TURBOCAD VERSION 15.2 and the new parametric part scripting

TurboCAD has several programmable options. One uses VBS (Visual Basic Script) and has been covered in another Illustrating Shadows booklet, as well as in Illustrating More Shadows. The latest version of TurboCAD, version 15.2 (not 15.1 nor earlier), supports Parametric Script entities. In essence, a script is written that inserts an entity into a drawing, and from then on, that entity may be selected and its parameters may be modified to generate a whole new layout.

The Parametric Script language is interesting in that the "Output" statement is executed last, and other statements wherever they may be, are executed first. Case is critical, thus "output" is not executable, "Output" is the correct spelling. Trigonometric functions use degrees, however there is no INT nor ABS provided.

Some documents are important to have. As happens with IMSI, documentation is not made obvious. Searches of the internet are essential for clues to a new feature, and the parametric scripting feature is no exception.

Essential notes on parametric script building:

This explains how the editor marks syntax errors:-

http://www3.turbocadcommunity.com/tiki-index.php?page=PPM+Scripting
http://www3.turbocadcommunity.com/tiki-index.php?page=PPM+Scripting+Reference

Useful but incomplete notes on parametric scripts:-

http://downloads.imsidesign.com/HelpFiles/Scripting%20Parametric%20Parts%20Feb%2029,%202008.pdf

It is only through the above two documents that scripts can easily be written. The Parametric Script Editor is invoked by "VIEW" and then "Parametric Script Editor Pallette", and the script can be easily typed, or loaded from a file in any folder.

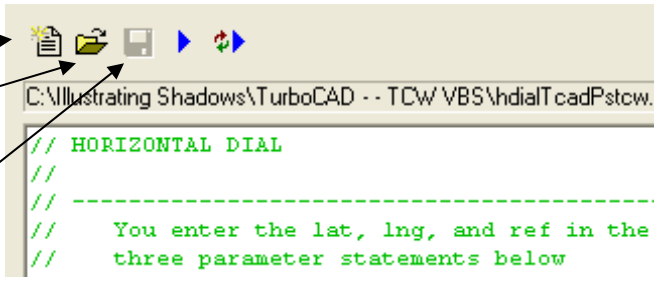
The Editor uses color coding:

black	variables
brown	text
blue	reserved words
magenta	is an error
	The status line of the palette indicates the error reason for the error.
green	comments

write a new program

use this folder to open a program

save this program (script)



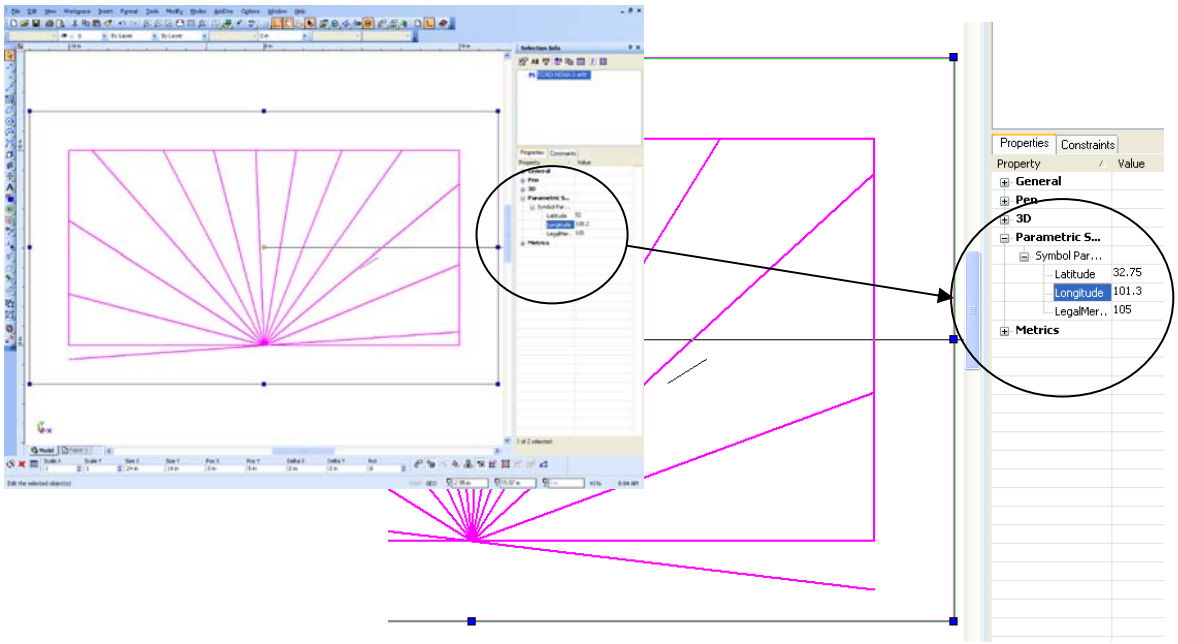
"Output(plate);" has "plate" in magenta and thus says "plate" is wrong. Plate has a syntax error in that two ")" are missing.

```
        Point(0,0) ,  Point(-xyl*tan(hl5),xyl)     ,  Point(0,0),
        Point(0,0) ,  Point(xyl, -xyl*tan(yl6)     ,  Point(0,0),
        Point(0,0) ,  Point(xyl, -xyl*tan(yl7)     ,  Point(0,0),
//      now the boundary box
        Point(-xyl,0) ,
        Point(-xyl,xyl) ,
        Point(xyl,xyl),
        Point(xyl,0),
        Point(0,0)
        );
//
Output (plate);
//
```

Once corrected, then "Output(plate);" will show in normal text.

```
        Point(0,0) ,  Point(-xyl*tan(hl5),xyl)     ,  Point(0,0),
        Point(0,0) ,  Point(xyl, -xyl*tan(yl6))    ,  Point(0,0),
        Point(0,0) ,  Point(xyl, -xyl*tan(yl7))    ,  Point(0,0),
//      now the boundary box
        Point(-xyl,0) ,
        Point(-xyl,xyl) ,
        Point(xyl,xyl),
        Point(xyl,0),
        Point(0,0)
        );
//
Output (plate);
//
```

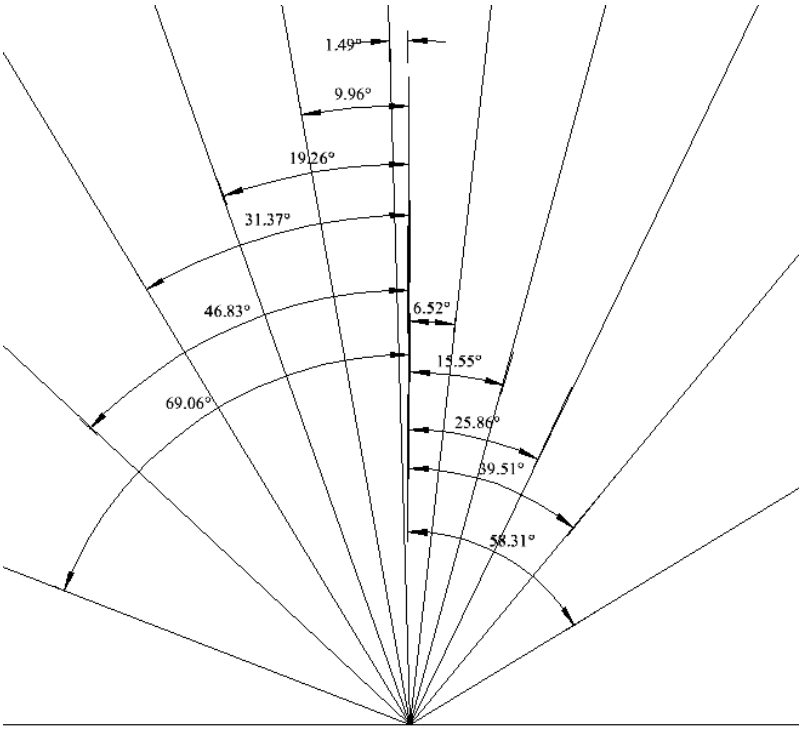
A fabulous feature of these script created objects is that once a parametric part has been built on the drawing area, it can be selected (as with any entity) and then VIEW, SELECTION INFO will enable you to see and change those parameters (set with the "Parameter" statement).

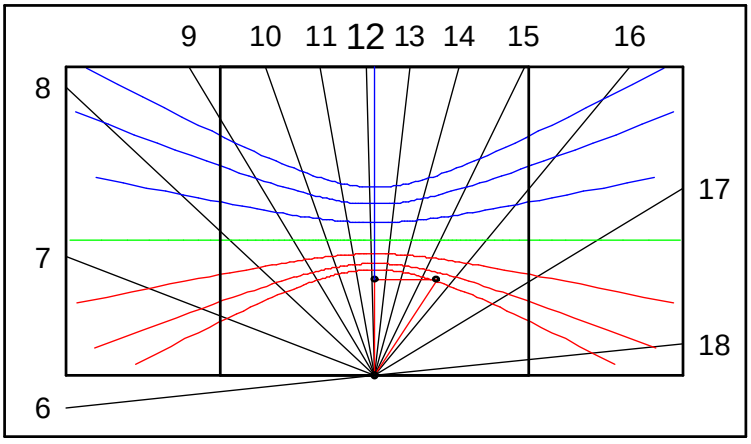


The parameters were changed from lat 52, long 108.2 and 105 to a new set of 32.75, 101.3 and 105. And the entire drawing redrew itself.

Typical output from a script, and accuracy

The TurboCAD Parametric Script included here generated the following dial layout. The hour line angles were measured with INSERT, DIMENSION, ANGULAR, then right click and select ANGLE NODE and then follow the prompts.





h-dial and calendar using gnomon linear height

Lat: 32.8 d.Long: 03.2

6	7	8	9	10	11	12	13	14	15	16	17	18
84.1	-68.9	-47.0	-31.2	-19.5	-10.1	-1.7	06.4	15.3	25.8	39.6	58.7	84.1

Hours below horizontal use the 90 reference line below horizontal.

The angles are very close to the same dial plate's hour line angles produced by another CAD system, DeltaCAD.

The Parametric Script (program)

```
// HORIZONTAL DIAL  ~  This seems to test correctly, but use as
//                     a basis for your own code. Not guaranteed.
//                  ~  TCAD-hDial-3-adv.ppm       [advanced h-dial]
// www.illustratingshadows.com   Simon Wheaton-Smith  Dec 10, 2008
//
// ---------------------------------------------------------------------
// You enter lat, lng, and ref in the three parameter statements below
//
lat = Parameter("Latitude",  32.75, ANGULAR, Interval(20,80));
lng = Parameter("Longitude", 108.2, ANGULAR, Interval(0,360));
ref = Parameter("LegalStd",  105.0, ANGULAR, Interval(0,360));
//
//   you change the values of latitude, longitude, and reference
//   longitude above
// ---------------------------------------------------------------

//   state the vertical and horizontal limit size
xyl  = 10;
//   and the multiplier for the outer border box
scl  = 0.2;
scz  = 1+scl;

//    derive the hours of correction for hour lines
dif  = 4*(lng - ref)/60;

// define the hours
y06  = atan(sin(lat)*tan(15*(6+dif)));
ha07 = atan(sin(lat)*tan(15*(5+dif)));
   x07  = IF( (ha07)>=45, -xyl,               -xyl*tan(ha07) );
   y07  = IF( (ha07)>=45,  xyl*tan(90-ha07),   xyl           );
ha08 = atan(sin(lat)*tan(15*(4+dif)));
   x08  = IF( (ha08)>=45, -xyl,               -xyl*tan(ha08) );
   y08  = IF( (ha08)>=45,  xyl*tan(90-ha08),   xyl           );
ha09 = atan(sin(lat)*tan(15*(3+dif)));
   x09  = IF( (ha09)>=45, -xyl,               -xyl*tan(ha09) );
   y09  = IF( (ha09)>=45,  xyl*tan(90-ha09),   xyl           );
x10 = atan(sin(lat)*tan(15* ( 2+dif)));
x11 = atan(sin(lat)*tan(15* ( 1+dif)));
x12 = atan(sin(lat)*tan(15* ( 0+dif)));
x13 = atan(sin(lat)*tan(15* (-1+dif)));
x14 = atan(sin(lat)*tan(15* (-2+dif)));
ha15 = -atan(sin(lat)*tan(15*(-3+dif))) ;
   x15  = IF( (ha15)>=45,  xyl,                xyl*tan(ha15) );
   y15  = IF( (ha15)>=45,  xyl*tan(90-ha15),   xyl           );
ha16 = -atan(sin(lat)*tan(15*(-4+dif))) ;
   x16  = IF( (ha16)>=45,  xyl,                xyl*tan(ha16) );
   y16  = IF( (ha16)>=45,  xyl*tan(90-ha16),   xyl           );
ha17 = -atan(sin(lat)*tan(15*(-5+dif))) ;
   x17  = IF( (ha17)>=45,  xyl,                xyl*tan(ha17) );
   y17  = IF( (ha17)>=45,  xyl*tan(90-ha17),   xyl           );
y18 = atan(sin(lat)*tan(15* (-6+dif)));

//    display the dial plate
plate=Polyline(
      Point(0,0) ,  Point(-xyl, xyl*tan(90-y06)),   // 0600
      Point(0,0) ,  Point( x07, y07)            ,   // 0700
      Point(0,0) ,  Point( x08, y08)            ,   // 0800
      Point(0,0) ,  Point( x09, y09)            ,   // 0900
      Point(0,0) ,  Point(-xyl*tan(x10),xyl)    ,   // 1000
      Point(0,0) ,  Point(-xyl*tan(x11),xyl)    ,   // 1100
      Point(0,0) ,  Point(-xyl*tan(x12),xyl)    ,   // noon
      Point(0,0) ,  Point(-xyl*tan(x13),xyl)    ,   // 1300
      Point(0,0) ,  Point(-xyl*tan(x14),xyl)    ,   // 1400
      Point(0,0) ,  Point( x15, y15)            ,   // 1500
```

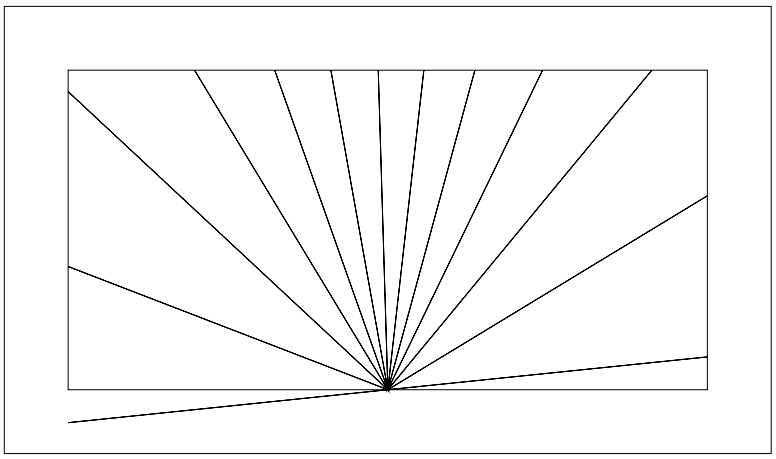
```
      Point(0,0) ,  Point( x16, y16)            ,   // 1600
      Point(0,0) ,  Point( x17, y17)            ,   // 1700
      Point(0,0) ,  Point(xyl, -xyl*tan(90-y18)),   // 1800
//    now the inner boundary box
      Point(0,0) ,
      Point(-xyl,0)   ,
      Point(-xyl,xyl) ,
      Point(xyl,xyl)  ,
      Point(xyl,0)    ,
// back at 0,0
      Point(0,0)
      );
//
box=Polyline(
//    now the outer boundary box
      Point( 0,      -scl*xyl) ,
      Point(-scz*xyl,-scl*xyl) ,
      Point(-scz*xyl, scz*xyl) ,
      Point(scz*xyl,  scz*xyl) ,
      Point(scz*xyl, -scl*xyl) ,
      Point( 0,      -scl*xyl) );

//Output (plate,box);
Output (plate);
Output (box);
//
```

There are several problems with the above script, one is that it does not handle shifting from a fixed x or y-value with a generated x or y-value, except for 0700-0800 and 1500-1700 hours.

Since few diallists use TurboCAD, and fewer still use the TurboCAD Professional version, the effort to go to those extremes was not made.

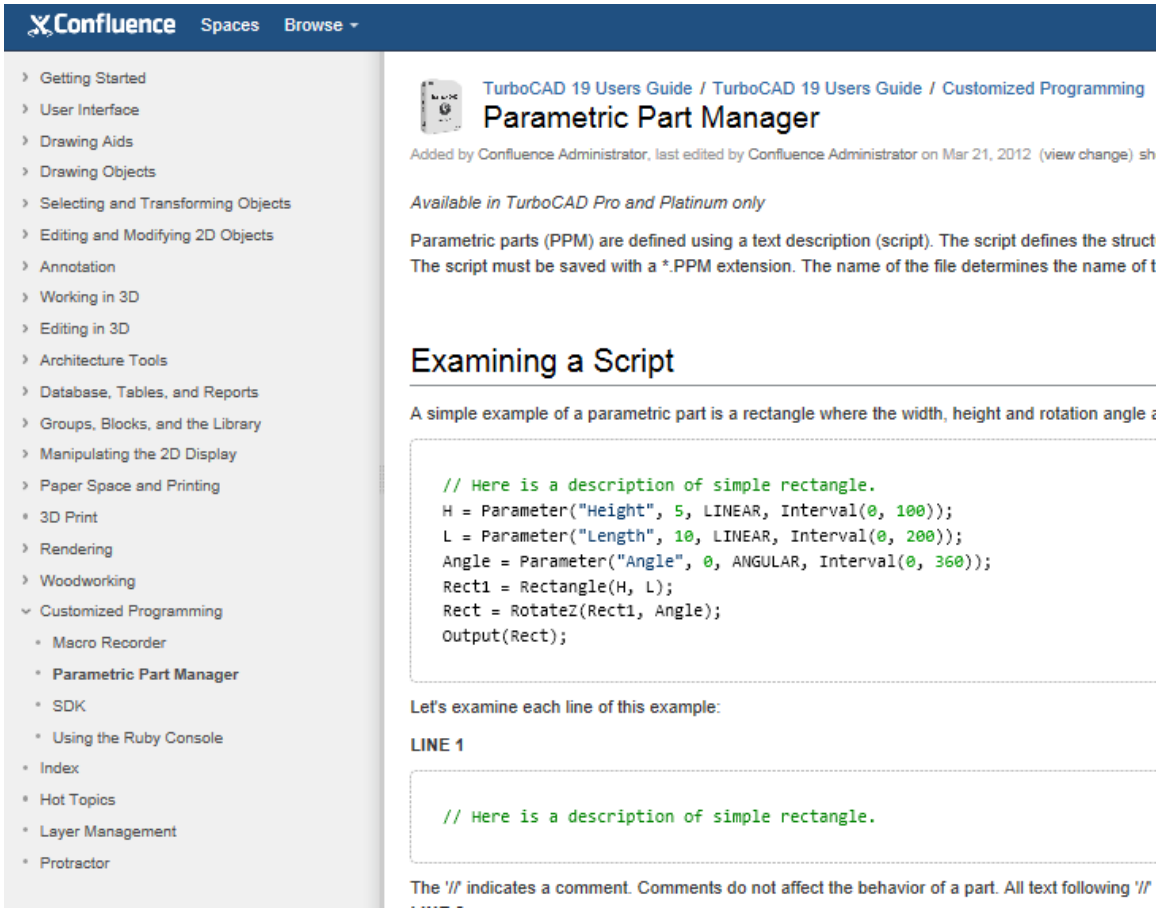
The final display of the above program is shown to the right.



NOTE: If you save a TCW file with an entity created with these scripts, and later open it, you can modify the parametric parms for the object and it will redisplay itself This has the potential for being a significant benefit beyond other CAD systems.

ADDITIONAL NOTES ~ March 2013

https://turbocaddoc.atlassian.net/wiki/display/TC19UG/Parametric+Part+Manager



the web page partially shown above has a discussion on script writing for parametric parts.