
Subject: Re: A routine to annotate PS files

Posted by [MarioIncandenza](#) on Wed, 24 Feb 2010 19:04:44 GMT

[View Forum Message](#) <> [Reply to Message](#)

Here is an updated version of the routine. Changes include:

- Now writes entire call stack, including \$MAIN\$ (\$MAIN\$ has a program file associated with it, and often that is information you need, it's now included);
- Includes a /QUIET option to suppress echoing of the tags to be written;
- Includes a (Unix-only, or maybe even more limited) option to /PARSE_FINGER, which calls 'finger `whoami`' and writes the 'Login:' and 'Name:' tags into the PS file. If it doesn't work on your system, just ignore it. It won't cause any problems.

```
-----
pro
  annotate_psfile,extra_tags=extra_tags,parse_finger=parse_finger,quiet=quiet
  tagf='("%% " ,a," : ",a)'      ; %% Tag : Value
  trace=scope_traceback(/structure)
  ntrace=n_elements(trace)
  geni=ntrace-2                  ; index to proximate calling script
  tag0=string('GenRoutine',trace[geni].routine,format=tagf)
  tag1=string('GenFile',trace[geni].filename,format=tagf)
  tags=[tag0,tag1]
  if(geni gt 0) then begin
    for icall=0,geni-1 do begin
      tag0=string('CallRoutine',trace[icall].routine,format=tagf)
      tag1=string('CallFile',trace[icall].filename,format=tagf)
      tags=[tags,tag0,tag1]
    endfor
  endif
  timecode=string(julday(),format='(C(CYI4.4,CMOI2.2,CDI2.2,CH I2.2,CMI2.2))')
  tag0=string('TimeCode: ',timecode,format=tagf)
  tags=[tags,tag0]
  if n_elements(extra_tags) ne 0 then begin
    extranames=tag_names(extra_tags)
    for iextra=0,n_elements(extranames)-1 do begin
      tag0=string(extranames[iextra],extra_tags.
(iextra),format=tagf)
      tags=[tags,tag0]
    endfor
  endif
  if n_elements(parse_finger) ne 0 then begin
    spawn,'finger `whoami`',result,error
    if(error eq '') then begin
      fingertags=stregex(result[0],'Login:(.*)Name:(.*)',/subexpr, /
extract)
```

```
    tag0=string("Login: ",strtrim(fingertags[1],2),format=tagf)
    tag1=string("Name: ",strtrim(fingertags[2],2),format=tagf)
    tags=[tags,tag0,tag1]
endif
endif

if n_elements(quiet) eq 0 then print,tags,format='(A)'
ntags=n_elements(tags)
for itag=0,ntags-1 do device,output=tags[itag]
return
end
```
