

Hans Hagen & Taco Hoekwater

Scary space

In pure T_EX

```
\show\  
\show\ %
```

gives the following log on Hans's machine

```
> \  
=macro:  
->\ .  
1.1 \show\  
  
?  
> \ =\ .  
1.2 \show \  
      %  
  
?  
)  
No pages of output.
```

and on Taco's machine gives:

```
> \^^M=macro:  
->\ .  
1.1 \show\  
  
?  
> \ =\ .  
1.2 \show \  
      %  
  
?  
)  
No pages of output.
```

The visualization of a $\^^M$ depends of the platform but since there's definitely a newline involved we need to take care of it. When parsing the input the following happens (this is mentioned in one of the dangerous bends in the T_EXbook):

```
\let\x\ <newline> => \let\x\<endlinechar>
```

This means that when you want to store the meaning of this primitive, you need to make sure that T_EX explicitly sees a space instead of a newline. So we get:

```
\let\normalspaceprimitive=\ % space-comment is really needed
```

In ConT_EXt this is used for:

```
\unexpanded\def\ {\mathortext\normalspaceprimitive{\dontleavehmode\space}}
```

If you don't use the explicit space a simple $\$ \ $$ will execute $\^^M$. In Plain T_EX (and in ConT_EXt) we have:

```
\def\^^M{ } % control <return> = control <space>
```

So this will result in a loop.