本日は宮崎君回

USB メモリの中身 20120704/ VisualWorks78nc.zip vw78jun793mac.zip vw78jun793win.zip

20120704/ の中身は ClickEarth.html ClickEarth.pdf ClickEarth.st ClickEarth_files

GUI シリーズ GUI パーツの扱い方をメインにやっていくので、あまりプログラムをがりがり書く感じではないとのこと

VisualWorks を起動 ClickEarth.html を見てみる。 今回やることの一通りの手順が書いてある

System Browser を開く。 KSU Package に Example と System だけがある事を確認

KSU Package に Example 2 System だけがめる事を维認 以下を workspace に入力して実行 I fileInBlock packageBlock I fileInBlock = [[aFilename I: JunFileRequesterDialog requestFilename. aFilename ifileIn] on: UserInterrupt do: [:anException I anException proceed]]. packageBlock := [aCollection := OrderedCollection new) add: #comment: -> 'Copyright 2008-2012 KSU (Kyoto Sangyo University). All Right Reserved.'; add: #bundle: -> #KSU; add: #package: -> 'KSU-Template'; add: #nameSpace: -> 'KSU-Template'; yourself. JunSystem perform: (aCollection collect: [:each I each key]) inject: String new into: [:selector :key I selector , key]) asSymbol withArguments: (aCollection collect: [:each I each value]) asArray]. fileInBlock value.

ClickEarth.st.st を選択すると読み込まれる。

KSU Package に KSU-Template が増える。 名前の通り、プログラムファイルのテンプレートが一式、入っている

Template の中の example1 を実行してみると 2つのウインドウが開く 白い方にいるいろと手を加えてプログラムを作成していく

○ ○ ○ 地球をクリック





windowSpec から Edit を押して GUI のデザインをやっていく 先ほど開いた画面と同じものが開かれここにデザインをしていく



その親玉

\$ \$ \$ \$ \$ \$ \$	
Main Window : 地球	Basics Details Position/Size Color Dr
	Window
	Label: Supplied by Application
	String: 地球をクリック
	Lookup key:
	Catalog
	Menu Bar
	Enable
	Menu: #menuBar
	Tool Bar
	Menu:
	Apply Cancel Prev Nex

パーツのパレット



この3つが開く

まずはメニューバーを追加

# <2 & # 5	
Tain Window : 地球	Basics Details Position/Size Color Dr
	Window
	Label: Supplied by Application
	String: 地球をクリック
	Lookup key:
	Catalog
	Menu Bar
	Menu: #menuBar
	Tool Bar
	Menu:
	L
	Apply Cancel Prev Nex

Menu Bar のチェックボックスをチェックして Apply するだけ! Menu: に #menuBar と既に入っているが普段から元々入っているわけではない(はず) よく見ると、 Menu に何か書いてあって resources の menuBar があるのでコレでヒモ付いたというだけ

Browser Edit Find View Package Class Protocol Method Tools Help Package Class H Base VisualWorks H StoreBase KSU-Example KSU-System KSU-System KSU-System KSU-Source H Sport Source Comment Definition Rewrite Code Critic Visual Source Comment Definition Rewrite Code Critic Edit Menu Image/Package: KSU-Template	● ● ● O ClickEarth clas	ss>>menuBar
Image: Class Package: Class Image: Class </td <td>Browser Edit Find View Package Class Protocol Method T</td> <td>ools Help</td>	Browser Edit Find View Package Class Protocol Method T	ools Help
Package Class Instance Class Shared Variable Instance Variable Base VisualWorks ImageOfEarth Interface specs example1 ImageOfEarth KSU-Example KSU-Template Interface specs ImageOfEarth ImageOfEarth KSU-Template KSU-Template ImageOfEarth ImageOfEarth ImageOfEarth Sast ImageOfEarth ImageOfEarth ImageOfEarth ImageOfEarth ImageOfEarth Sast Sast ImageOfEarth ImageOfEarth	💭 °H °O °+ "H °O "D 🎬 "O (" Find:	•
Method: #menuBar (resources) Package: KSU-Template	Package Class Base VisualWorks Clorp KSU-Example KSU-System KSU-Template Sort Code Critic Edit Menu	Instance Class Shared Variable examples example1 imageofEarth resources windowSpec
	Method: #menuBar (resources)	Package: KSU-Template

これが、 Click Widget 。 コレをクリックして、 先ほどの Window の適当な部分をドラッグすると



こんな感じで、枠が出来る

親玉の方に ClickWidget1 が追加されているので、こんな感じに Position を修正 (それぞれ選択すると対応付いているものが選択された状態になる 例を挙げると上の画面の枠を選択すると、下の画面の ClickWidget1 が選択された状態になり 下の画面の ClickWidget1 を選択すると、上の画面の枠が選択された状態になる)

≪ <⊇ & ↔ ⊑	trange G	na Look sp	eciai neip	
Main Window : 地球	Basics	Details Pos	ition Color	
A Cherriager				6
	-			
	т	0		Ø
	R	0	512	Ø
	в	0	256	Ø
		0	•	
			Offecto	Bood
			ear All	nead
	Apply	Cancel	Prev	Next

Propotion はいい感じに比率でオフセットしてくれる物 Offset は普通にピクセルで指定

Apply すると



枠が広がる。

Basics タブを開いて、 ID を #imageOfEarth と指定。 後ほどコレを用いて指定するので、覚えておきましょう。

& <2 & < ¹		
Main Window : 地球	Basics Details Position Click	on Color Widget
	ID: #imageOfEarth	
	Visual: Performer:	
	Default Click: Mappings Selector:	
	Apply Cancel	Prev

経度と緯度を表示するラベルを付ける





適当に2つ設置

String が表示される文字列 ID は使わないけれども、ちゃんとつけておくように



Edit Tools Layout A	rrange Grid Look Special Help
& <2 & <> 5	
👕 Main Window : 地球	Basics Details Position Color
	Label
A Label2	Label: Supplied by Application
	String: 緯度:
	Lookup key:
	Catalog:
	ID: #latitudeLabel
	E Label Is Image
	Apply Cancel Prev Next

二つをこんな感じで設定して Apply

● ○ ○	地球をクリック	
ファイル		
経度:	緯度:	

その結果、こんな感じ。



000	地球をクリック
ファイル	
	e
経度:	緯度:

同様に、2つのラベルの値を設定していく ID に #longitudeFiled と入力し Type は String のままで、 Apply

CUI P Edit Tools Layout Ar	ainter Tool on range Grid L	: 地球をクリック ook Special Help
A longitudeLabel A latitudeLabel be inputField1	Aspect: Menu: ID: Type: Format:	Input Field #longitudeFiled String
	Apply Can	Prev Next

ダ ⊘ ⊗ ↔ ⊑ Main Window : 地球	Basics Details Position Fiv-by Help V
imageOfEarth	Input Field Aspect: Menu: ID: #latitudeField Type: String + Format:

コレで Apply。(恐らく、 Position も何かしら設定したと思うのだが、失敗したので、適当に)

00	地球を	フリック	
ファイル			
		绘麻,	P
経度:		神後・	_

見た目は特に変わらず label を選んだ状態で Color タブを開く





適当に色を選んで指定。すると、ラベルの枠の色が指定した色に変わっている。

ここまで来たら、 Install する
 この時点で example1 を動かすと



なにかしら、やり忘れがあるらしい ということで作業続行

次は Menu Bar をいじっていくが、先ほどのパーツを置いていく元とは別に専用の Menu Bar エディタがある。 これ

	Menu Editor
Menu Edit Move View Test	
🖉 🗄 🖻 🛛 🗶 🗅 🕼	★ 书 년 규 묘.
ファイル 地球 終了	Basic Details State Label:

Smalltalk で GUI なアプリケーションを作るときは Application Model のサブクラスを作って、 windowSpec を作っていく

```
windowSpec のソースを見てみると
KSU-Template, ClickEarth, Class, windowSpec
```

#############ここから、脱線############# 宮崎君の Transcript が行方不明になってしまったので、頑張って探しましょう

00	Workspace
Page Edit Smalltalk	Options Help
D 🗟 🗄 👗	là 🛍 🗲 \land 🝂
Page 1 Variables	
Franscript inspect	
Text or Smalltalk code	All



これで、新しいのは作れるけど、今は×を押して、消してしまっただけで、 Transcript のインスタンス自体は生きてるはず なので、新しく作るのはちょっとイヤ

正解はコレで、

Page Edit Sma	Iltalk Options Help	
🗈 👼 🚍	ኤ 🗈 🛍 🗲 ⁄ء 🗶 🍂	
Page 1 Variable	95	
VisualLauncher	open	



新しく開いた Transcript は 実態は同じだそうで、 Transcript cr; show: 'hoge' を Do it すると、両方に hoge と表示される

0	🔵 🔘 Vis	ualWork	s Person	al /Use	ers/fun	niya/D	ocum	nents/	source	/Smallt	
File	System	Browse	Debug	Painter	Store	Tool	s Wi	ndow	Help		
∛ ∥)		°M °s	2 *** **	°g I	Q	Z	Ŕ			
hoge	,										l
											l
a B	Not conne	ected									

349	
٩H	
Avai	ableFreeOldSpaceBytes: 29,110,568 -> 29,110,568
Avai	ableFreeFixedSpaceBytes: 204,780 -> 401,332
Fili	ig in from:
	Users/fumiya/Documents/source/Smalltalk/20120704/ClickEarth.st
Clic	Earth class>windowSpec defined
hoge	
(B)	
12	lot connected

#############ここから本筋に戻って############## メルカトル図法の地図を表示するために、 windowSpec から Edit

4 D & 4 D	, 🗅 🖪 🗞 🚳	
Main Window:地球 imageOfEarth A longitudeLabel A latitudeLabel DogitudeFiled	Basics Details Posit Click ID: #imageOfEarth Visual: Performer:	ion Color Widget #imageOfEarth
	Default Click: Mappings Selector:	
	Apply Cancel	Prev

この様に書くと、 Visual に #imageOfEarth と入力して、 Apply すると、



こんな感じ

resources の中から imageOfEarth を探している 実態は画像を用意してくれるメソッド



自分で画像を追加するには



Image Editor を使う



Install を押すと、 Install 先を聞かれる

Class:		
		G
Selecto	or:	
] •
(OK Cancel	

clickEarth		
Classes:		
= ClickEarth	in {KSU}	

Selector に zzz 等と入れると、



こんな感じでふえる ペインタに戻って

& D & & T		
Main Window : 地球 imageOfEarth A longitudeLabel A latitudeLabel	Basics Details Pos Clic ID: #imageOfEar	ition Color ck Widget th
latitudeField	Visual: Performer:	#imageOfEarth
	Default Click: Mappings Selector:	#clicked:
	Apply Cancel	Prev

Default Click: に #clicked: とする

経度と緯度の Filed の Aspect を指定

GUI F Edit Tools Layout A 愛公 念~ 祭 型 Main Window : 地球	Painter Tool on rrange Grid L I I C Basics I	: 地球をクリック ook Special Help の Details Position Fly-by Help V
A longitudeLabel	Aspect: Menu: ID: Type:	Input Field #longitudeField #longitudeFiled String
	Format:	
	Apply Can	Prev Next

edit Tous Layout A	Aspect: Menu: ID: Type: Format:	Look Special Help

ここまで出来たら Install

Filed の実態はコレ KSU-Template, Instance, aspects, latitudeField

latitudeField

latitudeField ifNil: [latitudeField := String new asValue]. ^latitudeField

KSU-Template, Instance, aspects, longitudeField

longitudeField

longitudeField ifNil: [longitudeField := String new asValue]. ^longitudeField

imageOfEarth の #clicked: の実態はコレ KSU-Template, Instance, actions, clicked:

clicked: thePoint

I aPoint aWrapper aSensor aBoolean I aPoint := thePoint. (aWrapper := self builder ifMit: [^nil] ifMotNii: [naBuilder I aBuilder componentAt: #imageOfEarth]) ifNii: [^nil]. aSensor := aWrapper widget controller sensor. JunCursors crossCursor showWhile: [aBoolean := true. [aBoolean := true. [aBoolean := true. [aPoint := aSensor cursorPoint. aPoint y = 0 ifTrue: [aPoint := aPoint x @ JunGeometry accuracy]. (self pictureOfEarth bounds containsPoint: aPoint) ifTrue: [self updateLongitudeField: aPoint; updateViewfinderOfEarth. aPoint. aBoolean := aSensor shiftDown]]]

赤色文字部分が重要で、座標位置の更新をしている部分

この時点で example1 を実行してみると、





ちゃんと動いた。

この時点では、 click を離したときにしか反応しない。 例えば、ドラッグをしてもくるくる回らない。

ここまでの話で imageOfEarth の ID を使っていないけど、 imageOfEarth の ID はドラッグ機能を実装するときに使う(既にこのコードでは使われている)

本当に #imageOfEarth を使っている人がいるのか探してみる 探したいものに対して右クリックして、 Senders を選んで一番上の imageOfEarth を選択すると



この様な画面が開いて、コードで実際に使われている部分がハイライトされる



Smalltalk における MVC の関係



Application Model 側から Sensor の値(クリックした座標など)が欲しければ、 Builder に対して ID を指定して そこから Sensor をもらってくる

View & Controller は Filed どの Filed とヒモ付けるかに関しては、 aspects に指定して、 Application Model に どれと対応付けるべきか聞きに行く

Model と View & Controller は名前による結合だけで、疎な結合をしてる

menuBar の Source を見ると、設計図があって、 decode してそれを返せと書いてある KSU-Template, ClickEarth, Class, resources, menuBar

menuBar

"Tools.MenuEditor new openOnClass: self andSelector: #menuBar"

青色部分を inspect it してみると、

00	a Menu
Object Edit Go History Explo	ore Tools Help
Image: A state of the state	₩.
Basic Methods	
self	a Menu
adornment	
groupSizes	
items	
	_
self startUp	

Menu が返ってくる workspace を開いて self startUp とすると



menuBar が出てくる

既に clicked: があるけど、(現在のバージョンは良い感じに修正されているので)古いバージョンで話を進めために 現在のバージョンはバックアップしておく clicked: thePoint

I aPoint aWrapper aSensor aBoolean I aPoint := thePoint. (aWrapper := self builder ifNil: [^nil]

以下省略

先頭に _ を付けて accept

古いバージョンを改めて作る KSU-Template, ClickEarth, Instance, actions, clicked:

clicked: aPoint

(self pictureOfEarth bounds containsPoint: aPoint) ifFalse: [^nil].

self updateLongitudeField: aPoint; updateLatitudeField: aPoint; updateViewfinderOfEarth: aPoint

それを修正 KSU-Template, ClickEarth, Instance, actions, clicked:

clicked: aPoint

l aSensor l (self pictureOfEarth bounds containsPoint: aPoint) ifFalse: [^nil]. self updateLatitudeField: aPoint; updateLatitudeField: aPoint; updateViewfinderOfEarth: aPoint. aSensor := (self controllerAt: #imageOfEarth) sensor. "<-- Click Widget の sensor をもらう"

aSensor altDown inspect "<-- alt(option) キーが押されているかを inspect して確認"

昔は yellowButton などと呼んでいたけど、 Windows のせいで alt キーと… KSU-Template, ClickEarth, Instance, actions, clicked:

clicked: aPoint

I aSensor I (self pictureOfEarth bounds containsPoint: aPoint) ifFalse: [^nil]. self updateLongitudeField: aPoint; updateLatitudeField: aPoint; updateViewfinderOfEarth: aPoint. aSensor := (self controllerAt: #imageOfEarth) sensor.

```
[aSensor altDown] whileTrue: "<-- alt キーを押し続けている間は"
            [Processor yield.
Transcript
                cr;
                show: aSensor cursorPoint printString]
clicked (クリックをやめた後)が呼ばれたあとに alt が押されていたら反応する
ドラッグではない
KSU-Template, ClickEarth, Instance, actions, clicked:
```

clicked: aPoint

I aBlock aSensor I aBlock := . [:thePoint I (self pictureOfEarth bounds containsPoint: thePoint) ifFalse: [^nil]. self updateLongitudeField: thePoint; updateLatitudeField: thePoint; updateViewfinderOfEarth: thePoint]. aBlock value: aPoint. aSensor := (self controllerAt: #imageOfEarth) sensor. [aSensor altDown] whileTrue: [Processor yield. aBlock value: aSensor cursorPoint]

コードクローンがあると嫌なので、座標アップデート部分をクロージャ化して、更新するように

ドラッグ状態(alt キーが押された状態)で北極の方を超えると

00	Exception
	Unhandled exception: Cannot divide a number by zero
Deb	ug Proceed Terminate
	Copy Stack
Double cl Double>>< Double>>/	<pre>lass(ArithmeticVlector:arg:errorString: quotientFromDouble: /</pre>
Jun3dPoir	nt>>/
Jun3dPoir	nt(JunPoint)>>retry:coercing:

zero で割るなと怒られる 北極の方に座標を出すときに除算が含まれているので、こうなってしまう

clicked: aPoint

I aBlock aSensor I aBlock :=
[:thePoint]
thePoint y = 0 ifTrue: [thePoint y: JunGeometry accuracy].
(self pictureOfEarth bounds containsPoint: thePoint) ifFalse: [^nil].
self
updateLongitudeField: thePoint;
updateLatitudeField: thePoint;
updateViewfinderOfEarth: thePoint].
aBlock value: aPoint.
aSensor := (self controllerAt: #imageOfEarth) sensor.
[aSensor altDown] whileTrue:
[Processor yield.
aBlock value: aSensor cursorPoint]

青色部分を inspect すると

Object Edit	Go His	ets esse	re Tools	Help		
Basic Metho 23 🚨 self	ods	1	1.0d-12			T
 <i>IEEE FOIT</i> 1 2 2 3 3 4 5 6 7 	nula	×				

```
限りなく 0 に近い値なので、これで 0 除算はでなくなる
```

```
この時点では、メルカトル図法の地図からはみ出ると、それ以降、更新されなくなる
バックアップを取った元のコードでは、その問題は起きない
KSU-Template, ClickEarth, Instance, actions, _clicked:
_clicked: thePoint
       I aPoint aWrapper aSensor aBoolean I aPoint := thePoint.
      (aWrapper := self builder

iffNil: [^nil]

iffNotNil: [:aBuilder | aBuilder componentAt: #imageOfEarth]) ifNil: [^nil].

aSensor := aWrapper widget controller sensor.

JunCursors crossCursor showWhile:

[aBoolean] whileTrue:

[aBoolean] whileTrue:

[aBoolean] whileTrue:

[aPoint := aSensor cursorPoint.

aPoint y = 0 ifTrue: [aPoint := aPoint x @ JunGeometry accuracy].

(self pictureOfEarth bounds containsPoint: aPoint)

ifTrue:
       (aWrapper := self builder
                                                     [self
                                                            updateLongitudeField: aPoint;
                                                     updateLatitudeField: aPoint;
updateViewfinderOfEarth: aPoint.
aBoolean := aSensor shiftDown]]]
それを直すために ifFalse で nil を返して抜け出しているので、逆に ifTrue の時だけ更新する様にして
抜け出すことがないようにすると動き続けるようになる
KSU-Template, ClickEarth, Instance, actions, clicked:
clicked: aPoint
       I aBlock aSensor I
        aBlock :=
                      [:thePoint I
                      thePoint y = 0 ifTrue: [thePoint y: JunGeometry accuracy].
(self pictureOfEarth bounds containsPoint: thePoint)
                              ifTrue:
[self
                                            updateLongitudeField: thePoint;
updateLatitudeField: thePoint;
updateViewfinderOfEarth: thePoint]].
```

aBlock value: aPoint. aSensor := (self controllerAt: #imageOfEarth) sensor. [aSensor altDown] whileTrue: [Processor yield. aBlock value: aSensor cursorPoint]