
Subject: IDL code for TeX to IDL strings
Posted by [flatau](#) on Sat, 22 May 1993 18:39:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

Sender:
Followup-To:
Distribution:
Organization: Colorado State University, Fort Collins, CO 80523
Keywords:

Here is a simple code and test batch file for conversion of TeX strings to (archaic) IDL strings. It is very preliminary. If you improve on this code let me know.

pflatau@macao.ucsd.edu

Peter

----- cut here execute with @ -----

```
set_plot,'tek'  
.rnew bbb  
a=findgen(10)  
plot,a,/nodata,xstyle=4,ystyle=4  
xyouts,1,9,texstr('Test of TeX strings - \alpha \beta \gamma')  
xyouts,1,8,texstr('\beta_i!N(\theta) ')  
xyouts,1,7,texstr('\beta_\alpha^!U(\theta) - you can mix IDL strings')  
xyouts,1,6,texstr('NO FRACTIONS just i^j_k sub and superscripts')  
xyouts,1,5,texstr("")  
xyouts,1,4,texstr('\alpha \beta \gamma \delta \epsilon \zeta \eta \theta \iota \kappa \lambda \mu \nu  
\oo \pi \rho \sigma \tau \upsilon \phi \chi \psi \omega')  
xyouts,1,3,texstr(' ')  
xyouts,1,2,texstr('^^^___ Hello ')
```

----- cut here -----

```
function texstr,cin
```

```
; replaces TeX strings with IDL strings  
; written by P. J. Flatau and Ferhat F. Hatay May 20, 1993  
; you can mix TeX and IDL strings  
; very preliminary and incomplete  
; works with postscript fonts  
  
;  
; Problems: (just the beginning)  
; () \OO and \oo are used instead of \O \o.  
; () Don't use ~ in your string or it will be replaced by blank  
; () parsing picks \omega for \o, \Omega for \O.
```

; () font selection might be done more elegantly
 ; () it does not work with font keyword in xyouts but with !p.font.
 ; () xyouts,strrep('Ferhat \Omega'),font=0 ; will not work
 ; () !p.font=0 & xyouts,strrep('Ferhat \Omega'),font=0 ; will
 ; () problem with \x on Silicon Graphics (IDL bug ?)
 ; () no {} support at all
 ; () no mathematical symbols (but some of this can be easily added)

ctemp=cin

cary=[' ','_','^',\$
 '\alpha','\beta','\gamma','\delta',\$
 '\epsilon','\zeta','\eta','\theta',\$
 '\iota','\kappa','\lambda','\mu',\$
 '\nu','\oo','\pi','\rho',\$
 '\sigma','\tau','\upsilon','\phi',\$
 '\chi','\psi','\omega',\$

'\Alpha','\Beta','\Gamma','\Delta',\$
 '\Epsilon','\Zeta','\Eta','\Theta',\$
 '\Iota','\Kappa','\Lambda','\Mu',\$
 '\Nu','\OO','\Pi','\Rho',\$
 '\Sigma','\Tau','\Upsilon','\Phi',\$
 '\Chi','\Psi','\Omega'\$
]

crep=['~!N','!D','!U',\$
 'a','b','c','d',\$
 'e','f','g','h',\$
 'i','j','k','l',\$
 'm','o','p','q',\$
 'r','s','t','u',\$
 'v','w','x',\$
 'A','B','C','D',\$
 'E','F','G','H',\$
 'I','J','K','L',\$
 'M','O','P','Q',\$
 'R','S','T','U',\$
 'V','W','X'\$
]

creps=['~!N','!D','!U',\$
 'a','b','g','d',\$
 'e','z','h','q',\$
 'i','k','l','m',\$
 'n','o','p','r',\$
 's','t','u','f',\$
 'c','y','w',\$
 'A','B','G','D',\$
 'E','Z','H','Q',\$

```
'I','K','L','M',$
'N','O','P','R',$
'S','T','U','F',$
'C','Y','W'$
]
```

```
n = n_elements(cary)
```

```
;Hardware fonts with Complex and Complex Symbols.
```

```
;
```

```
sfont='!7'
```

```
rfont='!3'
```

```
; soft font selection with Times-Roman and Symbols.
```

```
;
```

```
if(!p.font eq 0 ) then begin
```

```
    sfont='!9'
```

```
    rfont='!7'
```

```
    crep=creps
```

```
endif
```

```
; Anyother soft-font goes with Symbols
```

```
;
```

```
if(!p.font gt 0 ) then begin
```

```
    sfont='!9'
```

```
    crep=creps
```

```
    rfont='!'+strcompress(!p.font,/remove_all)
```

```
endif
```

```
cin = rfont+cin
```

```
mark=1
```

```
while (mark ne -1) do begin    ;loop until everything is replaced
```

```
mark=-1
```

```
for i=0,n-1 do begin
```

```
    ipos=strpos(cin,cary(i))
```

```
    mark=max([mark,ipos])
```

```
    if(ipos ne -1) then begin
```

```
        ca=strmid(cin,0,strpos(cin,cary(i)))
```

```
        cb=strmid(cin,strpos(cin,cary(i))+strlen(cary(i)),strlen(cin ))
```

```
        cin=ca+sfont+crep(i)+rfont+cb
```

```
    endif
```

```
endfor
```

```
endwhile
```

```
while (((i=strpos(cin,'~')) ne -1) do strput,cin,' ',i ;replace ~ with blank
```

```
cout=cin
```

```
cin=ctemp
```

```
return,cout
```

end
