / picture }
946    {
947      arc   .inherit:n   =   {
948        ztool/draw/picture/line,
949        ztool/draw/picture/region,
950      },
951    }
952  \keys_define:nn { ztool / draw / picture / arc }
953    {
954      radius .fp_set:N   = \l__pic_arc_radius_fp,
955      radius .initial:n = .5,
956      start  .fp_set:N   = \l__pic_arc_start_fp,
957      start  .initial:n = 0,
958      end    .fp_set:N   = \l__pic_arc_end_fp,
959      end    .initial:n = 90,
960    }
961  \prg_generate_conditional_variant:Nnn
962    \bool_if:n { e } { p, T, F, TF }
963  \cs_new_protected:Npn \ztool_pic_arc:nn #1#2
964    {% #1:key-value; #2:coor
965      \group_begin:
966      \keys_set:nn { ztool / draw / picture / arc }{#1}
967      \__region_fill_color_miss:n { gray }
968      \__color_if_valid:VF \l__pic_region_fill_color_tl
969        { \bool_set_false:N \l__pic_region_fill_bool }
970      \exp_last_unbraced:Ne \__@@_pic_put:nnn {\__coor_st_nd:n {#2}}
971        {
972          \__pic_set_line_width:
973          \bool_if:eT \l__pic_region_fill_bool
974            {
975              \__pic_set_fill_color:
976              \exp_args:Ne \__@@_pic_arc:nnnn {*}
977                { \fp_use:N \l__pic_arc_start_fp  }
978                { \fp_use:N \l__pic_arc_end_fp    }
979                { \fp_use:N \l__pic_arc_radius_fp }
980            }
981          % NOTE: border must over the fill
982          \__pic_set_line_color:
983          \exp_args:Ne \__@@_pic_arc:nnnn {}
984            { \fp_use:N \l__pic_arc_start_fp  }
```

```
985        { \fp_use:N \l__pic_arc_end_fp    }
986        { \fp_use:N \l__pic_arc_radius_fp }
987      }
988    \group_end:
989  }
990 \NewDocumentCommand{\zarc}{O{}d()}
991  {% #1:key-value; #2:coor
992    \ztool_pic_arc:nn {#1}{#2}
993  }
994 \NewDocumentCommand{\zcircle}{O{}d()}
995  {
996    \ztool_pic_arc:nn {start=0, end=360, #1}{#2}
997  }


1000 % --> oval / rectangle
1001 % \oval[⟨arc⟩](⟨full-x-width, full-y-width⟩)[⟨part⟩]
1002 % part: (l, r) × (t, b)
1003 \cs_new:Npn \__@@_pic_oval:nnnn #1#2#3#4
1004  {% #1:arc; #2:part; #3:x-width; #4:y-width;
1005    \oval
1006      [\fp_eval:n {#1}]
1007      (\fp_eval:n {#3}, \fp_eval:n {#4})
1008      [ #2 ]
1009  }
1010 \keys_define:nn { ztool / draw / picture }
1011  {
1012    rectangle   .inherit:n  =  {
1013      ztool/draw/picture/line,
1014      ztool/draw/picture/region,
1015    },
1016  }
1017 \keys_define:nn { ztool / draw / picture / rectangle }
1018  {
1019    arc     .fp_set:N  = \l__pic_rec_arc_fp,
1020    arc     .initial:n = 0,
1021  }
1022 \int_new:N \l__pic_rec_quadrant_index_int
1023 \cs_new_protected:Npn \ztool_pic_rectangle:nnn #1#2#3
1024  {% #1:key-value; #2:start coor; #3:end coor;
1025    \group_begin:
1026    \keys_set:nn { ztool / draw / picture / rectangle }{ fill=false }
1027    \keys_set:nn { ztool / draw / picture / rectangle }{ #1 }
1028    \edef\rec@arc   { \fp_use:N  \l__pic_rec_arc_fp    }
1029    \edef\rec@width { \fp_eval:n {\__coor_st:n {#3} - \__coor_st:n {#2}} }
1030    \edef\rec@height{ \fp_eval:n {\__coor_nd:n {#3} - \__coor_nd:n {#2}} }
1031    \__region_fill_color_miss:n { gray }
1032    \__color_if_valid:VF \l__pic_region_fill_color_tl
1033      {
1034        \bool_set_false:N \l__pic_region_fill_bool
1035        \prg_map_break:Nn \l__ztool_pic_rec_fill {}}
1036      }
```

```
%% begin fill rounded rectangle
\__@@_pic_put:nnn {\__coor_st:n {#2}}{\__coor_nd:n {#2}}
  {
    \__pic_set_fill_color:
    \rule
      {\fp_eval:n {\rec@width *\dim_to_decimal:n {\l__pic_unit_dim}}pt}
      {\fp_eval:n {\rec@height*\dim_to_decimal:n {\l__pic_unit_dim}}pt}
  }
\int_set:Nn \l__pic_rec_quadrant_index_int { 0 }
\tl_map_inline:nn
  {
    {\__coor_st:n {#2}+\rec@width-\rec@arc, \__coor_nd:n
      {#2}+\rec@height-\rec@arc}
    {\__coor_st:n {#2}+\rec@arc,            \__coor_nd:n
      {#2}+\rec@height-\rec@arc}
    {\__coor_st:n {#2}+\rec@arc,            \__coor_nd:n {#2}+\rec@arc}
    {\__coor_st:n {#2}+\rec@width-\rec@arc, \__coor_nd:n {#2}+\rec@arc}
  }{
    \int_incr:N \l__pic_rec_quadrant_index_int
    \edef\qu@drant@index{\int_use:N \l__pic_rec_quadrant_index_int}
    \exp_last_unbraced:Ne \__@@_pic_put:nnn {\__coor_st_nd:n {##1}}
      {
        \__color_safe_use:V \l__pic_opacity_color_tl
        \__@@_pic_arc:nnnn {*}
          { (\qu@drant@index-1)*90 }
          { \qu@drant@index*90      }
          { sqrt(2)*\rec@arc        }
        \__pic_set_fill_color:
        \__@@_pic_arc:nnnn {*}{0}{360}{\rec@arc}
      }
  }
%% end fill rounded rectangle
\prg_break_point:Nn \l__ztool_pic_rec_fill { }
\__@@_pic_put:nnn {\__coor_st:n {#2}+\rec@width/2}{\__coor_nd:n
  {#2}+\rec@height/2}
  {
    \__pic_set_line_color:
    \__pic_set_line_width:
    \__@@_pic_oval:nnnn
      { \rec@arc }{ }
      { \rec@width  }
      { \rec@height }
  }
\group_end:
}
\NewDocumentCommand{\zrectangle}{O{}d()d()}
  {
    \ztool_pic_rectangle:nnn { #1 }{#2}{#3}
  }
```

35

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.