Display Math in ConTEXt

ConTEXt rehab for amsmath addicts

Abstract
This article explains how to do various kinds of alignments in ConTEXt. A visual output is presented, and it is then shown how that effect can be achieved in LaTEX and ConTEXt. We hope that article will make the transition from LaTEX with `amsmath` package to ConTEXt easier.

Keywords
ConTEXt, LaTEX, math alignment, amsmath

Introduction

Plain TEX provides several macros like `\eqalign`, `\eqalignno`, `\displaylines`, `\matrix`, `\pmatrix`, `\cases`, and `\halign`, for math alignments. These macros are adequate for most constructions that occur in practice. AMS-TeX and the `amsmath` package for LaTEX supply math alignment environments that provide a layer of abstraction for the user and makes it (slightly) easier for him/her to type the common math alignments. Most people learning TEX these days start from LaTEX and those writing substantial math use the `amsmath` package; they know nothing about the plain TEX math alignment macros. In earlier versions of ConTEXt, since the plain TEX macros could be used, no additional macros for math alignments were provided. This made writing math alignments difficult for users who came to ConTEXt with a LaTEX background. They did not know about plain TEX macros and kept looking for something equivalent to the `amsmath` package. There was an `amsl` package module by Giuseppe Bilotta, but it was very limited. Moreover, doing alignments with multiple alignment points in plain TEX requires a good understanding of the TEX alignment mechanism; making them obscure for a typical user. This resulted in a general impression that ConTEXt does not handle math very well.

Recently (in January 2006 to be precise), Hans added math alignment macros in ConTEXt. These macros provide a very nice user interface to plain TEX's alignment mechanism; they can be used to achieve the functionality of `amsmath` package macros; and, like all user macros in ConTEXt, they are easy to customize. These macros, however, neither copy the user interface of `amsmath` package, nor the implementation. So, translating your existing LaTEX math code into ConTEXt requires some effort and the result is not necessarily, pixel by pixel, identical.

In this article, I describe how to convert the common alignment constructions from LaTEX to ConTEXt, highlighting some of the flexibility offered by ConTEXt. This is a *visual* document: I first show how the output should look like, then present LaTEX and ConTEXt examples that give that output. This article is not meant as a tutorial for math alignments in LaTEX or ConTEXt, and I do not explain the LaTEX and the ConTEXt syntax. The article is not exhaustive; it provides a small sample of math alignments that can be done using LaTEX and ConTEXt. For an indepth treatment of LaTEX's math capabilities see Herbert Voß's mathmode.[1] For a introduction to ConTEXt math alignment see My Way on `\startalign` and friends.[2] The objective

of this article is not to compare the features of these two macro packages, rather it is to show that ConTEXt is capable of handling "complicated" math alignments.

Math Alignments

ConTEXt provides `mathalignment` series of macros (`\definemathalignment`, `\setupmathalignment`, `\startmathalignment`, and `\stopmathalignment`) to take care of the different math alignments. Below, I describe some common math constructs, and examples of how to achieve them in LaTEX and ConTEXt.

gather
The `gather` environment of `amsmath` package allows you to write multi–line formulas with each line center aligned. It is perhaps the simplest form of "alignment". In ConTEXt the same effect can be achieved using appropriate options to `\startmathalignment`.

$$\blacksquare = \blacksquare + \blacksquare + \blacksquare + \blacksquare$$
$$\blacksquare = \blacksquare + \blacksquare + \blacksquare + \blacksquare + \blacksquare$$

In LaTEX

```
\begin{gather}
  v = u  + at,           \\
  d = ut + \frac12 at^2.
\end{gather}
```

$$v = u + at, \tag{1}$$
$$d = ut + \frac{1}{2}at^2. \tag{2}$$

In ConTEXt

```
\placeformula \startformula
  \startmathalignment[n=1]
    \NC v = u  + at,           \NR[+]
    \NC d = ut + \frac12 at^2. \NR[+]
  \stopmathalignment
\stopformula
```

$$v = u + at, \tag{1}$$
$$d = ut + \frac{1}{2}at^2. \tag{2}$$

left gather
Sometimes one wants multi–line formulas, where each line is left or right aligned, rather than center aligned as in the `gather` environment. Although, LaTEX does not provide any in–built environment for such constructions, it is easy to exploit the `align` environment to achieve this output. In ConTEXt passing `align=left` to `\startmathalignment` gives the desired output.

$$\blacksquare = \blacksquare + \blacksquare$$
$$\blacksquare = \blacksquare + \blacksquare + \blacksquare + \blacksquare + \blacksquare + \blacksquare$$

In LaTeX

```
\begin{align}
  & v = u  + at,          \\
  & d = ut + \frac12 at^2.
\end{align}
```

$$v = u + at, \tag{1}$$

$$d = ut + \frac{1}{2}at^2. \tag{2}$$

In ConTeXt

```
\placeformula \startformula
  \startmathalignment[n=1,align=left] %align=left does the magic
    \NC v = u  + at,           \NR[+]
    \NC d = ut + \frac12 at^2. \NR[+]
  \stopmathalignment
\stopformula
```

$$v = u + at, \tag{1}$$

$$d = ut + \frac{1}{2}at^2. \tag{2}$$

right gather
For multi–line formulas with each line right aligned, in LaTeX you can exploit the `align` environment, while in ConTeXt you need to pass `align=right` to `\startmathalignment`

$$\blacksquare\blacksquare = \blacksquare\blacksquare + \blacksquare\blacksquare + \blacksquare\blacksquare\blacksquare + \blacksquare\blacksquare$$

$$\blacksquare\blacksquare = \blacksquare\blacksquare + \blacksquare\blacksquare$$

In LaTeX

```
\begin{align}
  v = u  + at ,          & \\
  d = ut + \frac12 at^2.  &
\end{align}
```

$$v = u + at, \tag{1}$$

$$d = ut + \frac{1}{2}at^2. \tag{2}$$

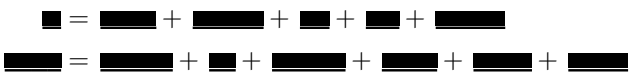
In ConTeXt

```
\placeformula \startformula
  \startmathalignment[n=1,align=right] %align=right does the magic
    \NC v = u  + at,           \NR[+]
    \NC d = ut + \frac12 at^2. \NR[+]
  \stopmathalignment
\stopformula
```

$$v = u + at, \tag{1}$$

$$d = ut + \frac{1}{2}at^2. \tag{2}$$

align
This is the simplest and the most widely used form of alignment. In the simplest case, there are two columns, one right aligned and the other left aligned. In LaTeX the `align` environment takes care of such alignments; in ConTeXt `\startmathalignment`.



In LaTEX

```
\begin{align}
  v &= u  + at,           \\
  d &= ut + \frac12 at^2.
\end{align}
```

$$v = u + at, \tag{1}$$

$$d = ut + \frac{1}{2}at^2. \tag{2}$$

In ConTEXt

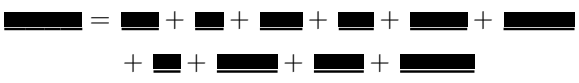
```
\placeformula \startformula
  \startmathalignment
    \NC v \NC = u  + at,           \NR[+]
    \NC d \NC = ut + \frac12 at^2. \NR[+]
  \stopmathalignment
\stopformula
```

$$v = u + at, \tag{1}$$

$$d = ut + \frac{1}{2}at^2. \tag{2}$$

split

The `split` environment of `amsmath` package is used for writing a single formula which needs more than one line. The whole formula gets a single number. In ConTEXt you have to manually specify which line to number.



In LaTEX

```
\begin{equation} \begin{split}
  (x+1)^8 ={} & x^8 + 8 x^7 + 28 x^6 + 56 x^5 + 70 x^4 \\
              & + 56 x^3 + 28 x^2 + 8 x + 1.
\end{split} \end{equation}
```

$$\begin{aligned}(x+1)^8 = {} & x^8 + 8x^7 + 28x^6 + 56x^5 + 70x^4 \\ & + 56x^3 + 28x^2 + 8x + 1.\end{aligned} \tag{1}$$

In ConTEXt

```
\placeformula \startformula
  \startmathalignment
  \NC (x+1)^8 = \NC x^8 + 8 x^7 + 28 x^6 + 56 x^5 + 70 x^4 \NR
  \NC           \NC + 56 x^3 + 28 x^2 + 8 x + 1.           \NR[+]
  \stopmathalignment
\stopformula
```

$$\begin{aligned}(x+1)^8 = {} & x^8 + 8x^7 + 28x^6 + 56x^5 + 70x^4 \\ & + 56x^3 + 28x^2 + 8x + 1.\end{aligned} \tag{1}$$

Notice that in LaTeX the spacing around $=$ in the first line has to be manually corrected by typing `{}`. ConTeXt takes care of this automatically. In LaTeX you can control the position of the tag with `tbtags` or `centertags` option to the `amsmath` package. Right now, with the math alignment macros in ConTeXt you can only achieve the result equivalent to `tbtags`. To get the result equivalent to `centertags` you have to use the `\eqalign` macro of plain TeX.

alignat

The `alignat` environment of `amsmath` package allows you to align at several places. The alignment order alternates between right and left aligned columns. In ConTeXt the same effect can be achieved by `m=...` option of `\startmathalignment`.

In LaTeX

```
\begin{alignat}{2}
  \nabla\cdot  \mathbf E &=  \frac{\rho}{\varepsilon_0},  \qquad
& \nabla\times \mathbf E &= -\frac{\partial \mathbf B}{\partial t},\\
  \nabla\cdot  \mathbf B &= 0,
& \nabla\times \mathbf B &= \mu_0{\mathbf j}+\varepsilon_0\mu_0
                            \frac{\partial \mathbf E}{\partial t}.
\end{alignat}
```

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0}, \qquad \nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}, \tag{1}$$

$$\nabla \cdot \mathbf{B} = 0, \qquad \nabla \times \mathbf{B} = \mu_0 \mathbf{j} + \varepsilon_0 \mu_0 \frac{\partial \mathbf{E}}{\partial t}. \tag{2}$$

In ConTeXt

```
\placeformula \startformula
  \startmathalignment[m=2,distance=2em]%Notice distance=2em
  \NC \nabla\cdot  \bf E \NC=  \frac{\rho}{\varepsilon_0},
  \NC \nabla\times \bf E \NC= -\frac{\partial \bf B}{\partial t},\NR[+]
  \NC \nabla\cdot  \bf B \NC= 0,
  \NC \nabla\times \bf B \NC= \mu_0{\bf j}+\varepsilon_0\mu_0
                              \frac{\partial \bf E}{\partial t}. \NR[+]
 \stopmathalignment
\stopformula
```

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0}, \qquad \nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}, \tag{1}$$

$$\nabla \cdot \mathbf{B} = 0, \qquad \nabla \times \mathbf{B} = \mu_0 \mathbf{j} + \varepsilon_0 \mu_0 \frac{\partial \mathbf{E}}{\partial t}. \tag{2}$$

flalign

The `flalign` environment is the same as `alignat` environment but with the equations a little more "out spaced". In ConTeXt you can control the space between the "blocks" by `distance=...` option to `\startmathalignment`.

In LaTEX

```
\begin{flalign*}
  \nabla\cdot  \mathbf E &=  \frac{\rho}{\varepsilon_0},
& \nabla\times \mathbf E &= -\frac{\partial \mathbf B}{\partial t}.\\
  \nabla\cdot  \mathbf B &= 0,
& \nabla\times \mathbf B &= \mu_0{\mathbf j}+\varepsilon_0\mu_0
                            \frac{\partial \mathbf E}{\partial t}.
\end{flalign*}
```

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0}, \qquad \nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}.$$

$$\nabla \cdot \mathbf{B} = 0, \qquad \nabla \times \mathbf{B} = \mu_0 \mathbf{j} + \varepsilon_0 \mu_0 \frac{\partial \mathbf{E}}{\partial t}.$$

In ConTEXt

```
\startformula
  \startmathalignment[m=2,distance=2em plus 1 fil]%Notice distance=...
   \NC \nabla\cdot  \bf E \NC=  \frac{\rho}{\varepsilon_0}, \NR
   \NC \nabla\times \bf E \NC= -\frac{\partial \bf B}{\partial t}, \NR
   \NC \nabla\cdot  \bf B \NC= 0,
   \NC \nabla\times \bf B \NC= \mu_0{\bf j}+\varepsilon_0\mu_0
                               \frac{\partial \bf E}{\partial t}.  \NR
 \stopmathalignment
\stopformula
```

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0}, \qquad \nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t},$$

$$\nabla \cdot \mathbf{B} = 0, \qquad \nabla \times \mathbf{B} = \mu_0 \mathbf{j} + \varepsilon_0 \mu_0 \frac{\partial \mathbf{E}}{\partial t}.$$

intertext
The `\intertext` macro from `amsmath` allows you to break the alignment and write some text, which does not affect the alignment. ConTEXt provides the `\intertext` macro and a `\startintertext`, `\stopintertext` environment for the same.

$$\blacksquare = \blacksquare + \blacksquare + \blacksquare + \blacksquare$$

$$\blacksquare \quad \blacksquare \quad \blacksquare \quad \blacksquare \quad \blacksquare$$

$$\blacksquare = \blacksquare + \blacksquare + \blacksquare + \blacksquare + \blacksquare + \blacksquare$$

In LaTEX

```
\begin{align*}
  \cos 2\theta &= \cos^2 \theta + \sin^2 \theta \\
  \intertext{replace $\sin^2 \theta$ by $1 - \cos^2 \theta$}
  &= 2\cos^2 \theta - 1
\end{align*}
```

$$\cos 2\theta = \cos^2\theta + \sin^2\theta$$

replace $\sin^2\theta$ by $1 - \cos^2\theta$

$$= 2\cos^2\theta - 1$$

In ConTEXt

```
\startformula
  \startmathalignment
    \NC \cos 2\theta \NC= \cos^2 \theta + \sin^2 \theta \NR
    \intertext{replace $\sin^2 \theta$ by $1 - \cos^2 \theta$}
    \NC \NC = 2\cos^2 \theta - 1 \NR
  \stopmathalignment
\stopformula
```

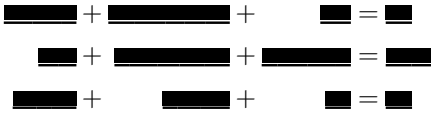
$$\cos 2\theta = \cos^2\theta + \sin^2\theta$$

replace $\sin^2\theta$ by $1 - \cos^2\theta$

$$= 2\cos^2\theta - 1$$

linear equations

In LaTeX linear equations can be handled using `alignat` environment; in ConTEXt appropriate options to `\startmathalignment` take care of this construction.[3]



In LaTEX

```
\begin{alignat}{5}
    x_1  & {} + {}& x_2     &{} + {}& 6x_3  &{} = {}& 170, \\
   3x_1  & {} - {}& 110x_2  &{} - {}& x_3   &{} = {}& 4,   \\
  14x_1  & {} + {}& 13x_2   &{} + {}& 10x_3 &{} = {}& 25.
\end{alignat}
```

$$\begin{aligned} x_1 + \quad x_2 + \; 6x_3 &= 170, && (1)\\ 3x_1 - 110x_2 - \quad x_3 &= \quad 4, && (2)\\ 14x_1 + \; 13x_2 + 10x_3 &= \; 25. && (3) \end{aligned}$$

In ConTEXt

```
\placeformula \startformula
  \startmathalignment
                [n=7,align={right,left,right,left,right,left,right}]
    \NC   x_1  \NC + \NC x_2     \NC + \NC 6x_3  \NC = \NC 170, \NR[+]
    \NC  3x_1  \NC - \NC 110x_2  \NC - \NC x_3   \NC = \NC 4,   \NR[+]
    \NC 14x_1  \NC + \NC 13x_2   \NC + \NC 10x_3 \NC = \NC 25.  \NR[+]
  \stopmathalignment
\stopformula
```

$$x_1 + x_2 + 6x_3 = 170, \tag{1}$$
$$3x_1 - 110x_2 - x_3 = 4, \tag{2}$$
$$14x_1 + 13x_2 + 10x_3 = 25. \tag{3}$$

In LaTEX we are limited to left and right aligned columns. In ConTEXt it is easy to change the alignment of individual columns. For example

```
\placeformula \startformula
  \startmathalignment[n=7,
           align={middle,middle,middle,middle,middle,middle,middle}]
    \NC   x_1  \NC + \NC x_2     \NC + \NC 6x_3  \NC = \NC 170, \NR[+]
    \NC  3x_1  \NC - \NC 110x_2  \NC - \NC x_3   \NC = \NC 4,   \NR[+]
    \NC 14x_1  \NC + \NC 13x_2   \NC + \NC 10x_3 \NC = \NC 25.  \NR[+]
  \stopmathalignment
\stopformula
```

$$x_1 + x_2 + 6x_3 = 170, \tag{1}$$
$$3x_1 - 110x_2 - x_3 = 4, \tag{2}$$
$$14x_1 + 13x_2 + 10x_3 = 25. \tag{3}$$

multi–column numbered equations
Sometimes, while writing formulas in blocks, you need to number formulas in all blocks. I do not know of any easy way to do this in LaTEX. Herbert Voß's Mathmode[1] has an example in Section 73 of using `tabular` to achieve this effect. ConTEXt provides `\startformulas` for multi–column formulas, which allows numbering of formulas in each column.

$$\blacksquare = \blacksquare + \blacksquare + \blacksquare \tag{1}$$
$$\blacksquare = \blacksquare + \blacksquare + \blacksquare$$
$$\blacksquare = \blacksquare \tag{2}$$
$$\blacksquare = \blacksquare$$

```
\placeformula \startformulas
 \startformula \startmathalignment
  \NC \nabla\cdot  \bf E \NC=  \frac{\rho}{\varepsilon_0}, \NR[+]
  \NC \nabla\cdot  \bf B \NC= 0,                          \NR[+]
 \stopmathalignment \stopformula
 \startformula \startmathalignment
  \NC \nabla\times \bf E \NC= -\frac{\partial \bf B}{\partial t},\NR[+]
  \NC \nabla\times \bf B \NC= \mu_0{\bf j}+\varepsilon_0\mu_0
                              \frac{\partial \bf E}{\partial t}. \NR[+]
 \stopmathalignment \stopformula
\stopformulas
```

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0}, \tag{1}$$

$$\nabla \cdot \mathbf{B} = 0, \tag{2}$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}, \tag{3}$$

$$\nabla \times \mathbf{B} = \mu_0 \mathbf{j} + \varepsilon_0 \mu_0 \frac{\partial \mathbf{E}}{\partial t}. \tag{4}$$

Defining your own alignments
In the examples given above, I passed the arguments to `\startmathalignment`. This can be repetitive if you have to use the same alignment many times. ConTEXt provides `\definemathalignment` to define a new math alignments. Suppose you have to type a lot of linear equations, you can define your own alignment as follows

```
\definemathalignment
  [linearequations]
  [n=7,align={middle,middle,middle,middle,middle,middle,middle}]

\placeformula \startformula
  \startlinearequations
    \NC   x_1  \NC + \NC x_2     \NC + \NC 6x_3  \NC = \NC 170, \NR[+]
    \NC  3x_1  \NC - \NC 110x_2  \NC - \NC x_3   \NC = \NC 4,   \NR[+]
    \NC 14x_1  \NC + \NC 13x_2   \NC + \NC 10x_3 \NC = \NC 25.  \NR[+]
  \stoplinearequations
\stopformula
```

$$\begin{aligned} x_1 + x_2 + 6x_3 &= 170, && (1)\\ 3x_1 - 110x_2 - x_3 &= 4, && (2)\\ 14x_1 + 13x_2 + 10x_3 &= 25. && (3) \end{aligned}$$

You can define similar alignments for each special case that you have to use.

Matrix and Arrays

ConTEXt provides `mathmatrix` series of macros (`\definemathmatrix`, `\setupmathmatrix`, `\startmathmatrix`, and `\stopmathmatrix`) to take care of matrix alignments. These macros can provide functionality of `array` environment as well as the `matrix` series of macros from `amsmath` package.

Simple Matrix
A matrix is a collection of objects that are arranged in rows and columns. In LaTEX this alignment is provided by the `array` environment. In ConTEXt `\startmathmatrix` provides this feature.

In LaTEX

```
\begin{equation*}
\setlength{\arraycolsep}{1em}
  \begin{array}{ccc}
    A   &   B & C  \\
    AA  &  BB & CC \\
    AAA & BBB & CCC
  \end{array}
\end{equation*}
```

$$\begin{matrix} A & B & C \\ AA & BB & CC \\ AAA & BBB & CCC \end{matrix}$$

In ConTEXt

```
\startformula
  \startmathmatrix[n=3]
    \NC A   \NC B   \NC C   \NR
    \NC AA  \NC BB  \NC CC  \NR
    \NC AAA \NC BBB \NC CCC \NR
  \stopmathmatrix
\stopformula
```

$$\begin{matrix} A & B & C \\ AA & BB & CC \\ AAA & BBB & CCC \end{matrix}$$

In LaTEX the alignment of each column can be changed by the `r,c,l` options to `array`. In ConTEXt you need to pass appropriate arguments to `align=...`.
In LaTEX

```
\begin{equation*}
\setlength{\arraycolsep}{1em}
  \begin{array}{lcr}
    A   &   B & C  \\
    AA  &  BB & CC \\
    AAA & BBB & CCC
  \end{array}
\end{equation*}
```

$$\begin{array}{lcr} A & B & C \\ AA & BB & CC \\ AAA & BBB & CCC \end{array}$$

In ConTEXt

```
\startformula
  \startmathmatrix[n=3,align={left,middle,right}]
    \NC A   \NC B   \NC C   \NR
    \NC AA  \NC BB  \NC CC  \NR
    \NC AAA \NC BBB \NC CCC \NR
  \stopmathmatrix
\stopformula
```

$$\begin{matrix} A & B & C \\ AA & BB & CC \\ AAA & BBB & CCC \end{matrix}$$

pmatrix, et. al
The `amsmath` package provides `pmatrix`, `bmatrix`, etc. environments that make it easy to typeset matrix surrounded by delimiters. In ConTEXt it is straightforward to define such matrices uses `\definemathmatrix`
In LaTEX

```
\begin{equation*}
  A = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}
\end{equation*}
```

$$A = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

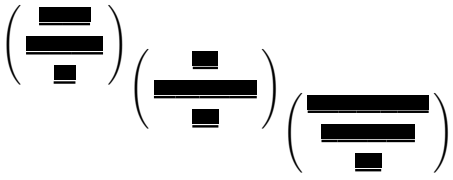
In ConTEXt

```
\definemathmatrix
  [pmatrix]
  [left={\left(\,},right={\,\right)}]

\startformula
  A = \startpmatrix 1 \NR 2 \NR 3 \NR \stoppmatrix
\stopformula
```

$$A = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

delarray package



The `delarray` package in LaTEX allows you to typeset arrays with properly scaled delimiters, even when the array is not middle aligned to the baseline. In ConTEXt the `\startmathmatrix` takes care of proper scaling of delimiters.
In LaTEX

```
\begin{equation*}
  \begin{array}[b]({c}) 1 \\ 2 \\ 3 \end{array}
  \begin{array}[c]({c}) 1 \\ 2 \\ 3 \end{array}
  \begin{array}[t]({c}) 1 \\ 2 \\ 3 \end{array}
\end{equation*}
```

$$\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

In ConTEXt

```
\definemathmatrix
  [pmatrix]
  [left={\left(\,},right={\,\right)}]

\startformula
  \startpmatrix[location=low]    1 \NR 2 \NR 3 \NR \stoppmatrix
  \startpmatrix[location=middle] 1 \NR 2 \NR 3 \NR \stoppmatrix
  \startpmatrix[location=high]   1 \NR 2 \NR 3 \NR \stoppmatrix
\stopformula
```

$$\binom{1}{2}_{\binom{1}{2}_{\binom{1}{2}}}$$

Cases

Cases is another common math alignment.

$$\blacksquare\blacksquare\blacksquare = \begin{cases} \blacksquare\blacksquare + \blacksquare\blacksquare, & \blacksquare\blacksquare\blacksquare\blacksquare\blacksquare\blacksquare\blacksquare \\ \blacksquare\blacksquare + \blacksquare\blacksquare + \blacksquare\blacksquare, & \blacksquare\blacksquare\blacksquare\blacksquare\blacksquare\blacksquare \end{cases}$$

The amsmath package provides a cases environment to build such alignments. ConTEXt provides \startmathcases.
In LaTEX

```
\begin{equation*}
  |x| =
  \begin{cases}
     x, & \text{if $x \ge 0$;} \\
    -x, & \text{otherwise.}
  \end{cases}
\end{equation*}
```

$$|x| = \begin{cases} x, & \text{if } x \geqslant 0; \\ -x, & \text{otherwise.} \end{cases}$$

In ConTEXt

```
\startformula
  |x| =
  \startmathcases
    \NC  x, \NC if $x \ge 0$ ; \NR
    \NC -x, \NC otherwise.     \NR
  \stopmathcases
\stopformula
```

$$|x| = \begin{cases} x, & \text{if } x \geqslant 0 ; \\ -x, & \text{otherwise.} \end{cases}$$

In the cases environment, the rows are set in textstyle. The mathtools package package provides a dcases environment to set the rows in displaystyle. In ConTEXt you can set the rows in displaystyle by passing style=\displaystyle to \startmathcases (or defining a new cases structure using \definemathcases).
In LaTEX

```
\begin{equation*}
  f(x) =
  \begin{dcases}
    \int_0^x g(y)\,dy,    & \text{if $x \ge 0$;} \\
    \int_{-x}^0 g(y)\,dy, & \text{otherwise.}
  \end{dcases}
\end{equation*}
```

$$f(x) = \begin{cases} \int_0^x g(y)\,dy, & \text{if } x \geqslant 0; \\ \int_{-x}^0 g(y)\,dy, & \text{otherwise.} \end{cases}$$

In ConTEXt

```
\startformula
  f(x) =
  \startmathcases[style=\displaystyle]
    \NC \int_0^x g(y)\,dy,    \NC if $x \ge 0$; \NR
    \NC \int_{-x}^0 g(y)\,dy, \NC otherwise.    \NR
  \stopmathcases
\stopformula
```

$$f(x) = \begin{cases} \int_0^x g(y)\,dy, & \text{if } x \geqslant 0; \\ \int_{-x}^0 g(y)\,dy, & \text{otherwise.} \end{cases}$$

Predefined Alignments

ConTEXt already has

```
\definemathalignment[align]
\definemathmatrix[matrix]
\definemantcases[cases]
```

defined. This means that in all the above examples, you can shorten `\startmathalignment ...\stopmathalignment` to `\startalign ...\stopalign`, `\startmathmatrix ...\stopmathmatrix` to `\startmatrix ...\stopmatrix`, and `\startmathcases ...\stopmathcases` to `\startalign ...\stopalign`.

Conclusion

ConTEXt now provides macros for math alignments. This makes it easier for the users to write complicated math alignments in ConTEXt. The syntax is consistent with the rest of ConTEXt macros, and thereby different from `amsmath` package syntax. Hopefully, this article will help eliminate the myth that ConTEXt is not able to handle complicated math. In ConTEXt features are added on user requests; so if there is something that you need which is not present in ConTEXt, ask for a feature request on the mailing list.

Notes

1. Herbert Voß, "*Math mode,*" available from http://tug.ctan.org/cgi-bin/getFile.py?fn=/info/math/voss/mathmode/Mathmode.pdf
2. Aditya Mahajan, "*My Way on* `\startalign` *and friends,*" available from http://dl.contextgarden.net/myway/mathalign.pdf
3. Compare these solutions from Exercise 22.9 in the TEXbook.

Aditya Mahajan
adityam@umich.edu