

J8 の gl2 グラフィックスへの移行

SHIMURA Masato

2018 年 6 月 13 日

目次

1	ラヌクロイドを題材とする J8QT グラフィックスへの移行方法	1
2	C.Reiter の dwin	2
3	ラヌクロイドを QT 版の gl2 で描く	3
4	窓とボタンの操作	5

1 ラヌクロイドを題材とする J8QT グラフィックスへの移行方法

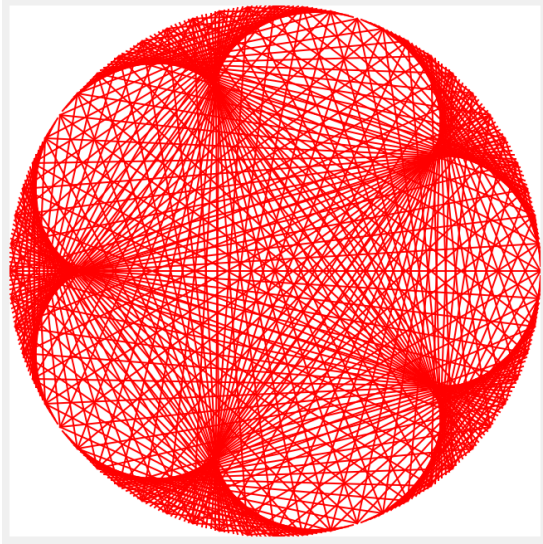
1. 出典: 西川利男 シンポジウム 2003
2. 論文: JWindowsGraphics への誘い: J で綺麗なグラフィックスを描こう

J8 はフォームエディターのサポートをやめ、更に gl2 ネイティブのグラフィックスはデリケートだ。Qt はスマートフォンをカバーするグラフィックスなので、フォームデザインもスマートフォン向けになっている。

フォームエディターを使って作成したグラフィックスの例を選び Qt 版向けのフォームを作ってみる。

例題として西川:2003 年のシンポジウムで発表されたラヌクロイドのグラフィックスを用いる。

ラヌクロイドの計算スクリプト。y には 1-360 をループで与える



```

va=: 6
vb=: 6

chord =: 4 : 0
'a b' =. x
deg =. 1r180p1
t0 =. (2, 1)&o. deg * y
t1 =. (2, 1)&o. (va, vb) * deg * y
t0, t1
)

```

2 C.Reiter の dwin

そこで最初はフォームを全く使わない C.Reiter の addons/graphics/fvj4 に入っている dwin.ijs で描いてみよう

- フォーム不要
- 起点 (0,0) は左下
- マイナス、実数を受け付ける
- キャンバスの大きさは起動時に指定
- 自動アジャスト。dwin に左下、右上の範囲を指定することで、計算結果を画面一杯に繁殖できる
- 読み込みデータは縦型
- 描画関数は独自

```
6 6 ngraph_main ''
```

a は円を分割する個数 (n - 1)

b は a と揃えないと破調になる

```
ngraph_main=: 4 : 0
```

```

NB. Usage u 32
'a b'=. x
_1 _1 1 1 dwin 'ngraph'
  for_ctr. i. 360 do.
    Dat=: (a,b) chord ctr
    255 0 0 dline Dat
  end.
)

WIN_WH=: 640 640

```

3 ラヌクロイドを QT 版の gl2 で描く

Qt 版のグラフィックスは、スマートフォンの画面とボタンを整然と並べる機能を従来の gl2 グラフィックスの中に取り入れたものである。

- gl2 は慎重にフォームをデザインしなければならない。
- 3 点セットの作成例
 1. Graph 名前は自由。キャンバスと窓、ボタンの配置
 2. paint 引数は取りにくい
 3. run 定型
- 窓やボタンはフリーデザインでは無い

兎に角動くコードを下敷きにして組み立てていくのが早計である。

- フォームの設計

```

GRAPH=: 0 : 0
pc graph closeok;
minwh 500 500;cc ranunc isigraph;
bin vh;
cc label static;cn "Enter: a";
cc a edit ;
bin s 1;
cc label static;cn "Enter: b";
cc b edit ;
bin s 1;
cc set button;cn "&Set";
cc redraw button;cn "&ReDraw";
cc close button;cn "&Close";
bin z;
pas 0 0;
rem form end;
)

```

- run :フォームとチャイルドを指定する

```

run=:graph_run=: 3 : 0
wd GRAPH
gsel ranunc      NB. OK
wd 'pshow'
)

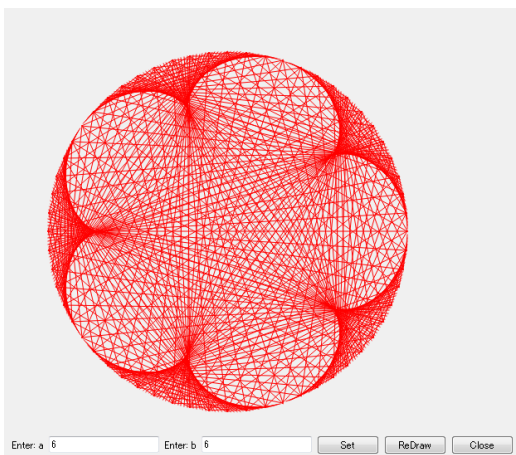
```

- paint: ここでループをいれ一筆ずつ描く

```

graph_ranunc_paint=: 3 : 0
for_ctr. i. 360 do.
  glrgb 255 0 0
  glpen 1 1
  glbrush ''      NB. red brush
  gllines >.@adj (va,vb) chord ctr
end.
)

```



画面はマイナスや実数を受け付けないので、計算結果を画面サイズに適合させ整数化している

```
gllines >.@adj (va,vb) chord ctr
```

4 窓とボタンの操作

1. 入力窓 (edit): a,b 二つの入力窓を edit で作成し、見出しをつける。手入力

```
cc label static;cn "Enter: a";  
cc a edit ;  
bin s 1;  
cc label static;cn "Enter: b";  
cc b edit ;  
bin s 1;
```

2. 初期値のセット : 数値は text を付けると文字化してセット出来る

```
cc label static;cn "Enter: a"; NB. 見出し  
cc a edit ; NB. edit を a と名付ける  
set a text 6; NB. 6 を文字化して入力
```

3. 入力された数値の取り込み。Set ボタン

取り込んだ数値 (文字) は数値化している (最初の値は別途 va,vb で与えている)

```
graph_set_button=: 3 : 0  
va=: ". a  
vb=: ". b  
)
```

4. 再ドロー

run と同じ。前もって画面をクリアしておく

```
graph_redraw_button=: 3 : 0  
wd GRAPH  
gsel ranunc NB. OK
```

```
glclear''  
wd 'pshow  
)
```

NB. 画面のクリアではない。