

physicx, 扩展的 physics 宏包

雾月, Longaster*

2022 年 9 月 13 日 v0.3.4

目录

第 1 节 简介	1	4.4 genegralmatrix	11
第 2 节 基本用法与宏包选项	1	4.5 matrix 类通用键值选项	11
第 3 节 括号, quantity	2	4.6 例子	14
3.1 新的括号命令	2	第 5 节 杂项	17
3.2 旧的括号命令	3	5.1 定义新的括号命令	18
第 4 节 矩阵, matrix	5	5.2 定义新的矩阵命令	19
4.1 diagonalmatrix	6	版本历史	21
4.2 commatrix	8	代码索引	21
4.3 qxmatrix	9		

第 1 节 简介

physics 宏包定义了一些简写的命令, 但是已经 9 年未更新了, physicx 对其进行了一定的扩展, 基本兼容 physics 原有的命令。

要求: expl3 宏包版本至少 Released 2021-08-27, 不准备兼容低版本, 尽管并不依赖某些新特性。如果使用 TeXLive, 可以使用自带的控制台进行更新。如果你想兼容低版本, 可以在 LPPL 1.3c 协议下自行修改。

第 2 节 基本用法与宏包选项

太长不看版: 若仍要使用 physics 宏包, 则使用 physics 宏包选项, 不可再加载 physics 宏包。

通常情况下, 可用像使用 physics 宏包那样使用 physicx 的命令, 它们基本兼容的, 但是 physicx 宏包并不依赖 physics 宏包, 但目前 physicx 并不提供诸如 \dd 的命令, 你可以使用 physics 宏包选项来加载 physics 宏包。也可以自行加载, 但必须首先加载 physics 再加载 physicx 宏包, 一般情况下使用 physics 宏包选项即可。

宏包选项: compat 将尽量兼容 physics 宏包。physics: 自动启用 compat 选项并加载 physics 宏包。short: 将重定义一些简写的命令。mathtools: 启用 mathtools 宏包定义的一些环境, 如 matrix* 等。unimath: 加载 unicode-math 宏包并设置一些兼容操作。

reqty 控制是否重定义 physics 宏包的括号类命令。noqty 相当于 reqty=false。初始为重定义。

*Email: longaster@163.com

`fixdif` 用于控制是否接入 `fixdif` 宏包的功能, 当这个选项设置为真时, 用户必须自行加载 `fixdif` 宏包, 若还启用了 `physics` 选项, 则将微分号和偏微分号替换为 `fixdif` 的实现。

`original` 用于恢复被 `physics` 宏包重定义的 `\div`、`\Re`、`\Im` 命令。仅当使用了 `physics` 选项或加载了 `physics` 宏包时生效。

第3节 括号, *quantity*

3.1 新的括号命令

“新的括号命令”指 `physicx` 宏包定义的括号命令。

<code>\xquantity</code>	<code>\xquantity {⟨quantity类键值选项⟩} {⟨code⟩}</code>
<code>\qxqty</code>	<code>\qxqty [⟨quantity类键值选项⟩] {⟨code⟩}</code>
	<code>\qxqty * [⟨quantity类键值选项⟩] {⟨code⟩}</code>

New: 2021-11-13

这些命令用于替代 `physics` 宏包的括号类命令。

带 `*` 的命令禁止自动调整括号大小。

`\xquantity` 的速度更快些。

可以自定义 `quantity` 类命令, 见第 5.1 节。

例 1:

```
$ \xquantity{x^2} \quad \xquantity{type=v}{\int} \quad \displaystyle
\xquantity{type=v}{\int_a^b} $ \quad
$ \xquantity{size=\bigg}{x^2} \quad \displaystyle
\xquantity{size=\bigg}{\int_a^b} $
\[
\xquantity{x^2} \quad \quad \xquantity{type=v}{\int} \quad
\xquantity{type=v}{\int_a^b} \quad \quad \xquantity{size=\bigg}{x^2}
\xquantity{size=\bigg}{\int_a^b}
\]

$ \qxqty{x^2} \quad \quad \qxqty{type=v}{\int} \quad \displaystyle
\xqty[v]{\int_a^b} $ \quad
$ \qxqty[size=\bigg]{x^2} \quad \displaystyle
\xqty[size=\bigg]{\int_a^b} $
\[
\xqty{x^2} \quad \quad \qxqty{type=v}{\int} \quad
\xqty[v]{\int_a^b} \quad \quad \qxqty[size=\bigg]{x^2} \quad
\xqty[size=\bigg]{\int_a^b}
\]
```

$$(x^2) \quad |f| \quad \left| \int_a^b \right| \quad (x^2) \quad \left(\int_a^b \right)$$

$$(x^2) \quad |f| \quad \left| \int_a^b \right| \quad (x^2) \quad \left(\int_a^b \right)$$

$$(x^2) \quad |f| \quad \left| \int_a^b \right| \quad (x^2) \quad \left(\int_a^b \right)$$

$$(x^2) \quad |f| \quad \left| \int_a^b \right| \quad (x^2) \quad \left(\int_a^b \right)$$

physicx 宏包的一个标准的括号命令由如下 8 个部分构成:

$\langle pre \rangle \langle left-size \rangle \langle left \rangle \langle arg \rangle \langle code \rangle \langle right-size \rangle \langle right \rangle \langle post \rangle$

每一部分都有一个对应的键,可以在 $\langle quantity$ 类键值选项 \rangle 中设置。

pre	它们代表了标准括号命令的对应部分。当 <code>left-size</code> 、 <code>right-size</code> 有一个为空时,则会移除
post	<code>left-size</code> 、 <code>right-size</code> 这两个部分。

left	<code>left-size</code> 初始为 <code>\left</code> , <code>right-size</code> 初始为 <code>\right</code> 。
------	---

right	
-------	--

left-size	
-----------	--

right-size	
------------	--

code	
------	--

size	<code>size</code> 同时设置 <code>left-size</code> 和 <code>right-size</code> 。 <code>noauto</code> 将 <code>size</code> 设置为空。一般情况下,可以不写
noauto	<code>size=</code> , 而只写它的值。如 <code>\qxqty[size=\Big]{x}</code> 与 <code>\qxqty[\Big]{x}</code> 效果相同。

args	<code>args</code> 键将 $\langle arg \rangle$ 部分设置为 $[\langle args \rangle]$, 而 <code>args*</code> 键将 $\langle arg \rangle$ 部分设置为其值。
------	--

args*	
-------	--

type	<code>type = $\langle p b B v V a m pm bm Bm vm Vm sm \dots \rangle$</code> 初始值 = p
------	--

`type` 键使用预定义的括号类型。

它们的值正如 **amsmath** 的矩阵环境的 `matrix` 前的部分。 p 代表 $()$, b 代表 $[]$, B 代表 $\{ \}$, v 代表 $||$, V 代表 $|||$, a 代表 $\langle \rangle$ 。

带有 m 的则为矩阵类的括号。

如果使用了 **mathtools** 宏包或使用了 **mathtools** 宏包选项,则还定义了 `p*`、`sm*`、`spm`、`spm*` 等值,它们对应 `pmatrix*`、`smallmatrix*`、`psmallmatrix`、`psmallmatrix*` 等环境。可以使用 `args` 键来设置它们的参数。**mathtools** 宏包必须手动加载。

它们是通过 `left`、`right`、`size` 键来设置的。

注意: `left`、`right` 的设置必须合法,即若 `size` 不为空,则 `left`、`right` 必须可以跟在 `\left`、`\right` 这两个调整大小的命令后面。不能如同:

`left=\begin{matrix}`, `right=\end{matrix}`, `left-size=\left`, `right-size=\right`。

一般情况下,可以不写 `type=` 而只写它的值。

例 2:

```
\[
  \xquantity{left=\begin{pmatrix}, right=\end{pmatrix}, noauto}
    { ABC & dec \\ hij & HJK } \quad
  \xquantity{left=\begin{smallmatrix}, right=\end{smallmatrix}, noauto}
    { ABC & dec \\ hij & HJK } \quad
  \xquantity{type=m*} { ABC & dec \\ hij & HJK } \quad
  \xquantity{type=sm} { ABC & dec \\ hij & HJK }
\]
```

$$\left(\begin{matrix} ABC & dec \\ hij & HJK \end{matrix} \right) \quad \begin{smallmatrix} ABC & dec \\ hij & HJK \end{smallmatrix} \quad \begin{pmatrix} ABC & dec \\ hij & HJK \end{pmatrix} \quad \begin{smallmatrix} ABC & dec \\ hij & HJK \end{smallmatrix}$$

3.2 旧的括号命令

“旧的括号命令”是指 **physics** 宏包定义的命令。可以使用 `reqty` 宏包选项设置是否使用新的机制重新定义它们。

在仅使用 **physicx** 而不使用 **physics** 宏包时,本节命令不宜再使用,仅出于兼容性原因而保留。

增加了 `\oorder`、`\OOrder` 命令, 以输出 $o(\blacksquare)$ 、 $\mathcal{O}(\blacksquare)$, 它们的大小写总是固定的。而 `\order` 则会依据 `physics` 宏包是否加载、`compat` 选项是否给定来使用大小写, 若给定, 出于兼容性考虑, `\order` 输出大写的 $\mathcal{O}(\blacksquare)$ 。

Full	Short	Description
<code>\quantity</code>	$\backslash\mathrm{qty}(\backslash\mathrm{typical})_{\square} \rightarrow (\blacksquare)$ $\backslash\mathrm{qty}(\backslash\mathrm{tall})_{\square} \rightarrow (\blacksquare)$ $\backslash\mathrm{qty}(\backslash\mathrm{grande})_{\square} \rightarrow (\blacksquare)$ $\backslash\mathrm{qty}[\backslash\mathrm{typical}]_{\square} \rightarrow [\blacksquare]$ $\backslash\mathrm{qty} \backslash\mathrm{typical} _{\square} \rightarrow \blacksquare $ $\backslash\mathrm{qty}\{\backslash\mathrm{typical}\}_{\square} \rightarrow \{\blacksquare\}$ $\backslash\mathrm{qty}\langle\backslash\mathrm{typical}\rangle_{\square} \rightarrow \langle\blacksquare\rangle$ $\backslash\mathrm{qty}=\backslash\mathrm{typical}=_{\square} \rightarrow \ \blacksquare\ $ $\backslash\mathrm{qty}\backslash\mathrm{big}\{\}_{\square} \rightarrow \{\}$ $\backslash\mathrm{qty}\backslash\mathrm{Big}\{\}_{\square} \rightarrow \{\}$ $\backslash\mathrm{qty}\backslash\mathrm{bigg}\{\}_{\square} \rightarrow \{\}$ $\backslash\mathrm{qty}\backslash\mathrm{Bigg}\{\}_{\square} \rightarrow \{\}$ $\backslash\mathrm{pqty}\{\} \leftrightarrow \backslash\mathrm{qty}()$ $\backslash\mathrm{bqty}\{\} \leftrightarrow \backslash\mathrm{qty}[]$ $\backslash\mathrm{vqty}\{\} \leftrightarrow \backslash\mathrm{qty} $ $\backslash\mathrm{Bqty}\{\} \leftrightarrow \backslash\mathrm{qty}\{\}$	automatic () braces automatic [] braces automatic braces automatic { } braces automatic < > braces automatic braces manual sizing (works with any of the above bracket types)
<code>\absolutevalue</code>	$\backslash\mathrm{abs}\{a\} \rightarrow a $ $\backslash\mathrm{abs}\backslash\mathrm{Big}\{a\} \rightarrow a $ $\backslash\mathrm{abs}*\backslash\mathrm{grande} \rightarrow \blacksquare $	automatic sizing; equivalent to $\backslash\mathrm{qty} a $ inherits manual sizing syntax from $\backslash\mathrm{qty}$ star for no resize
<code>\norm</code>	$\backslash\mathrm{norm}\{a\} \rightarrow \ a\ $ $\backslash\mathrm{norm}\backslash\mathrm{Big}\{a\} \rightarrow \ a\ $ $\backslash\mathrm{norm}*\backslash\mathrm{grande} \rightarrow \ \blacksquare\ $	automatic sizing manual sizing star for no resize
<code>\evaluated</code>	$\backslash\mathrm{eval}\{x\}_0^{\infty} \rightarrow x _0^{\infty}$ $\backslash\mathrm{eval}\backslash\mathrm{Big}\{x\}_0^{\infty} \rightarrow x _0^{\infty}$ $\backslash\mathrm{eval}(x _0^{\infty} \rightarrow (x _0^{\infty}$ $\backslash\mathrm{eval}[x _0^{\infty} \rightarrow [x _0^{\infty}$ $\backslash\mathrm{eval}[\backslash\mathrm{venti} _0^{\infty} \rightarrow [\blacksquare _0^{\infty}$ $\backslash\mathrm{eval}*\backslash\mathrm{venti} _0^{\infty} \rightarrow [\blacksquare _0^{\infty}$	vertical bar for evaluation limits alternate form alternate form automatic sizing star for no resize
<code>\order</code>	$\backslash\mathrm{order}\{x^2\} \rightarrow \mathcal{O}(x^2)$ $\backslash\mathrm{order}\backslash\mathrm{Big}\{x^2\} \rightarrow \mathcal{O}(x^2)$ $\backslash\mathrm{oorder}\{x^2\} \rightarrow o(x^2)$ $\backslash\mathrm{OOrder}\{x^2\} \rightarrow \mathcal{O}(x^2)$	order symbol; automatic sizing and space handling manual sizing lowercase o uppercase O

	<code>\order*{\grande} \rightarrow \mathcal{O}(\text{Box})</code>	star for no resize
<code>\commutator</code>	<code>\comm{A}{B} \rightarrow [A, B]</code>	automatic sizing
	<code>\comm\Big{A}{B} \rightarrow \Big[A, B\Big]</code>	manual sizing
	<code>\comm*{A}{\grande} \rightarrow [A, \text{Box}]</code>	star for no resize
<code>\anticommutator</code>	<code>\acomm{A}{B} \rightarrow \{A, B\}</code>	same as <code>\poissonbracket</code>
<code>\poissonbracket</code>	<code>\pb{A}{B} \rightarrow \{A, B\}</code>	same as <code>\anticommutator</code>

出于兼容性考量, 仍然保留了 `\matrixquantity` (`\mqty`)、`\smallmatrixquantity` (`\smqty`) 命令的定义, 它们与 `\quantity` 命令相似, 只是将它们的值放入相应的矩阵中。可以使用更高级的矩阵命令, 见第 4 节。

你还可以自定义 `quantity` 类命令, 见第 5 节。

第 4 节 矩阵, *matrix*

`physicx` 宏包极大地扩展了 `physics` 原有的矩阵类命令。提供了丰富的键值接口来设置矩阵。

定义了三个基本命令: `\genegralmatrix`、`\commamatrix`、`\diagonalmatrix`, 及 `\qxmatrix`。

例 3: `\diagonalmatrix`

```

\diagonalmatrix{1,2,3,4}$,
\diagonalmatrix[p]{1,2,3,4}$,
\diagonalmatrix[p]{0={1,2,3,4}, 2={22,33}, '-2={-22,-33}}$,
\diagonalmatrix[p, empty=\Box]{0={1,2,3,4}, 2={22,33}, '-2={-22,-33}}$,

\diagonalmatrix[p, empty=\Box,
  item={{4}{4}=\blacksquare, {{2}{3}=\oplus}},
] {0={1,2,3,4}, 2={22,33}, '-2={-22,-33}}$,
\diagonalmatrix[transpose, p, empty=\Box]{0={1,2,3,4}, 2={22,33}, '-2={-22,-33}}$,
\diagonalmatrix[transpose, p, empty=\Box,
  item={{4}{4}=\blacksquare, {{2}{3}=\oplus}},
] {0={1,2,3,4}, 2={22,33}, '-2={-22,-33}}$,

```

$$\begin{array}{cccc}
 1 & & & \\
 & 2 & & \\
 & & 3 & \\
 & & & 4
 \end{array}
 \begin{pmatrix} 1 & & & \\ & 2 & & \\ & & 3 & \\ & & & 4 \end{pmatrix},
 \begin{pmatrix} 1 & & 22 & \\ & 2 & & 33 \\ & & 3 & -22 \\ & & -33 & 4 \end{pmatrix},
 \begin{pmatrix} 1 & \Box & 22 & \Box \\ \Box & 2 & \Box & 33 \\ \Box & \Box & 3 & -22 \\ \Box & \Box & -33 & 4 \end{pmatrix},$$

$$\begin{pmatrix} 1 & \Box & 22 & \Box \\ \Box & 2 & \oplus & 33 \\ \Box & \Box & 3 & -22 \\ \Box & \Box & -33 & \blacksquare \end{pmatrix},
 \begin{pmatrix} 1 & \Box & \Box & \Box \\ \Box & 2 & \Box & \Box \\ 22 & \Box & 3 & -33 \\ \Box & 33 & -22 & 4 \end{pmatrix},
 \begin{pmatrix} 1 & \Box & \Box & \Box \\ \Box & 2 & \Box & \Box \\ 22 & \oplus & 3 & -33 \\ \Box & 33 & -22 & \blacksquare \end{pmatrix},$$

例 4: \commamatrix

```
\commamatrix{1, 2, 3, 4}$, \quad
\commamatrix[b]{1, 2, 3, 4}$, \quad
\commamatrix{1; 2; 3; 4}$, \quad
\commamatrix[b]{1; 2; 3; 4}$, \quad
\commamatrix[b]{1, 2; 3, 4}$, \quad
\commamatrix[b]{1 & 2 \\\ 3, 4; 5, 6}$, \quad
```

$$1 \quad 2 \quad 3 \quad 4, \quad \begin{bmatrix} 1 & 2 & 3 & 4 \end{bmatrix}, \quad \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix}, \quad \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix},$$

例 5: \generalmatrix

```
\generalmatrix{p}{A}{m}{n}$,
\generalmatrix{v}{A}{3}{3}$,
\generalmatrix*{p}{A}{3}{3}$,
```

$$\begin{pmatrix} A_{11} & A_{12} & A_{13} & \cdots & A_{1n} \\ A_{21} & A_{22} & A_{23} & \cdots & A_{2n} \\ A_{31} & A_{32} & A_{33} & \cdots & A_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & A_{m3} & \cdots & A_{mn} \end{pmatrix}, \quad \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}, \quad \begin{pmatrix} A_{11} & A_{12} & A_{13} & \cdots \\ A_{21} & A_{22} & A_{23} & \cdots \\ A_{31} & A_{32} & A_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix},$$

4.1 diagonalmatrix

diagonalmatrix, 即对角矩阵。

```
\diagonalmatrix \diagonalmatrix [\langle matrix 类键值选项 \rangle] \{\langle diag 键选项 \rangle\}
\diagonalmatrix = [\langle matrix 类键值选项 \rangle] \{\langle diag 键选项 \rangle\}
\diagonalmatrix + [\langle matrix 类键值选项 \rangle] \{\langle diag 键选项 \rangle\}
\diagonalmatrix = + [\langle matrix 类键值选项 \rangle] \{\langle diag 键选项 \rangle\}
```

= 开启保存模式, 即不输出计算的矩阵, 而将其全局保存至 \physicxtmp 宏中, 将其展开一次即可得到矩阵的值 (不包括外部的 \begin{matrix} 等)。

= 将覆盖 saveto 和 saveto* 键的设置, 即总是将矩阵保存到 \physicxtmp 中, 而不管 saveto 和 saveto* 的设置。

+ 启用 enhanced 模式, 在该模式下能使用更多的键值选项, 但速度相较而言慢些。

diag 键选项:

```
auto-update auto-update 将自动更新矩阵的行数和列数, 使得给出的元素总是完整地出现在矩阵中。
noauto-update true 在设置主对角线元素的同时设置矩阵的行数和列数为主对角线元素个数。行数和列
true 数可以被后续的处理更改。
false 数字键将设置矩阵的对角元素的值。对角元素的位置是当前矩阵的相应元素的位置。
0 不带 “'” 的设置主对角线元素, 即 “\”, 带 “'” 设置副对角线元素, 即 “/”。正数为上方,
1 负数为下方。
-1 注意副对角线元素的设置依赖矩阵的大小, 即矩阵的行数和列数, 在设置副对角线元素时
'1 行数和列数必须被直接或间接地设置。
'-1 当 \diag 键选项 不为键值对时, 则将其设置为主对角线元素。
... 这个矩阵命令自动更新 MaxMatrixCols 计数器, 即矩阵最大的列数。
```

见下方的几个例子。

例 6:

```
\diagonalmatrix[v,empty=\Box]{a,b,c,d}$ \
\diagonalmatrix[v,empty=\Box]{ 0={a,b,c,d} }$ \
\diagonalmatrix[v,empty=\Box]{auto-update, 2={a,b,c,d} }$ \
\diagonalmatrix[v,empty=\Box]{ 2={a,b,c,d} }$ \
\diagonalmatrix[v,empty=\Box]{auto-update, '0={a,b,c,d} }$ \
\diagonalmatrix[v,empty=\Box]{ '0={a,b,c,d} }$
```

$$\begin{vmatrix} a & & & \\ & b & & \\ & & c & \\ & & & d \end{vmatrix} \begin{vmatrix} a & & & \\ & b & & \\ & & c & \\ & & & d \end{vmatrix} \begin{vmatrix} & & a & & \\ & & & b & \\ & & & & c \\ & & & & & d \end{vmatrix} \begin{vmatrix} & & & a \\ & & & b \\ & & & c \\ d & & & \end{vmatrix}$$

例 7:

```
\diagonalmatrix[v,empty=\Box]{ 0={a,b,c,d}, 1={h,j,j} }$ \
\diagonalmatrix[v,empty=\Box]{ 0={a,b,c,d}, 1={h,j,j,k} }$ \
\diagonalmatrix[v,empty=\Box]{ auto-update, 0={a,b,c,d}, 1={h,j,j,k} }$ \
\diagonalmatrix[v,empty=\Box]{ auto-update, 0={a,b,c,d}, 1={h,j,j,k} , -4={m} }$ \
```

$$\begin{vmatrix} a & h & & \\ & b & j & \\ & & c & j \\ & & & d \end{vmatrix} \begin{vmatrix} a & h & & \\ & b & j & \\ & & c & j \\ & & & d \end{vmatrix} \begin{vmatrix} a & h & & \\ & b & j & \\ & & c & j \\ & & & d & k \end{vmatrix} \begin{vmatrix} a & h & & \\ & b & j & \\ & & c & j \\ & & & d & k \\ m & & & & \end{vmatrix}$$

例 8:

```
\diagonalmatrix[v,empty=\Box]{ 0={1,2,3,4}, -1={11,22,33}, '-1={p,pp,ppp} }$ \
\diagonalmatrix[v,empty=\Box]{ 0={1,2,3,4}, 2=A, -1={\sum,\cup,\prod}, '-1={\int,\oint_a^n} }$ \
\diagonalmatrix[v,empty=\Box,sep=\C]{ 0=1 \C 2 \C 3 \C 4, 2=A, -1={\sum \C \cup \prod, '-1={\int \C \oint_a^n} }$ \
\diagonalmatrix[v,empty=\Box,sep=\C]{ 0=1 \C 2 \C 3 \C 4, 2=A, -1={\sum \C \cup \prod \C \int \C \oint_a^n} }$ \
{\ttfamily\meaning\physicstmp}
```

$$\begin{vmatrix} 1 & & & \\ 11 & 2 & & p \\ & 22 & pp & \\ & & ppp & 33 & 4 \end{vmatrix} \begin{vmatrix} 1 & & A & \\ \sum & 2 & & f \\ & \cup & \oint_a^n & \\ & & \prod & 4 \end{vmatrix} \begin{vmatrix} 1 & & A & \\ \sum & 2 & & f \\ & \cup & \oint_a^n & \\ & & \prod & 4 \end{vmatrix}$$

macro:->1&\Box &A&\Box \\\sum &2&\Box &\int \\\Box &\cup &\oint_a^n&\Box \\\Box &\Box &\prod &4

`\diagonalmatrix` 中可用的 `matrix` 类键值选项:

见表 2, 它们的具体用法见第 4.5 节。

`<diag 键选项>` 设置的即是 `diag` 键。

`enhanced` 模式下增加的可用键值:

表 2

expand	rows	cols	auto-update	empty
check	diag	diag+	diag-now	diag-data
item	item+	item-now	item-data	check-range
begin	end	args	args*	after-begin
after-begin+	after-end	after-end+	sepdim	type
saveto	saveto*	transpose	'	T
MaxMatrixCols	enhanced	!enhanced	sep	adi-order
beginning	beginning+	ending	ending+	

row-list	col-list	element-code	element-code*
element-except	element-except+	expand-element	row-iterate
col-iterate			

4.2 commamatrix

commamatrix, 即逗号分隔的矩阵。

```

\commamatrix <matrix类键值选项> {<矩阵元素>}
\commamatrix = <matrix类键值选项> {<矩阵元素>}
\commamatrix + <matrix类键值选项> {<矩阵元素>}
\commamatrix = + <matrix类键值选项> {<矩阵元素>}

```

=, + 的功能同 \diagonalmatrix。

在非 enhanced 模式下, 做的工作仅仅是把 align 分隔符替换为 &, 把 cr 分隔符替换为 \\, 功能有限, 但速度更快。

这个矩阵命令 **不会** 自动更新 MaxMatrixCols 计数器, 需要通过 MaxMatrixCols 键设置, 或通过 \setcounter 设置。

见下方的几个例子。

例 9:

```

$\commamatrix[b]{a,b;c,d}$ \
$\commamatrix[b]{ {a,b} ; c,d}$ \
$\commamatrix[b]{a & b \\ c & d}$ \
$\commamatrix[b]{a, b & c \\ d & e, f}$ \
$\commamatrix[b, align=\T, cr=\N]{a \T b \T c \N d \T e \T f}$ \
$\commamatrix[b, align=\T, cr=\N]{a,a \T b;b \T c \N d \T e \T f}$ \

$\commamatrix=[b, align=\T, cr=\N]{a \T b \T c \N d \T e \T f}$ \
{\ttfamily\meaning\physicxtmp}

```

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}
 \begin{bmatrix} a, b \\ c & d \end{bmatrix}
 \begin{bmatrix} a & b \\ c & d \end{bmatrix}
 \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}
 \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}
 \begin{bmatrix} a, a & b; b & c \\ d & e & f \end{bmatrix}$$

macro:->a &b &c \\d &e &f

\commamatrix 可用的键值选项:

见表 3, 具体用法见第 4.5 节。

<矩阵元素> 设置的即是 array 键。

enhanced 模式下增加的键值选项:

表 3

array	expand	rows	cols	check-range
begin	end	args	args*	after-begin
after-end	sepdim	type	save-to	save-to*
MaxMatrixCols	enhanced	!enhanced	cr	align

row-list	col-list	element-code	element-code*
element-except	element-except+	expand-element	row-iterate
col-iterate			

4.3 qxmatrix

\qxmatrix 扩展的 \xmatrix。

```
\qxmatrix <[<matrix类键值选项>] <{<main>} <{<last row>} <[<row>] <{<last col>} <[<col>]>
\xqmatrix * <[<matrix类键值选项>] <{<main>} <{<last row>} <[<row>] <{<last col>} <[<col>]>
\xqmatrix = <[<matrix类键值选项>] <{<main>} <{<last row>} <[<row>] <{<last col>} <[<col>]>
\xqmatrix = * <[<matrix类键值选项>] <{<main>} <{<last row>} <[<row>] <{<last col>} <[<col>]>
```

= 的作用同前所述。

* 输出无穷维矩阵,即在最后的行和列添加 $\cdots; \ddots$ 。

见如下几个例子。

例 10:

```
$\qxmatrix{A}{3}{3}$ \
$\qxmatrix{A}{7}{3}{9}{3}$ \
$\qxmatrix{A}{m}{2}{n}{2}$ \

$\qxmatrix[m]{A}{m}{2}{n}{2}$ \
$\qxmatrix*{A}{m}{2}{n}{2}$ \
$\qxmatrix[
  element-code=\displaystyle\pdv{\#1_{\#2}}{x_{\#3}},
  sepdim=2ex,
  p,
]{A}{m}{2}{n}{2}$ \
```

$$\begin{pmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix} \begin{pmatrix} A_{11} & A_{12} & A_{13} & \cdots & A_{19} \\ A_{21} & A_{22} & A_{23} & \cdots & A_{29} \\ A_{31} & A_{32} & A_{33} & \cdots & A_{39} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_{71} & A_{72} & A_{73} & \cdots & A_{79} \end{pmatrix} \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{pmatrix}$$

$$\begin{matrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{matrix} \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1n} & \cdots \\ A_{21} & A_{22} & \cdots & A_{2n} & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} & \cdots \\ \vdots & \vdots & & \vdots & \ddots \end{pmatrix} \begin{pmatrix} \frac{\partial A_1}{\partial x_1} & \frac{\partial A_1}{\partial x_2} & \cdots & \frac{\partial A_1}{\partial x_n} \\ \frac{\partial A_2}{\partial x_1} & \frac{\partial A_2}{\partial x_2} & \cdots & \frac{\partial A_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial A_m}{\partial x_1} & \frac{\partial A_m}{\partial x_2} & \cdots & \frac{\partial A_m}{\partial x_n} \end{pmatrix}$$

例 11:

```
\newcommand{\twotwomat}[2][\]{\qmatrix[type=#2,#1]{A}{n}[2]{n}[2]}
\setmatrixtype*{jacobian}{
  element-code=\displaystyle\pdv{##1_{{##2}}}{x_{{##3}}},
  sepdim=2ex}
$\twotwomat[p]{jacobian}$

\setmatrixtype*{parenmat}{ element-code=##1_{{##2}}(x_{{##3}}) }
$\twotwomat{parenmat}$

\setmatrixtype*{plusmain}{ element-code={
  \ifthenelse{\equal{##2}{##3}}{1+}{##1_{{##2}}(x_{{##3}})
} }
$\twotwomat[v]{plusmain}$
```

$$\left(\begin{array}{cccc} \frac{\partial A_1}{\partial x_1} & \frac{\partial A_1}{\partial x_2} & \cdots & \frac{\partial A_1}{\partial x_n} \\ \frac{\partial A_2}{\partial x_1} & \frac{\partial A_2}{\partial x_2} & \cdots & \frac{\partial A_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial A_n}{\partial x_1} & \frac{\partial A_n}{\partial x_2} & \cdots & \frac{\partial A_n}{\partial x_n} \end{array} \right) \begin{array}{cccc} A_1(x_1) & A_1(x_2) & \cdots & A_1(x_n) \\ A_2(x_1) & A_2(x_2) & \cdots & A_2(x_n) \\ \vdots & \vdots & \ddots & \vdots \\ A_n(x_1) & A_n(x_2) & \cdots & A_n(x_n) \end{array}$$

$$\left| \begin{array}{cccc} 1 + A_1(x_1) & A_1(x_2) & \cdots & A_1(x_n) \\ A_2(x_1) & 1 + A_2(x_2) & \cdots & A_2(x_n) \\ \vdots & \vdots & \ddots & \vdots \\ A_n(x_1) & A_n(x_2) & \cdots & 1 + A_n(x_n) \end{array} \right|$$

一个稍复杂的例子:

例 12:

```
% \usepackage{ifthen}
\def\hessianaux#1{
  \def\twopartital##1##2##3{%
    \displaystyle\ifthenelse{\equal{##2}{##3}}
      {\pdv[2]{##1_{{##2}}}{x_{{##3}}}}
      {\pdv{##1_{{##2}}}{x_{{##3}}}}}
\newcommand{\hessian}[4][\]{%
  \hessianaux{#3}%
  \qmatrix[element-code=twopartital,#1]{#2}{#4}[1]{#4}[1]}
$\hessian[v,sepdim=1ex]{f}{x}{m}$
$\hessian[b,sepdim=1ex]{\mathcal{H}}{p}{n}$
```

$$\left[\begin{array}{ccc} \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_m \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_m^2} \end{array} \right] \left[\begin{array}{ccc} \frac{\partial^2 \mathcal{H}}{\partial p_1^2} & \cdots & \frac{\partial^2 \mathcal{H}}{\partial p_1 \partial p_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 \mathcal{H}}{\partial p_n \partial p_1} & \cdots & \frac{\partial^2 \mathcal{H}}{\partial p_n^2} \end{array} \right]$$

`\qmatrix` 可用的键值选项:

见表 4, 具体用法见第 4.5 节。

表 4

array	expand	rows	cols
auto-update	main	row-list	col-list
infinite	!infinite	element-code	element-code*
element-except	element-except+	expand-element	row-iterate
col-iterate	begin	end	args
args*	sepdim	type	saveto
saveto*	MaxMatrixCols		

4.4 genegralmatrix

generalmatrix, 即通用矩阵命令。

```

\generalmatrix \generalmatrix + {<matrix类键值选项>}
\generalmatrix = + {<matrix类键值选项>}
\generalmatrix + * {<matrix类键值选项>}
\generalmatrix = + * {<matrix类键值选项>}
\generalmatrix {<matrix类键值选项>} {<main>} {<last row>} [<row>] {<last col>} [<col>]
\generalmatrix = {<matrix类键值选项>} {<main>} {<last row>} [<row>] {<last col>} [<col>]
\generalmatrix * {<matrix类键值选项>} {<main>} {<last row>} [<row>] {<last col>} [<col>]
\generalmatrix = * {<matrix类键值选项>} {<main>} {<last row>} [<row>] {<last col>} [<col>]

```

不带 + 的命令就是 \qxmatrix 命令, 不过其第一个参数是必须用 { } 给定的。

这个矩阵命令自动更新 MaxMatrixCols 计数器, 即矩阵最大的列数。

4.5 matrix 类通用键值选项

array

expand expand = {none|text-expand|f|romanual|x|edef}

初始值 = none

对输入的元素进行展开, 即先对 array、diag、item 展开, 再进行处理。

rows rows = {整数}

cols cols = {整数}

auto-update auto-update = true|false

初始值 = false

是否自动更新行数和列数。

main

row-list 设置相应行、列的角标。

col-list

infinite 是否无穷维矩阵。

!infinite

```

element-code    element-code = {\code}
element-code*   element-code* = \except-empty|except-blank|except-dots|except-tl|
                                except-regex|only-regex|\macro name)

```

Updated: 2022-04-27

`element-code` 使用三个参数, 分别为 `main`、行角标、列角标。此选项设置每一个矩阵元素要显示的代码。`\physicx@matricelement` 宏保存这 `\code` (带有三个参数)。

`element-code*` 控制要排除的元素, 这些元素不使用 `element-code` 中的处理代码。一般情况下, 应该首先使用 `element-code`, 再使用 `element-code*`, 如果先使用 `element-code*`, 再使用 `element-code`, 则 `element-code*` 不会生效。

- `except-empty` 将排除空元素 (完全为空, 不包含任何字符);
- `except-blank` 排除空白元素 (包括空格);
- `element-dots` 排除 `\cdots`、`\vdots`、`\ldots`、`\ddots`;
- `except-tl` 排除 `\physicxexcept` 的中记号, 如果字符是一个整体要被排除, 应该使用花括号: `\def\physicxexcept{\{abc\}{ABC}}`, 将排除元素为 `abc` 或 `ABC` 的情况;
- `except-regex` 将排除那些匹配正则表达式的元素, 这个正则表达式保存在 `\physicxexcept` 中;
- `only-regex` 将排除那些不匹配正则表达式的元素, 这个正则表达式保存在 `\physicxexcept` 中;
- 它的值也可以是一个宏的名称, 它使用这个宏来处理。这个宏应有三个参数。

注意 `element-code*` 会重定义 `\physicx@matricelement`。

```

element-except
element-except+

```

这个键设置 (或附加) `\physicxexcept` 的值, 这个宏的值用于 `element-code*` 的相关设置中。

```

expand-element  expand-element = true|false

```

初始值 = `false`

是否对生成的矩阵值进行展开, 一般情况是不需要的, 当使用 `saveto` 保存到宏中, 则可能需要启用该选项。

```

empty

```

当矩阵的某元素未被设置时, 使用其值设置之。

```

check  check = \none|empty|ignore|igep|all

```

是否对输入的元素进行检查。

`empty` 选项检查当输入的元素为空时, 使用 `empty` 键的值替换之。

`ignore` 检查当输入的元素为 `\PHYSICXIGNORE` 时, 不对该元素进行设置操作。

`igep` 对 `empty` 和 `ignore` 进行检查。

```

row-iterate
col-iterate

```

设置行、列的角标的索引方法, 它们接收一个参数, 分别代表对应元素的行索引和列索引。

```

last-row
last-col

```

```

diag    diag = {\diag 键选项}
diag+   diag+ = {\diag 键选项}
diag-now diag-now = {\diag 键选项}
diag-data diag-data = {\预先定义的diag数据}
diag-data+ diag-data+ = {\预先定义的diag数据}

```

`diag-now` 立即解析给出的 `\diag` 键选项, 不待 `adi` 处理, 或 `adi` 处理完毕后还需设置矩阵元素。可用于 `beginning`、`ending` 键中进行额外的设置。

`diag-data` 可使用 `\setmatrixdata` 设置的数据, 见第 5 节。

```

item
item+
item-now
item-data
item-data+

```

```
check-range check-range = true|false
```

初始值 = **true**

当解析 `item` 选项时,是否检查元素索引越界,为真时,不设置越界元素。

```

begin begin = {<code>}
end   end   = {<code>}

```

设置输出矩阵时使用的环境。一般为 `begin=\begin{matrix},end=\end{matrix}`。

但是可以是任意代码,一般情况下 `begin` 和 `end` 需配对。

```

args args = <code>
args* args* = <code>

```

当 `begin` 键使用的环境需要参数时,可使用这个键设置。

`args` 设置方括号括起来的参数,将 [`<code>`] 放到 `begin` 键的代码之后。

`args*` 将 `<code>` 放到 `begin` 键的代码之后。

如:`begin=\begin{matrix*},args=c`,则为 `\begin{matrix*}[c]`。

`begin=\begin{array},args*=[t]{ccc}`,则为 `\begin{array}[t]{ccc}`。

```

after-begin
after-begin+
after-end
after-end+

```

```
sepdim sepdim = <dim>
```

初始值 = **Opt**

```
type type = <m|p|b|B|v|V|sm|...>
```

初始值 = **m**

设置输出矩阵的类型。`m` 使用 `matrix` 环境,`p` 使用 `pmatrix` 环境,等。

当加载了 `mathtools` 宏包或使用了 `mathtools` 宏包选项时,还额外定义了 `m*` 和 `sm*`、`sp`、`sp*` 等环境,它们代表 `martix*`、`smallmatrix*`、`psmallmatrix` 等环境。

还可使用由 `\setmatrixtype` 定义的类型。见第 5 节。

一般情况下,如果 `type` 的值与 `matrix` 类的键没有相同的名字,则可省略 `type`,直接写它的值。不能保证未来版本不会增加新的键,若要使得简写仍然有效,应该保证它的名字比较特别,使得出现相同键名的概率低。

```

saveto saveto = \<macro name>
saveto* saveto* = <macro name>

```

将矩阵的值保存到相应的宏中。

```
transpose transpose = true|false
```

初始值 = **false**

```

'
T

```

为真,则将矩阵元素转置,不对角标转置。

```
MaxMatrixCols MaxMatrixCols = <整数>
```

设置 `MaxMatrixCols` 计数器的值。一般情况下无需给出键名,直接使用 `<整数>` 即可。

```

enhanced enhanced = true|false
!enhanced

```

初始值 = **false**

是否启用 `enhanced` 模式。

<hr/>	<code>cr</code>	<code>cr</code>	= <符号>	初始值 = ;
	<code>align</code>	<code>align</code>	= <符号>	初始值 = ,
	<code>sep</code>	<code>sep</code>	= <符号>	初始值 = ,

设置分隔符。<符号>可以是任意记号,不会对其进行展开。

`sep` 设置 `diag`、`row-list`、`col-list` 的分隔符。

`cr` 设置换行 (\\) 对应的符号。`align` 设置 `align (&)` 对应的符号,用于 `array` 键或 `\commamatrix` 中。

<hr/>	<code>adi-order</code>	<code>adi-order</code>	= <adi dia iad aid ida dai>	初始值 = <code>adi</code>
-------	------------------------	------------------------	-----------------------------	------------------------

设置 `array/main/diag/item` 的处理顺序。

<hr/>	<code>beginning</code>	设置在 <code>adi</code> 处理前后要执行的代码,可以是任意代码,一般用于设置。
	<code>beginning+</code>	
	<code>ending</code>	
	<code>ending+</code>	

4.6 例子

例 13:

```
$\commamatrix[
  begin=\begin{array},end=\end{array},args*={{clr}}, % two {{ }}
]{A, E, I; MNOP, QRST, UVWX}$
$\commamatrix[
  begin=\left[\begin{array}, end=\end{array}\right], args*={{clr}},
]{A, E, I; MNOP, QRST, UVWX}$
```

$$\begin{array}{ccc} A & E & I \\ MNOP & QRST & UVWX \end{array} \left[\begin{array}{ccc} A & E & I \\ MNOP & QRST & UVWX \end{array} \right]$$

例 14:

```
$\generalmatrix+{rows=5,cols=4,main=A}$
$\generalmatrix+{array={ 1,2,3,4; a,b,c,d; \oplus,\otimes,\cup,\spadesuit},rows=3,cols=4,type=p}$
$\generalmatrix+{
  array={ 1,2,3,4; a,b,c,d; \oplus,\otimes,\cup,\spadesuit },
  type=p,auto-update
}$
```

$$\begin{array}{cccc} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \\ A_{51} & A_{52} & A_{53} & A_{54} \end{array} \left(\begin{array}{cccc} 1_{11} & 2_{12} & 3_{13} & 4_{14} \\ a_{21} & b_{22} & c_{23} & d_{24} \\ \oplus_{31} & \otimes_{32} & \cup_{33} & \spadesuit_{34} \end{array} \right) \left(\begin{array}{cccc} 1_{11} & 2_{12} & 3_{13} & 4_{14} \\ a_{21} & b_{22} & c_{23} & d_{24} \\ \oplus_{31} & \otimes_{32} & \cup_{33} & \spadesuit_{34} \end{array} \right)$$

例 15:

```

 $\backslash\text{generalmatrix} + \{$ 
  array={ 1,2,3,4; a,b,c,d;  $\backslash\text{oplus}$ , $\backslash\text{otimes}$ , $\backslash\text{cup}$ , $\backslash\text{spadesuit}$  },
  type=v,rows=3,cols=4,
  row-list={A,B,C}, col-list={ $\backslash\text{blacklozenge}$ , $\backslash\text{Box}$ , $\backslash\text{heartsuit}$ }
 $\}$ 
 $\backslash\text{generalmatrix} + * \{$ 
  array={ 1,2,3,4; a,b,c,d;  $\backslash\text{oplus}$ , $\backslash\text{otimes}$ , $\backslash\text{cup}$ , $\backslash\text{spadesuit}$  },
  type=v,rows=3,cols=4,
  row-list={A,B,C}, col-list={ $\backslash\text{blacklozenge}$ , $\backslash\text{Box}$ , $\backslash\text{heartsuit}$ }
 $\}$ 
 $\backslash\text{generalmatrix} + * \{$ 
  array={ 1,2,3,4; a,b,c,d;  $\backslash\text{oplus}$ , $\backslash\text{otimes}$ , $\backslash\text{cup}$ , $\backslash\text{spadesuit}$  },
  type=v,rows=3,cols=4, element-code*=except-dots,
  row-list={A,B,C}, col-list={ $\backslash\text{blacklozenge}$ , $\backslash\text{Box}$ , $\backslash\text{heartsuit}$ }
 $\}$ 

```

$$\begin{array}{c}
 \left| \begin{array}{cccc} 1_{A\spadesuit} & 2_{A\lozenge} & 3_{A\heartsuit} & 4_A \\ a_{B\spadesuit} & b_{B\lozenge} & c_{B\heartsuit} & d_B \\ \oplus_{C\spadesuit} & \otimes_{C\lozenge} & \cup_{C\heartsuit} & \spadesuit_C \end{array} \right| \left| \begin{array}{ccccc} 1_{A\spadesuit} & 2_{A\lozenge} & 3_{A\heartsuit} & 4_A & \cdots_A \\ a_{B\spadesuit} & b_{B\lozenge} & c_{B\heartsuit} & d_B & \cdots_B \\ \oplus_{C\spadesuit} & \otimes_{C\lozenge} & \cup_{C\heartsuit} & \spadesuit_C & \cdots_C \\ \vdots_{\spadesuit} & \vdots_{\lozenge} & \vdots_{\heartsuit} & \vdots & \ddots \end{array} \right| \left| \begin{array}{ccccc} 1_{A\spadesuit} & 2_{A\lozenge} & 3_{A\heartsuit} & 4_A & \cdots \\ a_{B\spadesuit} & b_{B\lozenge} & c_{B\heartsuit} & d_B & \cdots \\ \oplus_{C\spadesuit} & \otimes_{C\lozenge} & \cup_{C\heartsuit} & \spadesuit_C & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{array} \right|
 \end{array}$$

例 16:

```

 $\backslash\text{generalmatrix} + \{$ 
  rows=5,cols=5,type=v,
  main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} },
 $\}$ 
 $\backslash\text{generalmatrix} + \{$ 
  rows=5,cols=5,type=v, element-code={#1},
  main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} },
 $\}$ 
 $\backslash\text{generalmatrix} + \{$ 
  rows=5,cols=5,type=v, element-code={#1},
  main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} },
  item={ {1,3}{2,4}= $\backslash\text{Box}$  },
 $\}$ 

```

$$\begin{array}{c}
 \left| \begin{array}{ccccc} 1_{11} & A_{12} & A_{13} & A_{14} & -1_{15} \\ A_{21} & 2_{22} & A_{23} & -2_{24} & A_{25} \\ A_{31} & A_{32} & -3_{33} & A_{34} & A_{35} \\ A_{41} & -4_{42} & A_{43} & 4_{44} & A_{45} \\ -5_{51} & A_{52} & A_{53} & A_{54} & 5_{55} \end{array} \right| \left| \begin{array}{ccccc} 1 & A & A & A & -1 \\ A & 2 & A & -2 & A \\ A & A & -3 & A & A \\ A & -4 & A & 4 & A \\ -5 & A & A & A & 5 \end{array} \right| \left| \begin{array}{ccccc} 1 & \square & A & \square & -1 \\ A & 2 & A & -2 & A \\ A & \square & -3 & \square & A \\ A & -4 & A & 4 & A \\ -5 & A & A & A & 5 \end{array} \right|
 \end{array}$$

例 17:

```
% new a \generalmatrix command, use enhanced mode, with 1 arg
\newgeneralmatrix + \ffempty [1] { rows=5,cols=5,element-code={##1},type=v, #1 }
$\ffempty{ main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} } }$
$\ffempty{
  main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} },
  item={ {1,3}{2,4}=\Box }
}$
$\ffempty{
  main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} },
  item={
    {1,3}{2,4}=\Box, % (1,2), (1,4), (3,2), (3,4) = \Box
    {4}{5}=\PHYSICXIGNORE, % ignore setting of (4,5)
    {-}{5}=\blacksquare, % col=5
    {4}{-}=\heartsuit, % row=4
  }
}$
```

$$\begin{array}{c}
 \left| \begin{array}{ccccc} 1 & A & A & A & -1 \\ A & 2 & A & -2 & A \\ A & A & -3 & A & A \\ A & -4 & A & 4 & A \\ -5 & A & A & A & 5 \end{array} \right| \left| \begin{array}{ccccc} 1 & \square & A & \square & -1 \\ A & 2 & A & -2 & A \\ A & \square & -3 & \square & A \\ A & -4 & A & 4 & A \\ -5 & A & A & A & 5 \end{array} \right| \left| \begin{array}{ccccc} 1 & \square & A & \square & \blacksquare \\ A & 2 & A & -2 & \blacksquare \\ A & \square & -3 & \square & \blacksquare \\ \heartsuit & \heartsuit & \heartsuit & \heartsuit & A \\ -5 & A & A & A & \blacksquare \end{array} \right|
 \end{array}$$

例 18:

```
% xparse-like new command
\NewGeneralMatrix + \fftest { D(){adi} 0{ } } {
  rows=5,cols=5, element-code={##1},
  main=A, diag={ 0={1,2,3,4,5,6}, '0={-1,-2,-3,-4,-5,-6} },
  item={
    {1,3}{2,4}=\Box, % (1,2), (1,4), (3,2), (3,4) = \Box
    {4}{5}=\PHYSICXIGNORE, % ignore setting of (4,5)
    {-}{5}=\blacksquare, % col=5
    {4}{-}=\heartsuit, % row=4
  }, adi-order=#1, #2 }

$\fftest[type=v]$ % adi-order=adi, process order: array/main, diag, item
$\fftest(adi)[type=v]$ % adi-order=adi, process order: array/main, item, diag
$\fftest(adi)[type=v,
  diag+={ 2={HH,II,JJ}, -2={ , , } }, % add to diag
  item+={ {4}{1}={ } }, % add to item
  ending=\physicxset*{matrix}{ item-now={ {4}{4}=\fbox{LL} } }, % set item at ending
]$

$\fftest(adi)[type=v,
  diag+={ 2={HH,II,JJ}, -2={ , , } },
  item+={ {4}{1}={ } },
  ending=\physicxset*{matrix}{ item-now={ {4}{4}=\fbox{LL} } },
  saveto=\savetosomething,
]$
{\ttfamily\meaning\savetosomething}
```

```

1  □  A  □  ■ | 1  □  A  □  -1 | 1  □  HH  □  -1 |
A  2  A  -2  ■ | A  2  A  -2  ■ | A  2  A  II  ■ |
A  □  -3  □  ■ | A  □  -3  □  ■ |   □  -3  □  JJ |
♡  ♡  ♡  ♡  A | ♡  -4  ♡  4  A |   ♡  LL  A |
-5  A  A  A  ■ | -5  A  A  A  5 | -5  A  A  A  5 |
macro:->1\Box &HH&\Box &-1\\A&2&A&II&\blacksquare \\&\Box &-3&\Box &JJ\\
&&\heartsuit &\fbox {LL}&A\\-5&A&A&5
```

第 5 节 杂项

`\physicxset` `\physicxset {<通用键值选项>}`
`\physicxset * {<类型>} {<类型可用的键值选项>}`

`physicx` 宏包的设置命令。

<类型> 为 `quantity`、`matrix` 等。

`\setquantitytype` `\setquantitytype {<type name>} {<quantity类键选项>}`

自定义 `quantity` 类的 `type`。

`\physicxtmp` `\physicxempty` 保存 `empty` 键的值。`\physicxexcept` 保存 `element-except` 的值。
`\physicxempty` 永远不要在正文中使用 `\PHYSICXIGNORE`。仅应 `diag`、`item` 键的设置中使用。
`\physicxexcept`
`\PHYSICXIGNORE`

例 21:

```
% \catcode`\ =9 \catcode`\@=11
\@declarequantitycmd 1 1 \quantity
{
  { !g } { { \{ } { #4 } { \} } }
  { !o } { { [ } { #5 } { ] } }
  { !d() } { { ( } { #6 } { ) } }
  { !d|| } { { \vert } { #7 } { \vert } }
  { !d<> } { { \langle } { #8 } { \rangle } }
  { !d== } { { \Vert } { #9 } { \Vert } }
}
\@declarequantitycmd 1 0 \matrixquantity
{
  { !g }
  {
    { \IfBooleanT{#3}{\left\{ } }
    { \begin{matrix} #4 \end{matrix} }
    { \IfBooleanT{#3}{\right\} }
  }
  { !o } { { \begin{bmatrix} } { #5 } { \end{bmatrix} } }
  { !d() }
  {
    { \IfBooleanTF{#3}{\left\lgroup}{\left( } }
    { \begin{matrix} #6 \end{matrix} }
    { \IfBooleanTF{#3}{\right\rgroup}{\right) } }
  }
  { !d|| } { { \begin{vmatrix} } { #7 } { \end{vmatrix} } }
  { !d<> } { { \left\langle } { \begin{matrix} #8 \end{matrix} } { \right\rangle } }
  { !d== } { { \begin{Vmatrix} } { #9 } { \end{Vmatrix} } }
}
}
```

例 22:

```
% \catcode`\ =9 \catcode`\@=11
\@declareparencmd \pqty { m } { #6 } { } ( ) { }
\@declareparencmd \absolutevalue { m } { #6 } { } \vert \vert { }
\@declareparencmd \OOrder { m } { #6 } { \mathcal{0} } ( ) { }
\@declareparencmd \commutator { m m } { #6 , #7 } { } [ ] { }
```

5.2 定义新的矩阵命令

<u>\newdiagonalmatrix</u>	\newdiagonalmatrix	\<cmd>	[<arg nums>]	[<default>]	{<matrix keys>}	{<diag keys>}
\NewDiagonalMatrix	\newdiagonalmatrix +	\<cmd>	[<arg nums>]	[<default>]	{<matrix keys>}	{<diag keys>}
\newcommamatrix	\NewDiagonalMatrix	\<cmd>	{<args spec>}	{<matrix keys>}	{<diag keys>}	
\NewCommaMatrix	\NewDiagonalMatrix +	\<cmd>	{<args spec>}	{<matrix keys>}	{<diag keys>}	
<u>\newgeneralmatrix</u>						
\NewGeneralMatrix	\newcommamatrix	\<cmd>	[<arg nums>]	[<default>]	{<matrix keys>}	{<comma value>}
	\newcommamatrix +	\<cmd>	[<arg nums>]	[<default>]	{<matrix keys>}	{<comma value>}
	\NewCommaMatrix	\<cmd>	{<args spec>}	{<matrix keys>}	{<comma value>}	
	\NewCommaMatrix +	\<cmd>	{<args spec>}	{<matrix keys>}	{<comma value>}	
	\newgeneralmatrix	\<cmd>	[<arg nums>]	[<default>]	{<matrix keys>}	
	\newgeneralmatrix +	\<cmd>	[<arg nums>]	[<default>]	{<matrix keys>}	
	\NewGeneralMatrix	\<cmd>	{<args spec>}	{<matrix keys>}		
	\NewGeneralMatrix +	\<cmd>	{<args spec>}	{<matrix keys>}		

例 23:

```

\newdiagonalmatrix \xdmat [2] [] {#1} {0={#2}}
$ \xdmat{1,2,3,4}$ \quad $\xdmat[v]{1,2,3,4}$ \quad $\xdmat[v,empty=0]{1,2,3,4}$
\quad
$ \xdmat{1, \commamatrix{2&3\\4&5}} $ \quad
$ \xdmat{1, \mqty{2&3\\4&5}} $ \quad

\newdiagonalmatrix \xxdmat [3] [] {#1, diag={true, 0={#2}, #3} } {}
\newdiagonalmatrix \xadmat [2] [] {#1} {auto-update,'0={#2}}
$ \xxdmat{1,2,3,4}-{0={a,b,c,d}} $ \quad
$ \xxdmat[b]{1,2,5}-{1={3,4}} $ \quad
$ \xadmat{a,b,c,d} $

```

$$\begin{array}{c}
 \begin{array}{cccc}
 1 & & & \\
 & 2 & & \\
 & & 3 & \\
 & & & 4
 \end{array}
 \left| \begin{array}{c}
 1 \\
 2 \\
 3 \\
 4
 \end{array} \right|
 \left| \begin{array}{cccc}
 1 & 0 & 0 & 0 \\
 0 & 2 & 0 & 0 \\
 0 & 0 & 3 & 0 \\
 0 & 0 & 0 & 4
 \end{array} \right|
 \begin{array}{cccc}
 1 & & & 1 \\
 & 2 & 3 & \\
 & & & 2 & 3 \\
 & 4 & 5 & 4 & 5
 \end{array}
 \end{array}$$

$$\begin{array}{c}
 \begin{array}{cccc}
 1 & & a & \\
 & 2 & b & \\
 & c & 3 & \\
 d & & & 4
 \end{array}
 \left[\begin{array}{cc}
 1 & \\
 & 2 & 3 \\
 & & 4 & 5
 \end{array} \right]
 \begin{array}{cccc}
 & & a & \\
 & & & b \\
 & c & & \\
 & & d &
 \end{array}
 \end{array}$$

`\newgeneralmatrix`、`\NewGeneralMatrix` 的例子见例 17 和例 18。

版本历史

v0.1.2	(2021/11/11)	移除 \txqty 命令。	1
General: 新增 quantity 选项。	1		
v0.1.3	(2021/11/12)	v0.3	(2021/11/27)
General: 将 quantity 重命名为 reqty。	1	General: 简单兼容 unicode-math 宏包	1
v0.2	(2021/11/12 – 2021/11/13)	v0.3.3	(2022/07/30)
General: 新增 \xquantity、\qxqty、\txqty 命令。	1	General: 修改 sepdim 的行为, 在其为 0pt 时, 不显示使用	
旧的括号命令不再推荐使用, 仅由于兼容性原因保留。	3	\\[0.0pt]。	13
移除 \Order 命令。	3	对 fixdif 宏包提供支持。	1
v0.2.1	(2021/11/14)	v0.3.4	(2022/09/13)
General: 修复矩阵类不能使用 type 简写的错误。	3	General: 修复文档错误。	1
		新增 original 选项。 (#8)	1

代码索引

意大利体的数字表示描述对应索引项的页码; 带下划线的数字表示定义对应索引项的代码行号; 罗马字体的数字表示使用对应索引项的代码行号。

Symbols		cols	11
!infinite	11	\commatrix	5, 8, 14
'	13	cr	14
'-1	6		
'1	6	D	
-1	6	\dd	1
.	6	diag	12
\⟨cmd⟩	18, 19	diag+	12
\⟨macro name⟩	13	diag-data	12
		diag-data+	12
Numbers		diag-now	12
0	6	\diagonalmatrix	5–8
1	6		
		E	
A		element-code	12
adi-order	14	element-code*	12
after-begin	13	element-except	12
after-begin+	13	element-except+	12
after-end	13	empty	12
after-end+	13	end	13
align	14	ending	14
args	2, 13	ending+	14
args*	2, 13	enhanced	13
array	11	expand	11
auto-update	6, 11	expand-element	12
B		F	
begin	13	false	6
beginning	14		
beginning+	14	G	
		\genegralmatrix	5
C		\generalmatrix	6, 11
check	12		
check-range	13	I	
code	2	infinite	11
col-iterate	12	item	13
col-list	11	item+	13

item-data	13		
item-data+	13		
item-now	13		
L			R
last-col	12	right	2
last-row	12	right-size	2
left	2	row-iterate	12
left-size	2	row-list	11
		rows	11
M			S
main	11	saveto	13
\matrixquantity	4	saveto*	13
MaxMatrixCols	13	sep	14
\mqty	4	sepdim	13
		\setmatrixdata	12, 18
N		\setmatrixtype	13, 18
\NewCommaMatrix	19	\setquantitytype	17
\newcommamatrix	19	size	2
\NewDiagonalMatrix	19	\smallmatrixquantity	4
\newdiagonalmatrix	19	\smqty	4
\NewDocumentCommand	18		T
\NewGeneralMatrix	19, 20	T	13
\newgeneralmatrix	19, 20	T _E X and L ^A T _E X ₂ _ε commands:	
\NewXQuantity	18	\@declareparencmd	18
\newxquantity	18	\@declarequantitycmd	18
noauto	2	\Big	18
noauto-update	6	\big	18
		\Bigg	18
O		\bigg	18
\OOrder	3	\cdots	12
\oorder	3	\ddots	12
\order	3	\div	1
		\Im	1
P		\ldots	12
\physicxempty	17	\left	2, 3
\physicxexcept	12, 17	\newcommand	18
\PHYSICXIGNORE	12, 17	\physicx@matricelement	12
\physicxset	17	\Re	1
\physicxtmp	6, 17	\right	2, 3
post	2	\setcounter	8
pre	2	\vdots	12
		transpose	13
Q		true	6
\quantity	4	type	3, 13
\qxmatrix	5, 9-11		X
\qxqty	1, 18	\xmatrix	9
		\xquantity	1, 18