PlainTeX Verbatim

Benjamin T. Shepard

July 1, 2022

An inline verbatim such as `this is \inline` can be made with the code `|...|`. If visible spaces are desired, use `@|...|` instead, which will produce `this␣is␣\inline`. A display verbatim such as

```
this is \display
```

can be made with `||...||`. If visible spaces are desired, use `@||...||` instead, which will produce

```
this␣is␣\display.
```

Every normal TeX character (except `|`) as well as all of the special characters can be used inside this environment, as shown in the following example:

```
!$@$#%^&*()-+=\ /.,;'`~th@<is$ ]\bye *is{ #a ''1\rho > *\tes<t* \undefined .??["
```

Normally, the pipe character `|` can only be used inside the display verbatim `||...||`. However, the macros `\makepipeother` and `\makepipeactive` turn the character `|` into a normal character and back into an active character, respectively. This allows for use of the pipe literal in horizontal mode. There is also the macro `\pipe` which expands to

```
 macro: -> \bgroup\string|\egroup
```

displaying the literal `|`. These macros can be used to, for example, execute

```
 {\tt th\pipe{}s is a pipe}
```

or

```
                  \makepipeother{\tt th\pipe{}s is a pipe}\makepipeactive
```

which will each display `th|s is a pipe`. Of course, these macros cannot be used inside the `|...|` or `||...||` environments, as they will be treated as strings of catcode 12 characters.

The character `@` is able to be used regularly, and can also be used at the end of lines. Code such as

```
1 this is a t@st
2 @
3
4 this is a test@
5
6 \bye
```

will compile as expected. These verbatim environments also obey spaces, lines, and blank lines.