

The xymatrix package

The `xymatrix` package is included in the graphics package `Xy-pic`. It is possible to use `Xy-pic` directly for drawing commutative diagrams,¹ and some recommend its component `xygraph` for complicated diagrams, but I will only discuss `xymatrix`.

The documentation for `xymatrix` is embedded in that for `Xy-pic`. There are a user's guide and reference manual available at <http://www.maths.mq.edu.au/~ross/Xy-pic.html>.

The package supports a great variety of arrows, including curved arrows. Arrows do not automatically stretch to match their labels, but this can be done manually. To load it for this document, I used: `\usepackage[all,cmtip]{xy}`.

The syntax is illustrated by

$$\begin{array}{ccc}
 A & \xrightarrow{a} & B \\
 \downarrow b & & \downarrow c \\
 C & \xrightarrow{d} & D
 \end{array}
 \quad
 \begin{array}{l}
 \text{\code{xymatrix}\{ \\
 A \text{\code{\ar[d]}^b \text{\code{\ar[r]}^a} \&B\text{\code{\ar[d]}^c}\ \\
 C \text{\code{\ar[r]}^d} \quad \quad \quad \&D\}
 \end{array}$$

Note that only objects are separated by `&` and that all arrows are attached to their source. In contrast to `array`, the diagram when set with `xymatrix` has only two lines of code.

Arrows are specified in the form `\ar@{shape}[dir]` where `shape` describes the shape of the arrow and `dir` is a sequence of single letters — l for left, r for right, u for up, d for down — describing the direction of the arrow. For example, `\ar[rd]` is an arrow from its source to a target one right and one down. Here are some of the possible arrows:

$$\begin{array}{ll}
 A \longrightarrow B & \text{\code{\ar[r]}} \\
 A \longmapsto B & \text{\code{\ar@{|->}[r]}} \\
 A \longdashrightarrow B & \text{\code{\ar@{-}[r]}} \\
 A \curvearrowright B & \text{\code{\ar@{^}{()->}[r]}} \\
 A \twoheadrightarrow B & \text{\code{\ar@{->>}[r]}} \\
 A \cdots\cdots\rightarrow B & \text{\code{\ar@{.}>}[r]} \\
 A \dashrightarrow B & \text{\code{\ar@{-->}[r]}} \\
 A \Longrightarrow B & \text{\code{\ar@{=>}[r]}} \\
 A \rightsquigarrow B & \text{\code{\ar@{~>}[r]}}
 \end{array}$$

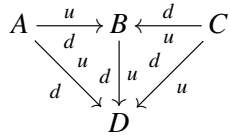
For example:

$$\begin{array}{ccc}
 A & \xrightarrow{\quad} & B & \xleftarrow{\quad} & C \\
 & \searrow & & \vdots & \swarrow \\
 & & & D & &
 \end{array}
 \quad
 \begin{array}{l}
 \text{\code{xymatrix}\{ \\
 A \text{\code{\ar@{->>}[rd]} \text{\code{\ar@{^}{()->}[r]} \\
 \&B \text{\code{\ar@{.}>}[d]} \\
 \&C \text{\code{\ar@_{_}{()->}[1]} \text{\code{\ar@{->>}[1d]}\ \\
 \&D\}
 \end{array}$$

To get labels on arrows, place them as superscripts or subscripts on the arrow (between braces if necessary), as illustrated by:

This is part of: Guide to Commutative Diagrams, www.jmilne.org/not/CDGuide.html
 September 23, 2010

¹See the articles by Paul Blaga in The *PracTeX Journal*, Issue 4, 2006, and Issue 1, 2007



```

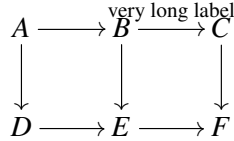
\xymatrix{
A \ar[r]^u_d \ar[rd]^u_d &
B \ar[d]^u_d &
C \ar[l]^u_d \ar[ld]^u_d \\
& & & & D}

```

Note that the position of the label rotates with the arrow. To have a label break an arrow, use the syntax: `\ar[r]|f`

$$A \xrightarrow{|f} B.$$

Arrows don't automatically stretch to match long labels,

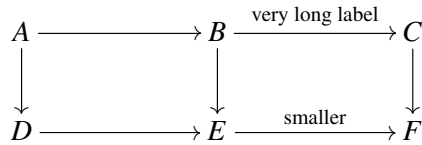


but you can fix this manually by stretching all the horizontal arrows (increasing the column separation):

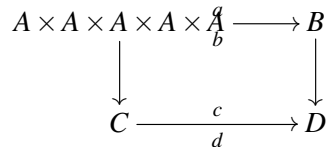
```

\[\xymatrixcolsep{5pc}\xymatrix{...}

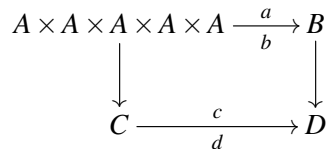
```



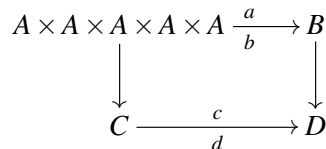
Arrows do stretch (or contract) to match large objects, but then the placement of the label may be wrong (it is placed halfway between the centres of the source and target objects)



To centre the label on the arrow, replace `\ar[r]^a_b` with `\ar[r]^~a_b`



It is also possible to move the labels manually by using, for example, `\ar[r]^<<<a_<<<b \ar[d]`



Objects are positioned using the "centre" of the object, which may not be what you want:

$$\hat{A} \longrightarrow \prod_{n \in \mathbb{Z}} A_n \longrightarrow \prod_{n \in \mathbb{Z}} A_n.$$

Replacing `\xymatrix` with `\xymatrix@1` helps a little:

$$\hat{A} \longrightarrow \prod_{n \in \mathbb{Z}} A_n \longrightarrow \prod_{n \in \mathbb{Z}} A_n.$$

Fortunately, Alexander Perlis² has found a fix for the problem, namely, add the line

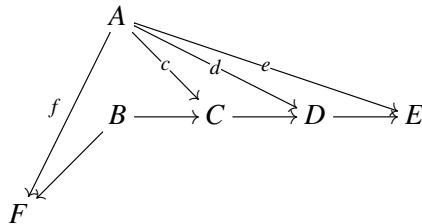
`\entrymodifiers={+!!<0pt,\fontdimen22\textfont2>}`

prior to each `\xymatrix`, or simply add it once and for all in the document's preamble:

$$\hat{A} \longrightarrow \prod_{n \in \mathbb{Z}} A_n \longrightarrow \prod_{n \in \mathbb{Z}} A_n.$$

An arrow that goes, for example, 3 cells right and 1 down, is invoked by `\ar[rrrd]`, as illustrated by:

```
\[
\xymatrix{
&A \ar[ldd]_f \ar[rd] | -{c} \ar[rrd] | -{d} \ar[rrrd] | -{e} \\
&B \ar[ld] \ar[r] &C \ar[r] &D \ar[r] &E \\
&F}
\]
```



To curve an arrow, insert `/~/` or `/_/_`, as in

$$A \overset{\curvearrowright}{\longrightarrow} B \quad \xymatrix{A \ar@{~}/[r] \ar@{/_}/[r] & B}$$

Add a dimension to increase the curvature:

$$A \overset{\curvearrowright}{\longrightarrow} B \quad A \ar@{~}/[r] \ar@{~}^1pc/[r] \ar@{~}^2pc/[r] & B$$

Change the arrow as follows:

$$A \overset{\curvearrowright}{\twoheadrightarrow} B \quad A \ar@{->>}@{~}/[r] & B$$

The next example illustrates most of these capabilities:

```
\[
\xymatrix{
T \ar@/_/[ddr]_y \ar@/^/[drr]^x \ar@{.}>[dr] | -{(x,y)} \\
&X \times_{\mathbb{Z}} Y \ar[d]^q \ar[r]_p & X \ar[d]_f \\
&Y \ar[r]^g & Z}
\]
```

²<http://math.arizona.edu/~aprl/publications/axisalignment/>

