

# 北斎の新形小紋帳から (J8 への移行)

SHIMURA Masato  
JCD02773@nifty.ne.jp

2015 年 8 月 17 日

## 目次

1	J8-QT 版の特色と癖	1
1.1	繊細な QT 版の gl2	2
1.2	paint の負荷テスト	3
2	北斎の移行	5
2.1	六つ手万字のデータ	5
2.2	J8 での修正	6
2.3	スクリプトの修正	6

### 概要

数年前葛飾北斎の「新型小紋帳」を J 言語 (J602) のグラフィックスを用いて数多く描き出した。J 言語は Ver.8 でグラフィックス環境に iPhone や MAC などで採用されている QT を全面的に採用した。同じ gl2 を用いた 2 次元グラフィックスでも J6 と J8 では微妙な差異があり、スクリプトはそのままでは通らない。J8 の QT 版グラフィックスの細部を確認しながら「新型小紋帳」の移植方法を試してみた。

## 1 J8-QT 版の特色と癖

J8XQT 版はグラフィックス環境に QT を採用し、gl2 をかなり強引に QT に対応させたが、QT 版 gl2 は非常に繊細で、J6 のスクリプトを完全には受け付けない。さらにスクリプトがエラーになるとシステムがよくハングアップする。

## 1.1 繊細な QT 版の gl2

### 1. 額を *HOKUSAI*, チャイルドを *wood* とする

- `hokusai_wood_paint` は `hokusai_run` に書き込めない。即ち、`hokusai_wood_paint` は
- `hokusai_run` の中に書き込むとエラーになる何処にも引用されない。
- `hokusai_wood_paint` 中のスクリプトがエラーになるとシステムがハングアップすることが多い。
- `paint` 中でスペルを間違えてもエラーのダイアログが出て、消すことが出来ず、*J* のシステムを終了する破目になる。

### 2. 引数

- 引数はダイレクトには取れない。
- `hokusai_wood_paint` の引用データはグローバル変数とし `paint` の外における。
- `paint` 関数の中で計算することはできる

### 3. 3点セットを書きだす

```
HOKUSAI=: 0 : 0
pc hokusai closeok;
minwh 500 500;cc wood isigraph
flush;
pas 0 0;
)

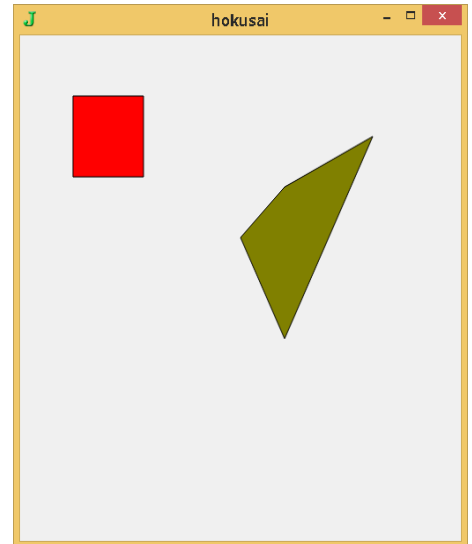
hokusai_run=: 3 : 0
wd HOKUSAI
wd'pshow'
)

hokusai_wood_paint=: 3 : 0
glrgb C0 NB. 128 128 0
glbrush ''
glpolygon temp2
glrgb C1 NB. 255 0 0
glbrush ''
glrect temp
)

run=:hokusai_run''
```

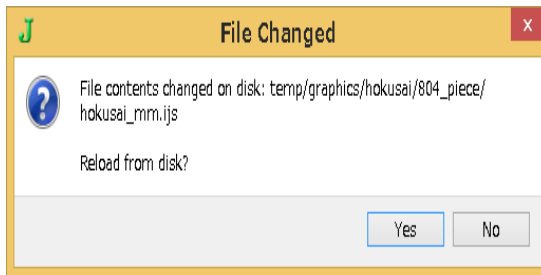
### 4. 3点セット外に置いた *paint* 用データ

```
temp1=: 100 100 150 200 300 350 450 400
temp2=: 250 300 300 300 200 100 300 150
'C0 C1 C2'=: 128 128 0;255 0 0;0 0 255
temp3=: 150 100 300 300 400 100 200 150
COLOR=: 128 128 0;255 0 0;0 0 255
```



5. *Edit* にも微妙な癖がある。これは改善されることを期待する

- *Edit* での変更が反映されないことがある。
- *Edit* に読み込んだスクリプトを別のエディタで変更すると次のダイアログが出て *Edit* に反映される。即ち、スクリプトの変更は外部エディタで行った方が効率的？



## 1.2 paint の負荷テスト

1. プロトタイプ。(通る)

```
hokusai_wood_paint=: 3 : 0
glrgb C0 NB. 128 128 0
glbrush ''
glpolygon tmp2
```

)

2. `glrgb,glbrush` を変えないで描いた図は同じ色で描ける。

```
hokusai_wood_paint=: 3 : 0
glrgb C0
glbrush ''
glpolygon temp
glpolygon temp2
)
```

3. ループは使える。3点セットでループに入れると色も変えられる。

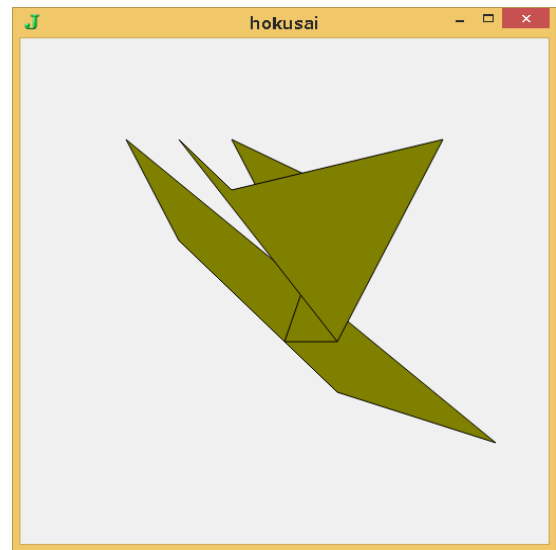
```
hokusai_wood_paint=: 3 : 0
tmp=: temp1;temp2;temp3
for_ctr. i. # tmp do.
  CLR=. >ctr{ COLOR
  tmp0=.> ctr{ tmp
  glrgb CLR
  glbrush ''
  gpolygon tmp0
end.
)
```

4. `L:0` による同時描画も可能。色は同一色だが小紋をループなしに描くことが出来る

```

hokusai_wood_paint=: 3 : 0
tmp=: temp1;temp2;temp3
  glrgb C0
  glbrush ''
  glpolygon L:0 tmp
)

```



## 2 北斎の移行

*J6* で北斎の新型小紋帳のパターンを描いたときは *C.Reiter* の *fvj2* に含まれていた *dwin2.ijs* の便利な機能を用いたが、*J8* ではまだサポートされていない。

- *dwin* ではグラフィックスの始点は左下を採用。*J* では *J6* の途中から右上に変更された。
- *dwin* では画面サイズは外付けで指定できた
- *dwin* では *paint* に外部からデータを渡すことが出来た。

### 2.1 六つ手万字のデータ

NB. No. 17

NB. 六つ手万字

NB. 6 fingers Gammadion

NB. 1st piece

MM0=:10 5, 8 4, 8 2, 6 1, 8 0, 10 1, 12 0, 12 2 ,:10 3

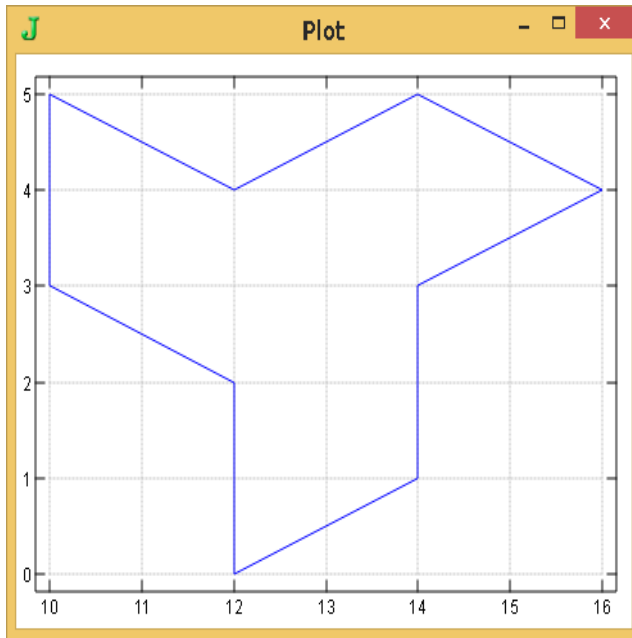
NB. 2nd piece

MM1=: 10 5,10 3,12 2,12 0,14 1,14 3,16 4,14 5,:12 4

MM0PARAM=: 10 5;6 \_1;\_2 5 NB. for automatic

MM1PARAM=: 10 5;6 \_1;\_2 5 NB. same MM0PARAM

- $MM0, MM1$  は六つ手のポリゴン
- $MMOPARAM, MMIPARAM$  は始点、右隣の始点との差分、上の始点との差分



## 2.2 J8 での修正

J8 では次の点を修正した

- J8 では小紋一枚に額縁一枚を付けることとし、スクリプトを小紋 1 枚毎に作成する。
- J8 の座標は絶対座標であり、 $minwh$  は額縁の外枠を広げるのみで、対象自体を拡大しない。J6 の  $dwin$  は写真はサイズを指定したら、顔はフレームに合わせて自動で拡大してくれるが、このような便利な機能はない。
- マイナスの座標は枠外になるのみ。

## 2.3 スクリプトの修正

### 1. 小紋の数とサイズの一括指定

- (a) J6 の小紋ピースの座標は細かいので数倍拡大しないと米粒になる。ピースの拡大は  $TIMES$  で書き出し一括指定する。
- (b) 小紋の数は  $SIZE$  で一括指定する。個数で枠のはみ出しを気にせず埋め尽くす。

```

NB. -----
SIZE=: 50 50
TIMES=: 12
NB. -----

```

2. 模様作成関数 `calc_each_poly` はそのまま用いて、模様毎に計算するツールを被せる。

```

calc_mm0=: 3 : 0
NB. Usage: calc_mm0 ''
SIZE calc_each_poly (< TIMES * MM0),< TIMES * L:0 MM0PARAM
)

```

```

calc_mm1=: 3 : 0
NB. Usage: calc_mm0 ''
SIZE calc_each_poly (<TIMES * MM1),<TIMES * L:0 MM1PARAM
)

```

3. 額縁のサイズはここで指定する。*J8* は *minwh* または *maxwh* で指定する

```

HOKUSAI=: 0 : 0
pc hokusai closeok;
minwh 200 200 ;cc wood isigraph flush;
pas 0 0;
)

```

```

hokusai_run=: 3 : 0
wd HOKUSAI
wd'pshow'
)

```

4. 小紋の一括描画。カラーはここで *RGB256* 階調で指定する

```

hokusai_wood_paint=: 3 : 0
tmp0=. , L:0 calc_mm0 ''
tmp1=. , L:0 calc_mm1 ''

```

NB. -----

```
glrgb 67 71 173
```

```
glbrush ''
```

```
glpolygon L:0 tmp0
```

```
glrgb 67 90 200
```

```
glbrush ''
```

```
glpolygon L:0 tmp1
```

```
)
```

```
run=: hokusai_run
```

