

zTool 接口文档

Eureka

2025 年 5 月 20 日

总目录

1	基本介绍	3	5	zdraw	13
2	l3sys-shell	4	6	TODO	16
3	File IO	6	7	zTool 源码	17
4	盒子操作	8		Index	39

1 基本介绍

\LaTeX 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, \LaTeX 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).

本宏包在 Github 上的地址如下:

https://github.com/zongpingding/zTeX_bundle

该仓库中包含本宏集的源码与用户手册; 当前宏集的稳定版本于半年之前发布, 最新的开发版请切换到 “dev” 分支; 本手册适用于当前最新的开发版.

2 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 L^AT_EX 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.

<code>\ztool_shell_escape:n</code> <code>\ztool_shell_escape:e</code> <hr/> Updated: 2024-12-05	<code>\ztool_shell_escape:n {<command>}</code> 当 <code>-shell-escape</code> 参数启用时, 此命令会在 shell 中执行 <code><command></code> , 如果 <code>-shell-escape</code> 参数未启用, 此命令将不会执行任何操作.
---	---

<code>\ztool_shell_mkdir:n</code> <code>\ztool_shell_mkdir:e</code> <hr/> Updated: 2024-12-05	<code>\ztool_shell_mkdir:n {<dir>}</code> 当 <code>-shell-escape</code> 参数启用时, 此命令会创建一个目录 <code><dir></code> , 如果 <code>-shell-escape</code> 参数未启用, 此命令将不会执行任何操作.
---	---

<code>\ztool_shell_cp:nn</code> <code>\ztool_shell_cp:(ee ne en)</code> <hr/> Updated: 2024-12-05	<code>\ztool_shell_cp:nn {<source>}{<target>}</code> 当 <code>-shell-escape</code> 参数启用时, 此命令将把文件 <code><source></code> 复制为文件 <code><target></code> , 如果 <code>-shell-escape</code> 参数未启用, 此命令将不会执行任何操作.
---	--

<code>\ztool_shell_mv:nn</code> <code>\ztool_shell_mv:(ee ne en)</code> <hr/> Updated: 2024-12-05	<code>\ztool_shell_mv:nn {<source>}{<target>}</code> 当 <code>-shell-escape</code> 参数启用时, 此命令将把文件 <code><source></code> 移动到目录 <code><target></code> , 如果 <code>-shell-escape</code> 参数未启用, 此命令将不会执行任何操作.
---	--

<code>\ztool_shell_rm:n</code> <code>\ztool_shell_rm:e</code> <hr/> Updated: 2024-12-05	<code>\ztool_shell_rm:n {<file>}</code> 当 <code>-shell-escape</code> 参数启用时, 此命令将删除文件 <code><file></code> , 如果 <code>-shell-escape</code> 参数未启用, 此命令将不会执行任何操作.
---	--

<code>\ztool_shell_rmdir:n</code> <code>\ztool_shell_rmdir:e</code> <hr/> Updated: 2024-12-05	<code>\ztool_shell_rmdir:n {<dir>}</code> 当 <code>-shell-escape</code> 参数启用时, 此命令将删除目录 <code><dir></code> , 如果 <code>-shell-escape</code> 参数未启用, 此命令将不会执行任何操作.
---	---

<code>\ztool_get_shell_pwd:N</code> <code>\ztool_get_shell_pwd:c</code> <hr/> Updated: 2024-12-05	<code>\ztool_get_shell_pwd:N {<tl>}</code> 当 <code>-shell-escape</code> 参数启用时, 此命令将返回当前的工作目录, 并将其存放在 <code><tl></code> 中, 如果 <code>-shell-escape</code> 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.

3 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 \ztool_file_new:nn {<boolean>}{<file>}
```

Updated: 2024-12-05

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:nnN {<bool>}{<file>}{<seq>}
\ztool_read_file_as_seq:(neN|nnc|nec)
```

Updated: 2024-12-05

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:** 此命令仅在当前组生效.

```
\ztool_gread_file_as_seq:nnN \ztool_read_file_as_seq:nnN {<bool>}{<file>}{<seq>}
\ztool_gread_file_as_seq:(neN|nnc|nec)
```

Updated: 2025-01-05

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:** 此命令的设置是全局有效的.

```
\ztool_append_to_file:nn \ztool_append_to_file:nn {<file>}{<content>}
\ztool_append_to_file:(no|nf|ee)
```

Updated: 2025-01-05

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:nnn {<file>}{<line>}{<content>}
\ztool_replace_file_line:(enn|ene|eee)
```

Updated: 2025-01-05

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:nnn {<file>}{<line>}{<content>}
\ztool_insert_to_file:(nen|nfn|een)
```

Updated: 2025-01-05

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:

例 1

```
\ExplSyntaxOn
\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|
```

4 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.

<code>\ztool_autoset_to_wd_and_ht:nnn</code>	<code>\ztool_autoset_to_wd_and_ht:nn {⟨width⟩}{⟨height⟩}{⟨content⟩}</code>
<code>\ztool_autoset_to_wd_and_ht:(nne een eee)</code>	

Updated: 2025-04-29

此命令用于将 $\langle content \rangle$ 的宽度调整为 $\min(\langle width \rangle, \langle height \rangle)$, 然后排版出来.

<code>\ztool_rotate:nn</code>	<code>\ztool_rotate:nn {⟨angle⟩}{⟨content⟩}</code>
<code>\ztool_rotate:(en ne ee)</code>	此命令用于将 $\langle content \rangle$ 旋转 $\langle angle \rangle$ 度, 然后排版出来.

New: 2025-04-29

<code>\ztool_scale_to_wd:nn</code>	<code>\ztool_scale_to_wd:nn {⟨dim⟩}{⟨content⟩}</code>
<code>\ztool_scale_to_wd:(en ne ee)</code>	此命令用于将 $\langle content \rangle$ 的宽度调整为 $\langle dim \rangle$, 但是不对盒子的高度做任何的调整, 然后排版出来.

New: 2025-04-29

<code>\ztool_scale_to_ht:nn</code>	<code>\ztool_scale_to_ht:nn {⟨dim⟩}{⟨content⟩}</code>
<code>\ztool_scale_to_ht:(en ne ee)</code>	此命令用于将 $\langle content \rangle$ 的高度 + 深度整体调整为 $\langle dim \rangle$, 但是不对盒子的宽度做任何的调整, 然后排版出来.

New: 2025-04-29

<code>\ztool_scale_to_wd_and_ht:nnn</code>	<code>\ztool_scale_to_wd_and_ht:nnn {⟨width⟩}{⟨height⟩}{⟨content⟩}</code>
<code>\ztool_scale_to_wd_and_ht:(nno nne eee)</code>	

New: 2025-04-29

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

<code>\ztool_box_item_align:Nnnn</code>	<code>\ztool_box_item_align:Nnnn ⟨cmd⟩{⟨width⟩}{⟨content⟩}{⟨align⟩}</code>
<code>\ztool_box_item_align:(cnnn Nnno cnno Nnen Nnee)</code>	

Updated: 2025-05-13

此命令用于将 $\langle content \rangle$ 的宽度调整为 $\langle width \rangle$, 然后排版出来, $\langle align \rangle$ 用于控制对齐方式, 可选值有: `left`, `center`, `right`, `scatter`. $\langle cmd \rangle$ 为一个命令, 其接受一个参数, 它将应用到 $\langle content \rangle$ 的每一个 Token 上. **注意:** $\langle content \rangle$ 中的空格会被忽略, 如果需要空格, 请使用 “\ ” 或 “~” 替代.

<code>\ztool_fp_to_rad:n</code>	<code>\ztool_fp_to_rad:n {⟨angle⟩}</code>
---------------------------------	---

New: 2025-05-12

此命令用于将 $\langle angle \rangle$ 从弧度制转换为角度制.

```
\ztool_affine_transformation:Nnnnn \ztool_affine_transformation:Nnnnn <coffin>\{<a>\}\{<b>\}\{<c>\}\{<d>\}
\ztool_affine_transformation:(Neeee|cnnnn|ceeee)
```

New: 2025-05-12

此命令用于对 $\langle coffin \rangle$ 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:

$$\begin{aligned} \Lambda &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} 1 & m \\ 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \\ &= \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} \cos \omega & -\sin \omega \\ \sin \omega & \cos \omega \end{bmatrix} \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}. \quad (4.1) \end{aligned}$$

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 \(4.1\)](#) 写成如下表达式:

$$\Lambda = \mathbf{T}_1(\theta) \cdot \mathbf{T}_1(\phi) \cdot \mathbf{T}_2(S_x) \cdot \mathbf{T}_3(S_y) \cdot \mathbf{T}_1(\omega) \cdot \mathbf{T}_2(s_x) \cdot \mathbf{T}_3(s_y). \quad (4.2)$$

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:

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

s_y 和 m 的求解结果如下:

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

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

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

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

最后我们只需要从右到左将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 L^AT_EX3 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 T_EX 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 pdfT_EX 中的 `\pdfsetmatrix` 命令.

下面的示例展示了如何使用这一章节中的几个 Box 操作命令:

例 2

```

\ExplSyntaxOn
\setlength{\fboxsep}{0pt}
% get dim of content
\dotfill\par
\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\par

% set content to dim
\dotfill\par
Hello,~world|
\ztool_set_to_ht:nn {.5cm} {Hello,~world}|
\ztool_set_to_wd:nn {40pt} {Hello,~world}\par

% scale one dimension
\dotfill\par
\ztool_scale_to_wd:nn {2em}{\fbox{AA}}\par
\ztool_scale_to_wd:nn {2em}{\fbox{AAA}}\par
\ztool_scale_to_wd:nn {2em}{\fbox{AAAAA}}\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}}}}\quad
\ztool_scale_to_ht:nn {2.5em}{\fbox{\vbox{\hbox{A}\hbox{A}\hbox{A}}}}\quad
\ztool_scale_to_ht:nn
{2.5em}{\fbox{\vbox{\hbox{A}\hbox{A}\hbox{A}\hbox{A}\hbox{A}}}}\par

```

```
% box item align
\dotfill\par
\def\boxItemCmd#1{\textcolor{blue}{|#1|}}
\underline{
  \ztool_box_item_align:Nnnn \boxItemCmd{15em}{\{Tom\}{Amy\}{Jennery}}{scatter}
}\par
\underline{
  \ztool_box_item_align:Nnnn \boxItemCmd{15em}{\{Tom\} {Amy\} \ {Jennery}}{center}
}\par

% affine transform
\dotfill\par
\hcoffin_set:Nn \l_tmpa_coffin {\rule{2em}{2em}}
\coffin_typeset:Nnnnn \l_tmpa_coffin {1}{b}{0pt}{0pt}
\ztool_affine_transformation:Nnnnn \l_tmpa_coffin {1}{0}{.5}{1}
\coffin_typeset:Nnnnn \l_tmpa_coffin {1}{b}{0pt}{0pt}
\ExplSyntaxOff
```

.....

7.54619pt 58.58835pt

.....

Hello, worldHello, worldHello, world

.....

AA

AAA

AAAAA

A A A A

.....

|Tom| |Amy| |Jennery|

|Tom||Amy| |Jennery|

.....



5 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：

- L^AT_EX 2_ε 内置 picture 环境；
- pict2e : L^AT_EX 2_ε 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	<code>\begin{zpic}[(key-value)] <draw commands> \end{zpic}</code>
New: 2025-05-13	此环境基于 L ^A T _E X 2 _ε 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	<code>\put (<x, y>) {<content>}</code>
New: 2025-05-13	此命令即为 L ^A T _E X 2 _ε 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	<code>\zline [(key-value)](<coor-1>)(<coor-2>)</code>
New: 2025-05-13	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	<code>\zvector [(key-value)](<coor-1>)(<coor-2>)</code>
New: 2025-05-13	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/	> = $\langle \text{latex pst} \rangle$ 初始值: latex
---------------------	---

此选项用于控制箭头的样式, 默认为 L^AT_EX 样式, 即 `\ltxarrows`; $\langle \text{pst} \rangle$, 即 PsTricks, 对应于 `\pstarrows` 命令.

<code>\zdraw</code>	<code>\zdraw [$\langle \text{key-value} \rangle$]($\langle \text{coord-1} \rangle$)...($\langle \text{coord-n} \rangle$);</code>
---------------------	---

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

注意: 此命令末尾的 “;” 是不能省略的, 否则会报错.

ztool/../../zdraw/vector	<code>vector = $\langle \text{false true} \rangle$</code> 初始值: false
ztool/../../zdraw/cycle	<code>cycle = $\langle \text{false true} \rangle$</code> 初始值: false
ztool/../../zdraw/fill	<code>fill = $\langle \text{false true 颜色} \rangle$</code> 初始值: false
ztool/../../zdraw/shift	<code>shift = {$\langle \text{浮点数}, \text{浮点数} \rangle$}</code> 初始值: {0, 0}

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

<code>\zarc</code>	<code>\zarc [$\langle \text{key-value} \rangle$]($\langle \text{浮点数}, \text{浮点数} \rangle$)</code>
--------------------	---

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

ztool/../../zarc/radius	<code>radius = $\langle \text{浮点数} \rangle$</code> 初始值: .5
ztool/../../zarc/start	<code>start = $\langle \text{浮点数} \rangle$</code> 初始值: 0
ztool/../../zarc/end	<code>end = $\langle \text{浮点数} \rangle$</code> 初始值: 90
ztool/../../zarc/fill	<code>fill = $\langle \text{false true 颜色} \rangle$</code> 初始值: false

$\langle \text{start} \rangle$ 按照逆时针旋转到角度 $\langle \text{end} \rangle$ 结束; $\langle \text{radius} \rangle$ 为圆弧的半径; $\langle \text{fill} \rangle$ 用于设置圆弧的填充颜色.

<code>\zcircle</code>	<code>\zcircle [$\langle \text{key-value} \rangle$]($\langle \text{浮点数}, \text{浮点数} \rangle$)</code>
-----------------------	--

New: 2025-05-13 此命令基于上述的 `\zarc` 命令, 默认情况下将以 ($\langle \text{浮点数}, \text{浮点数} \rangle$) 为圆心绘制一个完整的圆; $\langle \text{key-value} \rangle$ 和上述的 `\zarc` 命令中的 $\langle \text{key-value} \rangle$ 选项相同,

<code>\zrectangle</code>	<code>\zrectangle [$\langle \text{key-value} \rangle$]($\langle \text{coord-1} \rangle$)($\langle \text{coord-2} \rangle$)</code>
--------------------------	--

New: 2025-05-13 此命令用于绘制矩形, ($\langle \text{coord-1} \rangle$) 和 ($\langle \text{coord-2} \rangle$) 为矩形对角线的两个端点坐标; $\langle \text{key-value} \rangle$ 继承自 $\langle \text{parent=ztool/draw/picture/line} \rangle$, 其余的 $\langle \text{key-value} \rangle$ 请参见后续 $\langle \text{parent=ztool/draw/picture/zrectangle} \rangle$.

ztool/../../zrectangle/arc	<code>arc = $\langle \text{浮点数} \rangle$</code> 初始值: 0
ztool/../../zrectangle/fill	<code>fill = $\langle \text{false true 颜色} \rangle$</code> 初始值: false

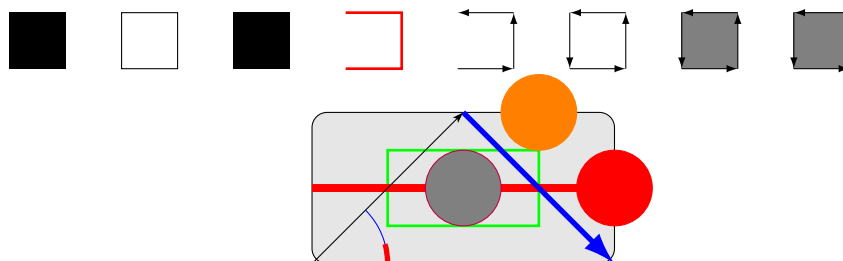
$\langle \text{fill} \rangle$ 用于设置矩形的填充颜色, $\langle \text{arc} \rangle$ 用于设置矩形圆角对应的半径.

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:

例 3

```
\mbox{}\vskip2em
\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 / vector
  \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}
```



6 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.

7 zTool 源码

```
1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 1
2 %% ztool.sty % 2
3 %% Copyright 2024, 2025 Zongping Ding. % 3
4 % % 4
5 % This work may be distributed and/or modified under the conditions of the % 5
6 % LaTeX Project Public License, either version 1.3 of this license or any % 6
7 % later version. % 7
8 % The latest version of this license is in % 8
9 % http://www.latex-project.org/lppl.txt % 9
10 % and version 1.3 or later is part of all distributions of LaTeX % 10
11 % version 2005/12/01 or later. % 11
12 % % 12
13 % This work has the LPPL maintenance status `maintained'. % 13
14 % % 14
15 % The Current Maintainer of this work is Zongping Ding. % 15
16 % % 16
17 % ztool.sty consists of the parts: % 17
18 % l3sys-shell, % 18
19 % file IO, % 19
20 % box manipulation, % 20
21 % zdraw. % 21
22 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 22
23 \NeedsTeXFormat{LaTeX2e} 23
24 \ProvidesExplPackage{ztool}{2025/05/20}{1.0.1}{A~pre-release~tool~package~for~LaTeX} 24
25 25
26 26
27 %%%% l3keys intial patch begin %%%% 27
28 % 1. https://github.com/latex3/latex3/issues/1738 28
29 % 2. https://tex.stackexchange.com/q/742604/294585 29
30 \cs_set_protected:Npn \__keys_initialise:n #1 30
31 { 31
32 \exp_after:wN \__keys_find_key_module:wNN 32
33 \l_keys_path_str \s__keys_stop 33
34 \l_keys_key_tl \l_keys_key_str 34
35 \tl_set_eq:NN \l_keys_key_tl \l_keys_key_str 35
36 \tl_set:Nn \l_keys_value_tl {#1} 36
37 \cs_if_exist:cTF { \c__keys_code_root_str \l_keys_path_str } 37
38 { 38
39 \str_clear:N \l__keys_inherit_str 39
40 \__keys_execute:nn \l_keys_path_str {#1} 40
41 } 41
42 { 42
43 \cs_if_exist:cT 43
44 { \c__keys_inherit_root_str \__keys_parent:o \l_keys_path_str } 44
45 { \__keys_execute_inherit: } 45
46 } 46
47 } 47
48 %%%% l3keys intial patch end %%%% 48
49 49
50 50
```

```

51 % ==> l3sys-shell tool
52 % NOTE: Copy of the original 'l3sys-shell' + some modifications
53 % windows path handle
54 \cs_new:Npn \ztool_sys_path_to_win:N #1
55 {
56   \quark_if_nil:NF #1 {
57     \token_if_eq_meaning:NNTF #1 /
58     { \c_backslash_str }
59     {#1}
60     \ztool_sys_path_to_win:N
61   }
62 }
63 }
64 \cs_new:Npn \ztool_sys_path_to_win:w #1 ~ #2 \q_stop
65 {
66   \ztool_sys_path_to_win:N #1 \q_nil
67   \tl_if_empty:nF {#2}
68   {
69     \c_space_tl
70     \__sys_path_to_win:w #2 \q_stop
71   }
72 }
73 \cs_new:Npn \ztool_sys_path_to_win:n #1
74 {
75   \exp_after:wN \ztool_sys_path_to_win:w \tl_to_str:n {#1} ~ \q_stop
76 }
77 % respective commands
78 \cs_new_protected:Npn \ztool_shell_escape:n #1
79 {
80   \sys_if_shell_unrestricted:T
81   { \sys_shell_now:n {#1} }
82 }
83 \cs_generate_variant:Nn \ztool_shell_escape:n {e}
84 \cs_new_protected:Npe \ztool_shell_mkdir:n #1
85 {
86   \ztool_shell_escape:e {
87     \sys_if_platform_unix:T
88     {mkdir~-p~\exp_not:N \tl_to_str:n {#1}}
89     \sys_if_platform_windows:T
90     {mkdir~ \exp_not:N \ztool_sys_path_to_win:n {#1}}
91   }
92 }
93 \cs_new_protected:Npe \ztool_shell_cp:nn #1#2
94 {
95   \ztool_shell_escape:e {
96     \sys_if_platform_unix:T
97     {
98       cp~-f~ \exp_not:N \tl_to_str:n {#1} ~
99       \exp_not:N \tl_to_str:n {#2}
100     }
101     \sys_if_platform_windows:T
102     {% can NOT use wildcards in CMD

```

```

103         copy~/y~ \exp_not:N \ztool_sys_path_to_win:n {#1} ~
104         \exp_not:N \ztool_sys_path_to_win:n {#2}
105     }
106 }
107 }
108 \cs_new_protected:Npe \ztool_shell_mv:nn #1#2
109 {
110     \ztool_shell_escape:e {
111         \sys_if_platform_unix:T
112         {
113             mv~ \exp_not:N \tl_to_str:n {#1} ~
114             \exp_not:N \tl_to_str:n {#2}
115         }
116         \sys_if_platform_windows:T
117         {
118             copy~/y~ \exp_not:N \ztool_sys_path_to_win:n {#1} ~
119             \exp_not:N \ztool_sys_path_to_win:n {#2}
120             \token_to_str:N & \token_to_str:N &
121             del~/f~/q~ \exp_not:N \ztool_sys_path_to_win:n {#1}
122         }
123     }
124 }
125 \cs_new_protected:Npe \ztool_shell_rm:n #1
126 {
127     \ztool_shell_escape:e {
128         \sys_if_platform_unix:T
129         { rm~-f~ \exp_not:N \tl_to_str:n {#1} }
130         \sys_if_platform_windows:T
131         { del~/f~/q~ \exp_not:N \ztool_sys_path_to_win:n {#1} }
132     }
133 }
134 \cs_new_protected:Npe \ztool_shell_rmdir:n #1
135 {
136     \ztool_shell_mkdir:n {#1}
137     \ztool_shell_escape:e {
138         \sys_if_platform_unix:T
139         { rm~-rf~ \exp_not:N \tl_to_str:n {#1} }
140         \sys_if_platform_windows:T
141         { rmdir~/s~/q~ \exp_not:N \ztool_sys_path_to_win:n {#1} }
142     }
143 }
144 \tl_new:N \l_ztool_shell_tmp_tl
145 \cs_new_protected:Npe \ztool_get_shell_pwd:N #1
146 {
147     \exp_not:N \sys_get_shell:nnN
148     {
149         \sys_if_platform_unix:T { pwd }
150         \sys_if_platform_windows:T { cd }
151     }{
152         \char_set_catcode_other:N \exp_not:N \
153         \char_set_catcode_other:N \exp_not:N \#
154         \char_set_catcode_other:N \exp_not:N \~

```

155	\char_set_catcode_other:N \exp_not:N \%	155
156	\char_set_catcode_space:N \exp_not:N \ %	156
157	\tex_endlinechar:D -1 \scan_stop:	157
158	}	158
159	\exp_not:N \l__ztool_shell_tmp_tl	159
160	\str_set:NV #1 \exp_not:N \l__ztool_shell_tmp_tl	160
161	}	161
162	\cs_new_protected:Npe \ztool_shell_split_ls:nN #1#2	162
163	{	163
164	\exp_not:N \sys_get_shell:nnN	164
165	{	165
166	\sys_if_platform_unix:T { ls~-1~ #1 }	166
167	\sys_if_platform_windows:T { dir~/b~ #1 }	167
168	}{	168
169	\ExplSyntaxOff	169
170	\char_set_catcode_other:N \exp_not:N \	170
171	\char_set_catcode_other:N \exp_not:N \#	171
172	\char_set_catcode_other:N \exp_not:N \~	172
173	\char_set_catcode_other:N \exp_not:N \%	173
174	\char_set_catcode_other:n { 13 }	174
175	}	175
176	\exp_not:N \l__ztool_shell_tmp_tl	176
177	\str_set:NV \exp_not:N \l__sys_tmp_tl \exp_not:N \l__sys_tmp_tl	177
178	\seq_set_split:NnV #2	178
179	{ \char_generate:nn { ``^~M } { 12 } }	179
180	\exp_not:N \l__ztool_shell_tmp_tl	180
181	\seq_pop_right:NN #2 \exp_not:N \l__sys_tmp_tl	181
182	}	182
183	\cs_generate_variant:Nn \ztool_shell_mkdir:n {e}	183
184	\cs_generate_variant:Nn \ztool_shell_cp:nn { ee, ne, en }	184
185	\cs_generate_variant:Nn \ztool_shell_mv:nn { ee, ne, en }	185
186	\cs_generate_variant:Nn \ztool_shell_rm:nn { e, f, o }	186
187	\cs_generate_variant:Nn \ztool_shell_rmdir:nn { e, f, o }	187
188	\cs_generate_variant:Nn \ztool_get_shell_pwd:N {c}	188
189	\cs_generate_variant:Nn \ztool_shell_split_ls:nN {nc}	189
190		190
191		191
192		192
193	<i>% ==> file IO operations</i>	193
194	\ior_new:N \g_ztool_file_read_ior	194
195	\ior_new:N \g_ztool_file_append_ior	195
196	\iow_new:N \g_ztool_file_append_iow	196
197	\tl_new:N \l_ztool_current_line	197
198	\str_clear:N \l_ztikz_file_ori_content_str	198
199	\seq_new:N \l_ztool_file_seq	199
200	\cs_new_protected:Npn \ztool_read_file_as_seq:nnN #1#2#3	200
201	{% #1: bool(True to keep spaces, False to trim); #2: file name; #3: seq	201
202	\seq_clear:N #3	202
203	\file_if_exist:nT {#2}	203
204	{	204
205	\ior_open:Nn \g_ztool_file_read_ior {#2}	205
206	\ior_map_inline:Nn \g_ztool_file_read_ior	206

```

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

```

259	{% #1:file name; #2:line index; #3:replacement	259
260	\seq_clear:N \l_ztool_file_seq	260
261	\file_if_exist:nT {#1}{	261
262	\ior_open:Nn \g_ztool_file_read_ior {#1}	262
263	\ior_str_map_inline:Nn \g_ztool_file_read_ior	263
264	{	264
265	\seq_put_right:Nn \l_ztool_file_seq {##1}	265
266	}	266
267	\ior_close:N \g_ztool_file_read_ior	267
268	\seq_set_item:Nnn \l_ztool_file_seq {#2}	268
269	{ #3 }	269
270	\iow_open:Nn \g_ztool_file_append_iow {#1}	270
271	\iow_now:Ne \g_ztool_file_append_iow	271
272	{ \seq_use:Nn \l_ztool_file_seq {^^J} }	272
273	\iow_close:N \g_ztool_file_append_iow	273
274	}	274
275	}	275
276	\cs_generate_variant:Nn \seq_set_item:Nnn { Nne }	276
277	\cs_generate_variant:Nn \ztool_replace_file_line:nnn { e, ene, eee }	277
278	\cs_new_protected:Npn \ztool_insert_to_file:nnn #1#2#3	278
279	{% #1:file name; #2:line index; #3:content	279
280	\seq_clear:N \l_ztool_file_seq	280
281	\file_if_exist:nT {#1}{	281
282	\ior_open:Nn \g_ztool_file_read_ior {#1}	282
283	\ior_str_map_inline:Nn \g_ztool_file_read_ior	283
284	{	284
285	\seq_put_right:Nn \l_ztool_file_seq {##1}	285
286	}	286
287	\ior_close:N \g_ztool_file_read_ior	287
288	\tl_set:No \l_ztool_current_line	288
289	{ \seq_item:Nn \l_ztool_file_seq {#2} }	289
290	\seq_set_item:Nne \l_ztool_file_seq {#2}	290
291	{ #3^^J\l_ztool_current_line }	291
292	\iow_open:Nn \g_ztool_file_append_iow {#1}	292
293	\iow_now:Ne \g_ztool_file_append_iow	293
294	{ \seq_use:Nn \l_ztool_file_seq {^^J} }	294
295	\iow_close:N \g_ztool_file_append_iow	295
296	}	296
297	}	297
298	\cs_generate_variant:Nn \ztool_insert_to_file:nn { ne, nf, ee }	298
299		299
300		300
301		301
302	% ==> box manipulation tool	302
303	\cs_set:Nn __ztool_leave_vmode:	303
304	{ \ifvmode \leavevmode \fi }	304
305	% catch box dimension	305
306	\box_new:N \l_ztool_measure_box	306
307	\cs_new:Npn \ztool_box_set_to:Nnn #1#2#3 {	307
308	\hbox_set:Nn \l_ztool_measure_box {#3}	308
309	\dim_set:Nn #2 {#1 \l_ztool_measure_box}	309
310	\box_set_eq:NN \l_ztool_measure_box \c_empty_box	310

311	}	311
312	\cs_new:Npn \ztool_box_gset_to:NNn #1#2#3 {	312
313	\hbox_set:Nn \l_ztool_measure_box {#3}	313
314	\dim_gset:Nn #2 {#1 \l_ztool_measure_box}	314
315	\box_set_eq:NN \l_ztool_measure_box \c_empty_box	315
316	}	316
317	\cs_new:Npn \ztool_get_ht:Nn	317
318	{ \ztool_box_set_to:NNn \box_ht:N }	318
319	\cs_new:Npn \ztool_get_ht_plus_dp:Nn	319
320	{ \ztool_box_set_to:NNn \box_ht_plus_dp:N }	320
321	\cs_new:Npn \ztool_get_wd:Nn	321
322	{ \ztool_box_set_to:NNn \box_wd:N }	322
323	\cs_new:Npn \ztool_get_dp:Nn	323
324	{ \ztool_box_set_to:NNn \box_dp:N }	324
325	\cs_new:Npn \ztool_gget_ht:Nn	325
326	{ \ztool_box_gset_to:NNn \box_ht:N }	326
327	\cs_new:Npn \ztool_gget_wd:Nn	327
328	{ \ztool_box_gset_to:NNn \box_wd:N }	328
329	\cs_new:Npn \ztool_gget_dp:Nn	329
330	{ \ztool_box_gset_to:NNn \box_dp:N }	330
331	\cs_generate_variant:Nn \ztool_get_ht:Nn { Ne, ce }	331
332	\cs_generate_variant:Nn \ztool_get_ht_plus_dp:Nn { Ne, ce }	332
333	\cs_generate_variant:Nn \ztool_get_wd:Nn { Ne, ce }	333
334	\cs_generate_variant:Nn \ztool_gget_ht:Nn { Ne, ce }	334
335	\cs_generate_variant:Nn \ztool_gget_wd:Nn { Ne, ce }	335
336		336
337		337
338	<i>%% modify box content</i>	338
339	<i>% 1. auto scale and rotate (smaller of two)</i>	339
340	\cs_new_protected:Npn \ztool_autoset_to_wd_and_ht:nnn #1#2#3	340
341	{% #1:width; #2:height; #3:object	341
342	\hbox_set:Nn \l_tmpa_box {#3}	342
343	\box_autosize_to_wd_and_ht:Nnn \l_tmpa_box {#1}{#2}	343
344	__ztool_leave_vmode:	344
345	\box_use:N \l_tmpa_box	345
346	}	346
347	\cs_new_protected:Npn \ztool_rotate:nn #1#2	347
348	{% #1:angle; #2:object	348
349	\hbox_set:Nn \l_tmpa_box {#2}	349
350	\box_rotate:Nn \l_tmpa_box {#1}	350
351	__ztool_leave_vmode:	351
352	\box_use:N \l_tmpa_box	352
353	}	353
354	\cs_generate_variant:Nn \ztool_rotate:nn { e, ne, ee }	354
355	\cs_generate_variant:Nn \ztool_autoset_to_wd_and_ht:nnn { nne, een, eee }	355
356		356
357	<i>% 2. width/height scale to same time</i>	357
358	\cs_new_protected:Npn \ztool_set_to_wd:nn #1#2	358
359	{% #1:width; #2:object	359
360	\hbox_set:Nn \l_tmpa_box {#2}	360
361	\box_resize_to_wd:Nn \l_tmpa_box {#1}	361
362	__ztool_leave_vmode:	362


```

363 \box_use:N \l_tmpa_box
364 }
365 \cs_new_protected:Npn \ztool_set_to_ht:nn #1#2
366 { % #1:height; #2:object
367 \hbox_set:Nn \l_tmpa_box {#2}
368 \box_resize_to_ht:Nn \l_tmpa_box {#1}
369 \__ztool_leave_vmode:
370 \box_use:N \l_tmpa_box
371 }
372 \cs_generate_variant:Nn \ztool_set_to_wd:nn { e, ne, ee }
373 \cs_generate_variant:Nn \ztool_set_to_ht:nn { e, ne, ee }
374
375 % 3. only scale one dimension
376 % NOTE: if boxwdcontent ≤ given dim, no manipulation
377 \cs_new_protected:Npn \ztool_scale_to_wd:nn #1#2
378 {
379 \hbox_set:Nn \l_tmpa_box {#2}
380 \dim_set:Nn \l_tmpa_dim { \box_wd:N \l_tmpa_box }
381 \fp_set:Nn \l_tmpa_fp
382 {
383 \fp_eval:n { min(1, \dim_ratio:nn {#1}{\l_tmpa_dim}) }
384 }
385 \box_scale:Nnn \l_tmpa_box {\l_tmpa_fp}{1}
386 \__ztool_leave_vmode:
387 \box_use:N \l_tmpa_box
388 }
389 \cs_new_protected:Npn \ztool_scale_to_ht:nn #1#2
390 { % take depth into consideration
391 \hbox_set:Nn \l_tmpa_box {#2}
392 \dim_set:Nn \l_tmpa_dim { \box_ht_plus_dp:N \l_tmpa_box }
393 \fp_set:Nn \l_tmpa_fp
394 {
395 \fp_eval:n { min(1, \dim_ratio:nn {#1}{\l_tmpa_dim}) }
396 }
397 \box_scale:Nnn \l_tmpa_box {1}{\l_tmpa_fp}
398 \__ztool_leave_vmode:
399 \box_use:N \l_tmpa_box
400 }
401 \cs_new_protected:Npn \ztool_scale_to_wd_and_ht:nnn #1#2#3
402 { % take depth into consideration
403 \hbox_set:Nn \l_tmpa_box {#3}
404 \dim_set:Nn \l_tmpa_dim { \box_wd:N \l_tmpa_box }
405 \dim_set:Nn \l_tmpb_dim { \box_ht_plus_dp:N \l_tmpa_box }
406 \fp_set:Nn \l_tmpa_fp
407 {
408 \fp_eval:n { min(1, \dim_ratio:nn {#1}{\l_tmpa_dim}) }
409 }
410 \fp_set:Nn \l_tmpb_fp
411 {
412 \fp_eval:n { min(1, \dim_ratio:nn {#2}{\l_tmpb_dim}) }
413 }
414 \box_scale:Nnn \l_tmpa_box {\l_tmpa_fp}{\l_tmpb_fp}

```



```

415 \__ztool_leave_vmode:
416 \box_use:N \l_tmpa_box
417 }
418 \cs_generate_variant:Nn \ztool_scale_to_wd:nn { e, ne, ee }
419 \cs_generate_variant:Nn \ztool_scale_to_ht:nn { e, ne, ee }
420 \cs_generate_variant:Nn \ztool_scale_to_wd_and_ht:nnn { nne, nno, eee }
421
422
423 %% box content align
424 \seq_new:N \l__ztool_boxitem_seq
425 \cs_set_protected:Npn \ztool_box_item_align:Nnnn #1#2#3#4
426 { % #1:cmd, #2:width, #3:object, #4:align format(left, right, scatter, center)
427   \hb@xt@#2{
428     \tl_map_inline:nn {#3}
429     {
430       \seq_put_right:No \l__ztool_boxitem_seq {\exp_not:N #1{##1}}
431     }
432     \str_case:nnF { #4 }
433     {
434       { left }{ \seq_use:Nn \l__ztool_boxitem_seq {} \hfill }
435       { right }{ \hfill\seq_use:Nn \l__ztool_boxitem_seq {} }
436       { scatter }{ \seq_use:Nn \l__ztool_boxitem_seq {} \hfill }
437       { center }{ \hfill\seq_use:Nn \l__ztool_boxitem_seq {} \hfill }
438       { tower }
439       {
440         \edef\seq@count{\seq_count:N \l__ztool_boxitem_seq}
441         \seq_map_indexed_inline:Nn \l__ztool_boxitem_seq
442         { % ##1: index, ##2: content
443           %% Method II: plain
444           \edef\item@width{\dim_eval:n {#2/(\seq@count+1)}}
445           \hskip\item@width\clap{##2}
446           }\hskip\item@width\hss
447         }
448       { custom }
449       {
450         \def\total@width{#2}
451         \def\align@cmd{#1}
452         \def\align@object{#3}
453         \def\align@format{#4}
454         \tl_use:N \l__ztex_boxitem_align_custom_tl
455       }
456     }{\relax}
457   }
458   \seq_clear:N \l__ztool_boxitem_seq
459 }
460 \cs_generate_variant:Nn \ztool_box_item_align:Nnnn { c, Nnno, cnno, Nne, Nnee }
461
462
463 %% affine transformation
464 % REF:
465 % 1. https://math.stackexchange.com/a/3521141/1235323
466 % 2. https://math.stackexchange.com/a/281087/1235323

```

```

467 \cs_new:Npn \ztool_fp_to_rad:n #1
468 { \fp_eval:n {#1/pi*180} }
469 \cs_new:Npn \ztool_matrix_det:nnnn #1#2#3#4
470 {
471     \fp_eval:n { #1*#4 - #2*#3 }
472 }
473 % (translation) + x-scale + y-scale + rotate
474 \fp_new:N \g_affine_precision_fp
475 \fp_set:Nn \g_affine_precision_fp {0.0001}
476 \fp_new:N \l__affine_@@_a_fp
477 \fp_new:N \l__affine_@@_b_fp
478 \fp_new:N \l__affine_@@_c_fp
479 \fp_new:N \l__affine_@@_d_fp
480 \msg_set:nnn { ztool }{affine-det-zero}
481 {
482     current~determination~of~the~affine~transformation~
483     matrix~equals~to~zero,~give~up~this~transformation
484 }
485
486 \coffin_new:N \l__affine_trans_coffin
487 \cs_generate_variant:Nn \coffin_typeset:Nnnnn { Nxxxx }
488 \cs_new:Npn \ztool_affine_transformation:Nnnnn #1#2#3#4#5
489 {% #1:box; #2:a11; #3:a21; #4:a12; #5:a22.
490     \fp_compare:nNnT
491         { abs(\ztool_matrix_det:nnnn {#2}{#3}{#4}{#5}) }
492         < { \g_affine_precision_fp }
493         { \prg_map_break:Nn \l__affine_matrix_det_zero
494           { \msg_warning:nn { ztool }{affine-det-zero} }}
495     \fp_set:Nn \l__affine_@@_a_fp {#2}
496     \fp_set:Nn \l__affine_@@_b_fp {#3}
497     \fp_set:Nn \l__affine_@@_c_fp {#4}
498     \fp_set:Nn \l__affine_@@_d_fp {#5}
499     \__box_affine_transform:N #1
500     \prg_break_point:Nn \l__affine_matrix_det_zero { }
501     \coffin_typeset:Nxxxx \l__affine_trans_coffin
502     { \l__ztool_affine_pole_a_tl }{ \l__ztool_affine_pole_b_tl }
503     { \l__ztool_affine_xoffset_dim }{ \l__ztool_affine_yoffset_dim }
504 }
505 \cs_generate_variant:Nn \ztool_affine_transformation:Nnnnn { Neeee, cnnnn, ceeee }
506 \cs_new:Npn \__box_affine_transform:N #1
507 {
508     % transform debug
509     \bool_if:NT \g_ztool_affine_debug_bool
510     {
511         \noindent\dotfill\[\begin{bmatrix}
512             \fp_use:N \l__affine_@@_a_fp & \fp_use:N \l__affine_@@_c_fp\\
513             \fp_use:N \l__affine_@@_b_fp & \fp_use:N \l__affine_@@_d_fp
514         \end{bmatrix}\]
515     }
516     % get affine parameters
517     \__affine_trans_get_sx:
518     \__affine_trans_get_theta:

```

519	__affine_trans_get_sy:	519
520	__affine_trans_get_Sx:	520
521	__affine_trans_get_Sy:	521
522	__affine_trans_get_phi:	522
523	__affine_trans_get_omega:	523
524	% start transform box/coffin	524
525	\coffin_scale:Nnn #1	525
526	{ \l__box_affine_sx_fp }	526
527	{ \l__box_affine_sy_fp }	527
528	\coffin_rotate:Nn #1	528
529	{ \ztool_fp_to_rad:n {\l__box_affine_omega_fp} }	529
530	\coffin_scale:Nnn #1	530
531	{ \l__box_affine_Sx_fp }	531
532	{ \l__box_affine_Sy_fp }	532
533	\coffin_rotate:Nn #1	533
534	{ \ztool_fp_to_rad:n {\l__box_affine_phi_fp} }	534
535	\coffin_rotate:Nn #1	535
536	{ \ztool_fp_to_rad:n {\l__box_affine_theta_fp} }	536
537	}	537
538	\keys_define:nn { ztool / affine }	538
539	{	539
540	debug .bool_gset:N = \g_ztool_affine_debug_bool,	540
541	debug .initial:n = false,	541
542	debug .default:n = true,	542
543	pole-1 .tl_set:N = \l__ztool_affine_pole_a_tl,	543
544	pole-2 .tl_set:N = \l__ztool_affine_pole_b_tl,	544
545	pole-1 .initial:n = { l },	545
546	pole-2 .initial:n = { b },	546
547	xoffset .dim_set:N = \l__ztool_affine_xoffset_dim,	547
548	yoffset .dim_set:N = \l__ztool_affine_yoffset_dim,	548
549	xoffset .initial:n = { Opt },	549
550	yoffset .initial:n = { Opt },	550
551	}	551
552	\NewDocumentCommand{\ztoolboxaffine}{0{ }m>{\SplitList{,}}m}	552
553	{% #1:key-value; #2:content; #3:matrix.	553
554	\group_begin:	554
555	\keys_set:nn { ztool / affine } {#1}	555
556	\hcoffin_set:Nn \l__affine_trans_coffin {#2}	556
557	\ztool_affine_transformation:Nnnnn \l__affine_trans_coffin #3	557
558	\group_end:	558
559	}	559
560	% internal affine transform functions	560
561	\cs_new:Nn __ztool_affine_debug_fp:N	561
562	{	562
563	\bool_if:NTF \g_ztool_affine_debug_bool	563
564	{ \string #1 % \show #1	564
565	~::~\fp_use:N #1\\	565
566	}{ \relax }	566
567	}	567
568	\fp_new:N \l__box_affine_sx_fp	568
569	\cs_new:Nn __affine_trans_get_sx:	569
570	{	570

```

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

```

```

621 \ztool_affine_debug_fp:N \l__box_affine_m_fp 621
622 } 622
623 \fp_new:N \l__box_affine_Sx_fp 623
624 \fp_new:N \l__box_affine_Sy_fp 624
625 \cs_new:Nn \__affine_trans_get_Sx: 625
626 { 626
627 \__affine_trans_get_m: 627
628 \fp_set:Nn \l__box_affine_Sx_fp 628
629 { sqrt(\l__box_affine_m_fp^2/4 + 1) - \l__box_affine_m_fp/2 } 629
630 \ztool_affine_debug_fp:N \l__box_affine_Sx_fp 630
631 } 631
632 \cs_new:Nn \__affine_trans_get_Sy: 632
633 { 633
634 \fp_set:Nn \l__box_affine_Sy_fp 634
635 { sqrt(\l__box_affine_m_fp^2/4 + 1) + \l__box_affine_m_fp/2 } 635
636 \ztool_affine_debug_fp:N \l__box_affine_Sy_fp 636
637 } 637
638 \fp_new:N \l__box_affine_phi_fp 638
639 \fp_new:N \l__box_affine_omega_fp 639
640 \cs_new:Nn \__affine_trans_get_phi: 640
641 { 641
642 \fp_set:Nn \l__box_affine_phi_fp 642
643 { -pi/4 - 1/2*atan(\l__box_affine_m_fp/2) } 643
644 \ztool_affine_debug_fp:N \l__box_affine_phi_fp 644
645 } 645
646 \cs_new:Nn \__affine_trans_get_omega: 646
647 { 647
648 \fp_set:Nn \l__box_affine_omega_fp 648
649 { pi/4 - 1/2*atan(\l__box_affine_m_fp/2) } 649
650 \ztool_affine_debug_fp:N \l__box_affine_omega_fp 650
651 } 651
652 652
653 653
654 654
655 % ==> ztool draw (based on package 'pict2e' and 'picture' env) 655
656 \RequirePackage{pict2e} 656
657 \cs_new:Npn \__@_begin_picture:nnnn #1#2#3#4 657
658 { \begin{picture} 658
659 (\fp_eval:n {#1}, \fp_eval:n {#2}) 659
660 (\fp_eval:n {-#3}, \fp_eval:n {-#4}) } 660
661 \cs_new:Nn \__@_end_picture: 661
662 { \end{picture} } 662
663 \cs_new:Npn \__@_pic_put:nnn #1#2#3 663
664 { \put(\fp_eval:n {#1}, \fp_eval:n {#2}){ #3 } } 664
665 \cs_generate_variant:Nn \__@_begin_picture:nnnn { VVVV, eeee } 665
666 \cs_generate_variant:Nn \__@_pic_put:nnn { VVV, een } 666
667 667
668 % picture environment alias 668
669 \keys_define:nn { ztool / draw / picture } 669
670 { 670
671 unit .dim_set:N = \l__pic_unit_dim, 671
672 unit .initial:n = { 1cm }, 672

```

```

673     width .fp_set:N = \l__pic_width_fp,
674     width .initial:n = 0,
675     height .fp_set:N = \l__pic_height_fp,
676     height .initial:n = 0,
677     xoffset .fp_set:N = \l__pic_xoffset_fp,
678     xoffset .initial:n = 0,
679     yoffset .fp_set:N = \l__pic_yoffset_fp,
680     yoffset .initial:n = 0,
681     opacity-color .tl_set:N = \l__pic_opacity_color_tl,
682     opacity-color .initial:n = { white },
683 }
684 \NewDocumentEnvironment{zpic}{0{}}
685 {
686     \group_begin:
687     \keys_set:nn { ztool / draw / picture } {#1}
688     \setlength\unitlength{ \l__pic_unit_dim }
689     \@@_begin_picture:VVVV
690     \l__pic_width_fp \l__pic_height_fp
691     \l__pic_xoffset_fp\l__pic_yoffset_fp
692 }{
693     \@@_end_picture:
694     \group_end:
695 }
696
697
698 % picture commands alias
699 \cs_new:Npn \__coord_st:n #1
700 { \clist_item:nn {#1}{1} }
701 \cs_new:Npn \__coord_nd:n #1
702 { \clist_item:nn {#1}{2} }
703 \cs_new:Npn \__coord_rd:n #1#2
704 { \clist_item:nn {#1}{3} }
705 \cs_new:Npn \__coord_st_nd:n #1
706 {
707     {\clist_item:nn {#1}{1}}
708     {\clist_item:nn {#1}{2}}
709 }
710 \cs_new:Npn \__coord_st_nd_rd:n #1
711 {
712     {\clist_item:nn {#1}{1}}
713     {\clist_item:nn {#1}{2}}
714     {\clist_item:nn {#1}{3}}
715 }
716 \cs_generate_variant:Nn \__coord_st:n { V, e }
717 \cs_generate_variant:Nn \__coord_nd:n { V, e }
718 \cs_generate_variant:Nn \__coord_rd:n { V, e }
719 \cs_generate_variant:Nn \__coord_st_nd:n { V, e }
720 \cs_generate_variant:Nn \__coord_st_nd_rd:n { V, e }
721
722 \bool_new:N \l__ztool_invalid_color_bool
723 \cs_new:Npn \__color_safe_use:n #1
724 {

```

```

725     \_color_if_valid:nT {#1}
726     { \color{#1} }
727 }
728 \prg_new_conditional:Npnn \_color_if_valid:n #1 {p, T, F, TF}
729 {
730     \def\ztool@targer@color{#1}
731     \def\ztool@color@none{none}
732     \bool_if:eTF
733     {
734         \tl_if_empty_p:e {#1} ||
735         \tl_if_eq_p:NN \ztool@color@none \ztool@targer@color
736     }{ \prg_return_false: }
737     { \prg_return_true: }
738 }
739 \prg_generate_conditional_variant:Nnn \_color_if_valid:n
740 { V, e }{ p, T, F, TF }
741 \cs_generate_variant:Nn \_color_safe_use:n { V, e }
742
743
744 % --> line/vector
745 \fp_new:N \l__draw_vector_slope_fp
746 \fp_new:N \l__draw_vector_normal_fp
747 \fp_new:N \l__draw_vector_xsep_fp
748 \cs_new:Npn \_@@_pic_line:nnn #1#2#3
749     {% #1:x; #2:y; #3:x-distance NOT the length
750     \line(\fp_eval:n {#1}, \fp_eval:n {#2})
751     { \fp_eval:n {#3} }
752 }
753 \cs_new:Npn \_@@_pic_vector:nnn #1#2#3
754     {% #1:x; #2:y; #3:x-distance NOT the length
755     \vector(\fp_eval:n {#1}, \fp_eval:n {#2})
756     { \fp_eval:n {#3} }
757 }
758 \keys_define:nn { ztool / draw / picture / line }
759 {
760     draw .tl_set:N = \l__pic_line_draw_color_tl,
761     draw .initial:n = { black },
762     % color .meta:n = draw = #1 , % alias for 'draw'
763     width .dim_set:N = \l__pic_line_width_dim,
764     width .initial:n = { .4pt },
765     dash .bool_set:N = \l__pic_line_dash_bool,
766     dash .initial:n = { false },
767 }
768 \cs_new_protected:Nn \_pic_set_line_width:
769 {
770     \linethickness{ \l__pic_line_width_dim }
771 }
772 \cs_new_protected:Nn \_pic_set_line_color:
773 {
774     \_color_safe_use:V \l__pic_line_draw_color_tl
775 }
776 \cs_new_protected:Nn \_pic_set_fill_color:

```

```

777 {
778     \__color_safe_use:V \l__pic_region_fill_color_tl
779 }
780 \def\z@pic@vector@style{\ltxarrows}
781 \keys_define:nn { ztool / draw / picture }
782 {
783     vector .inherit:n = { ztool/draw/picture/line },
784 }
785 \keys_define:nn { ztool / draw / picture / vector }
786 {
787     > .choice:,
788     > / latex .code:n = {\def\z@pic@vector@style{\ltxarrows}},
789     > / pst .code:n = {\def\z@pic@vector@style{\pstarrows}},
790     > / unknown .code:n =
791     {
792         \msg_set:nnn { ztool }{unknown-arrow-style}
793         { Unknown~arrow~style,~use~'latex'~or~'pst'. }
794         \msg_error:nn { ztool }{unknown-arrow-style}
795     }
796 }
797 \cs_new_protected:Npn \ztool_pic_line_vector:nnnn #1#2#3#4
798 {% #1:line/vector; #2:key-value; #3:start coor; #4:end coor;
799     \group_begin:
800     \keys_set:nn { ztool / draw / picture / #1 }{#2}
801     \fp_set:Nn \l__draw_vector_slope_fp
802     { (\__coor_nd:n {#4} - \__coor_nd:n {#3})
803       / (\__coor_st:n {#4} - \__coor_st:n {#3}) }
804     \fp_set:Nn \l__draw_vector_xsep_fp
805     { abs(\__coor_st:n {#4} - \__coor_st:n {#3}) }
806     \z@pic@vector@style
807     \__pic_set_line_width:
808     \exp_last_unbraced:Ne \__@@_pic_put:nnn {\__coor_st_nd:n {#3}}
809     {
810         \__pic_set_line_color:
811         \cs:w \__@@_pic_#1:nnn\cs_end: {1}
812         { \l__draw_vector_slope_fp }
813         { \l__draw_vector_xsep_fp }
814     }
815     \group_end:
816 }
817 \NewDocumentCommand{\zline}{0{}d()d()}
818 {
819     \ztool_pic_line_vector:nnnn {line}{#1}{#2}{#3}
820 }
821 \NewDocumentCommand{\zvector}{0{}d()d()}
822 {
823     \ztool_pic_line_vector:nnnn {vector}{#1}{#2}{#3}
824 }
825
826
827 % --> \zdraw -- similar to \tikz command in tikz
828 % NOTE: these line/vector commands are identical to

```



```

829 % 1. \Line (x1,y1)(x2,y2), \Vector(x1,y1)(x2,y2) 829
830 % 2. \polyline (x1,y1)⋯(xn,yn), \polyvector(x1,y1)⋯(xn,yn) 830
831 % 3. \polygon (x1,y1)⋯(xn,yn), when set 'cycle', 831
832 % \polygon*(x1,y1)⋯(xn,yn), when set 'fill' (auto cycle). 832
833 % 4. Trim leading space after '\polygon' or '*' to avoid error !! 833
834 \cs_new:Npn \__@@_pic_Line:nnnn #1#2#3#4 834
835 { \Line (#1, #2)(#3, #4) } 835
836 \cs_new:Npn \__@@_pic_Vector:nnnn #1#2#3#4 836
837 { \Vector (#1, #2)(#3, #4) } 837
838 \cs_new:Npn \__@@_pic_polyline:n #1 838
839 { 839
840 \tl_set:Nc \l_tmpa_tl {\tl_trim_spaces:e {#1}} 840
841 \exp_last_unbraced:NV \polyline \l_tmpa_tl 841
842 } 842
843 \cs_new:Npn \__@@_pic_polyvector:n #1 843
844 { 844
845 \tl_set:Nc \l_tmpa_tl {\tl_trim_spaces:e {#1}} 845
846 \exp_last_unbraced:NV \polyvector \l_tmpa_tl 846
847 } 847
848 \cs_new:Npn \__@@_pic_polygon:nn #1#2 848
849 { 849
850 \tl_set:Nc \l_tmpa_tl {\tl_trim_spaces:e {#1}} 850
851 \tl_set:Nc \l_tmpb_tl {\tl_trim_spaces:e {#2}} 851
852 \tl_set:Nc \l_tmpa_tl { \l_tmpa_tl\l_tmpb_tl } 852
853 \exp_last_unbraced:NV \polygon \l_tmpa_tl 853
854 } 854
855 \cs_generate_variant:Nn \__@@_pic_polygon:nn { nV, ne } 855
856 \tl_new:N \l__pic_region_fill_color_tl 856
857 \bool_new:N \l__pic_region_fill_bool 857
858 \keys_define:nn { ztool / draw / picture / region } 858
859 { 859
860 fill .choices:nn = { true, false }{ 860
861 \use:c { bool_set_ \l_keys_choice_tl :N } 861
862 \l__pic_region_fill_bool 862
863 }, 863
864 fill .initial:n = { false }, 864
865 fill .default:n = { true }, 865
866 fill / unknown .code:n = { 866
867 \tl_if_empty:eF \l_keys_value_tl 867
868 { \bool_set_true:N \l__pic_region_fill_bool } 868
869 \tl_set:Nc \l__pic_region_fill_color_tl { \l_keys_value_tl } 869
870 }, 870
871 } 871
872 \keys_define:nn { ztool / draw / picture } 872
873 { 873
874 zdraw .inherit:n = { 874
875 ztool/draw/picture/line, 875
876 ztool/draw/picture/vector, 876
877 ztool/draw/picture/region, 877
878 }, 878
879 } 879
880 \keys_define:nn { ztool / draw / picture / zdraw } 880

```

```

881 {
882     vector .bool_set:N = \l__pic_draw_vector_bool,
883     vector .initial:n = { false },
884     cycle .bool_set:N = \l__pic_draw_cycle_bool,
885     cycle .initial:n = { false },
886     shift .tl_set:N = \l__pic_draw_shift_tl,
887     shift .initial:n = { 0, 0 },
888 }
889 \cs_new:Npn \__region_fill_color_miss:n #1
890 {
891     \bool_if:eT {
892         \l__pic_region_fill_bool &&
893         \tl_if_empty_p:N \l__pic_region_fill_color_tl
894     }{ \tl_set:Nn \l__pic_region_fill_color_tl {#1} }
895 }
896 \cs_new_protected:Npn \ztool_pic_draw:nw #1#2;
897 {% #1:key-value; #2:coors list (use ';' to end scan just like tikz)
898     \group_begin:
899     \keys_set:nn { ztool / draw / picture / zdraw }{#1}
900     \__region_fill_color_miss:n { gray }
901     \edef\coors@first
902     {
903         \exp_last_unbraced:Ne
904         \__coors_list_first:w {\tl_trim_spaces:e {#2}}
905         \scan_stop:
906     }
907     \edef\draw@flag
908     {
909         \tl_map_function:nN {
910             \l__pic_draw_vector_bool
911             \l__pic_draw_cycle_bool
912             \l__pic_region_fill_bool
913         } \int_eval:n
914     }
915     \__@@_pic_put:nnn
916     { \__coor_st:V \coors@first + \__coor_st:V \l__pic_draw_shift_tl }
917     { \__coor_nd:V \coors@first + \__coor_nd:V \l__pic_draw_shift_tl }
918     {
919         \__pic_set_line_width:
920         \__pic_set_line_color:
921         \exp_after:wN \int_case:nnF \exp_after:wN {
922             \exp_after:wN \int_from_bin:n \exp_after:wN
923             { \draw@flag }
924         }{
925             {0}{ \__@@_pic_polyline:n {#2} }
926             {1}{ \__@@_pic_polygon:nn {*}{#2} }
927             {2}{ \__@@_pic_polygon:ne { }{#2} }
928             {3}{ \__@@_pic_polygon:nn {*}{#2} }
929             {4}{ \__@@_pic_polyvector:n {#2} }
930             {5}{
931                 \__pic_set_fill_color:
932                 \__@@_pic_polygon:nn {*}{#2}

```

933	__pic_set_line_color:	933
934	\exp_args:Ne __@@_pic_polyvector:n {#2(\coors@first)}	934
935	}	935
936	{6}{ \exp_args:Ne __@@_pic_polyvector:n {#2(\coors@first)} }	936
937	{7}{	937
938	__pic_set_fill_color:	938
939	__@@_pic_polygon:nn {*}{#2}	939
940	__pic_set_line_color:	940
941	\exp_args:Ne __@@_pic_polyvector:n {#2(\coors@first)}	941
942	}	942
943	}{\relax}	943
944	}	944
945	\group_end:	945
946	}	946
947	\cs_new:Npn __coors_list_first:w (#1)#2\scan_stop:	947
948	{ #1 }	948
949	\NewDocumentCommand{\zdraw}{0{}}	949
950	{ \ztool_pic_draw:nw {#1} }	950
951		951
952		952
953	% --> arc / circle	953
954	\cs_new:Npn __@@_pic_arc:nnnn #1#2#3#4	954
955	{% #1:fill bool; #2:start angle; #3:end angle; #4:radius	955
956	\arc #1[\fp_eval:n {#2}, \fp_eval:n {#3}]	956
957	{ \fp_eval:n {#4} }	957
958	}	958
959	\cs_new:Npn __@@_pic_circel:nn #1#2	959
960	{% #1:fill bool; #2:radius	960
961	__@@_pic_arc:nnnn {#1}{0}{360}{#2}	961
962	}	962
963		963
964		964
965	% --> circle	965
966	\keys_define:nn { ztool / draw / picture }	966
967	{	967
968	arc .inherit:n = {	968
969	ztool/draw/picture/line,	969
970	ztool/draw/picture/region,	970
971	},	971
972	}	972
973	\keys_define:nn { ztool / draw / picture / arc }	973
974	{	974
975	radius .fp_set:N = \l__pic_arc_radius_fp,	975
976	radius .initial:n = .5,	976
977	start .fp_set:N = \l__pic_arc_start_fp,	977
978	start .initial:n = 0,	978
979	end .fp_set:N = \l__pic_arc_end_fp,	979
980	end .initial:n = 90,	980
981	}	981
982	\prg_generate_conditional_variant:Nnn	982
983	\bool_if:n { e } { p, T, F, TF }	983
984	\cs_new_protected:Npn \ztool_pic_arc:nn #1#2	984

```

985   {% #1:key-value; #2:coord }
986   \group_begin:
987   \keys_set:nn { ztool / draw / picture / arc }{#1}
988   \__region_fill_color_miss:n { gray }
989   \__color_if_valid:VF \l__pic_region_fill_color_tl
990   { \bool_set_false:N \l__pic_region_fill_bool }
991   \exp_last_unbraced:Ne \__@@_pic_put:nnn {\__coord_st_nd:n {#2}}
992   {
993     \__pic_set_line_width:
994     \bool_if:eT \l__pic_region_fill_bool
995     {
996       \__pic_set_fill_color:
997       \exp_args:Ne \__@@_pic_arc:nnnn {*}
998       { \fp_use:N \l__pic_arc_start_fp }
999       { \fp_use:N \l__pic_arc_end_fp }
1000      { \fp_use:N \l__pic_arc_radius_fp }
1001    }
1002    % NOTE: border must over the fill
1003    \__pic_set_line_color:
1004    \exp_args:Ne \__@@_pic_arc:nnnn {}
1005    { \fp_use:N \l__pic_arc_start_fp }
1006    { \fp_use:N \l__pic_arc_end_fp }
1007    { \fp_use:N \l__pic_arc_radius_fp }
1008  }
1009  \group_end:
1010 }
1011 \NewDocumentCommand{\zarc}{0{}d()}
1012 {% #1:key-value; #2:coord }
1013 \ztool_pic_arc:nn {#1}{#2}
1014 }
1015 \NewDocumentCommand{\zcircle}{0{}d()}
1016 {
1017   \ztool_pic_arc:nn {start=0, end=360, #1}{#2}
1018 }
1019
1020
1021 % --> oval / rectangle
1022 % \oval[arc](<full-x-width, full-y-width>)[<part>]
1023 % part: (l, r) × (t, b)
1024 \cs_new:Npn \__@@_pic_oval:nnnn #1#2#3#4
1025 {% #1:arc; #2:part; #3:x-width; #4:y-width;
1026   \oval
1027   [\fp_eval:n {#1}]
1028   (\fp_eval:n {#3}, \fp_eval:n {#4})
1029   [ #2 ]
1030 }
1031 \keys_define:nn { ztool / draw / picture }
1032 {
1033   rectangle .inherit:n = {
1034     ztool/draw/picture/line,
1035     ztool/draw/picture/region,
1036   },

```

1037	}	1037
1038	\keys_define:nn { ztool / draw / picture / rectangle }	1038
1039	{	1039
1040	arc .fp_set:N = \l__pic_rec_arc_fp,	1040
1041	arc .initial:n = 0,	1041
1042	}	1042
1043	\int_new:N \l__pic_rec_quadrant_index_int	1043
1044	\cs_new_protected:Npn \ztool_pic_rectangle:nnn #1#2#3	1044
1045	{% #1:key-value; #2:start coor; #3:end coor;	1045
1046	\group_begin:	1046
1047	\keys_set:nn { ztool / draw / picture / rectangle }{ fill=false }	1047
1048	\keys_set:nn { ztool / draw / picture / rectangle }{ #1 }	1048
1049	\edef\rec@arc { \fp_use:N \l__pic_rec_arc_fp }	1049
1050	\edef\rec@width { \fp_eval:n {__coor_st:n {#3} - __coor_st:n {#2}} }	1050
1051	\edef\rec@height{ \fp_eval:n {__coor_nd:n {#3} - __coor_nd:n {#2}} }	1051
1052	__region_fill_color_miss:n { gray }	1052
1053	__color_if_valid:VF \l__pic_region_fill_color_tl	1053
1054	{	1054
1055	\bool_set_false:N \l__pic_region_fill_bool	1055
1056	\prg_map_break:Nn \l__ztool_pic_rec_fill {}	1056
1057	}	1057
1058	%% begin fill rounded rectangle	1058
1059	__@@_pic_put:nnn {__coor_st:n {#2}}{__coor_nd:n {#2}}	1059
1060	{	1060
1061	__pic_set_fill_color:	1061
1062	\rule	1062
1063	{\fp_eval:n {\rec@width *\dim_to_decimal:n {\l__pic_unit_dim}}pt}	1063
1064	{\fp_eval:n {\rec@height*\dim_to_decimal:n {\l__pic_unit_dim}}pt}	1064
1065	}	1065
1066	\int_set:Nn \l__pic_rec_quadrant_index_int { 0 }	1066
1067	\tl_map_inline:nn	1067
1068	{	1068
1069	{__coor_st:n {#2}+\rec@width-\rec@arc, __coor_nd:n	1069
	{#2}+\rec@height-\rec@arc}	
1070	{__coor_st:n {#2}+\rec@arc, __coor_nd:n	1070
	{#2}+\rec@height-\rec@arc}	
1071	{__coor_st:n {#2}+\rec@arc, __coor_nd:n {#2}+\rec@arc}	1071
1072	{__coor_st:n {#2}+\rec@width-\rec@arc, __coor_nd:n {#2}+\rec@arc}	1072
1073	}{	1073
1074	\int_incr:N \l__pic_rec_quadrant_index_int	1074
1075	\edef\qu@drant@index{\int_use:N \l__pic_rec_quadrant_index_int}	1075
1076	\exp_last_unbraced:Ne __@@_pic_put:nnn {__coor_st_nd:n {##1}}	1076
1077	{	1077
1078	__color_safe_use:V \l__pic_opacity_color_tl	1078
1079	__@@_pic_arc:nnnn {*}	1079
1080	{ (\qu@drant@index-1)*90 }	1080
1081	{ \qu@drant@index*90 }	1081
1082	{ sqrt(2)*\rec@arc }	1082
1083	__pic_set_fill_color:	1083
1084	__@@_pic_arc:nnnn {*}{0}{360}{\rec@arc}	1084
1085	}	1085
1086	}	1086

1087	<code>%% end fill rounded rectangle</code>	1087
1088	<code>\prg_break_point:Nn \l__ztool_pic_rec_fill { }</code>	1088
1089	<code>__@@_pic_put:nnn {__coord_st:n {#2}+\rec@width/2}{__coord:n {#2}+\rec@height/2}</code>	1089 ✓
1090	<code>{</code>	1090
1091	<code> __pic_set_line_color:</code>	1091
1092	<code> __pic_set_line_width:</code>	1092
1093	<code> __@@_pic_oval:nnnn</code>	1093
1094	<code> { \rec@arc }{ }</code>	1094
1095	<code> { \rec@width }</code>	1095
1096	<code> { \rec@height }</code>	1096
1097	<code>}</code>	1097
1098	<code>\group_end:</code>	1098
1099	<code>}</code>	1099
1100	<code>\NewDocumentCommand{\zrectangle}{0{}d()d()}</code>	1100
1101	<code>{</code>	1101
1102	<code> \ztool_pic_rectangle:nnn { #1 }{#2}{#3}</code>	1102
1103	<code>}</code>	1103

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.

Symbols		
-shell-escape	3–6	ztool/./zrectangle/fill 14
B		ztool/draw/picture/height 13
\begin	13	ztool/draw/picture/opacity-color 13
bool commands:		ztool/draw/picture/unit 13
\c_false_bool	6	ztool/draw/picture/width 13
\c_true_bool	6, 21	ztool/draw/picture/xoffset 13
		ztool/draw/picture/yoffset 13
		\tikz 32
C		V
coffin commands:		\Vector 33
\coffin_rotate:Nn	11	
\coffin_scale:Nnn	11	X
		xsim commands:
E		\xsim_file_write_start:nn 16
\end	13	\xsim_file_write_stop: 16
L		Z
\Line	33	\zarc 14
\ltxarrows	14	\zcircle 14
O		\zdraw 14, 32
\oval	36	\zline 13
P		zpic 13
\pdfsetmatrix	11	\zrac 14
\polygon	33	\zrectangle 14
\polyline	33	ztool commands:
\polyvector	33	\ztool_affine_transformation:Nnnnn ... 10, 11
\pstarrows	14	\ztool_append_to_file:nn 6
\put	13	\ztool_autoset_to_wd_and_ht:nn 9
S		\ztool_autoset_to_wd_and_ht:nnn 9
\show	27	\ztool_box_item_align:Nnnn 9
T		\ztool_file_new:nn 6
ztool/./line/dash	13	\ztool_fp_to_rad:n 9
ztool/./line/draw	13	\ztool_get_dp:Nn 8
ztool/./line/width	13	\ztool_get_ht:Nn 8
ztool/./vector/>	14	\ztool_get_ht_plus_dp:Nn 8
ztool/./zarc/end	14	\ztool_get_shell_pwd:N 4
ztool/./zarc/fill	14	\ztool_get_wd:Nn 8
ztool/./zarc/radius	14	\ztool_gget_dp:Nn 8
ztool/./zarc/start	14	\ztool_gget_dp:nn 8
ztool/./zdraw/cycle	14	\ztool_gget_ht:Nn 8
ztool/./zdraw/fill	14	\ztool_gget_wd:Nn 8
ztool/./zdraw/shift	14	\ztool_gread_file_as_seq:nnN 6
ztool/./zdraw/vector	14	\ztool_insert_to_file:nnn 7
ztool/./zrectangle/arc	14	\ztool_read_file_as_seq:nnN 6
		\ztool_replace_file_line:nnn 6
		\ztool_rotate:nn 9
		\ztool_scale_to_ht:nn 9

\ztool_scale_to_wd:nn	9	\ztool_shell_mv:nn	4
\ztool_scale_to_wd_and_ht:nnn	9	\ztool_shell_rm:n	4
\ztool_set_to_ht:nn	8	\ztool_shell_rmdir:n	4
\ztool_set_to_wd:nn	8	\ztool_shell_split_ls:nN	5
\ztool_shell_cp:nn	4	ztoolboxaffine	10
\ztool_shell_escape:n	4	\zvector	13
\ztool_shell_mkdir:n	4		