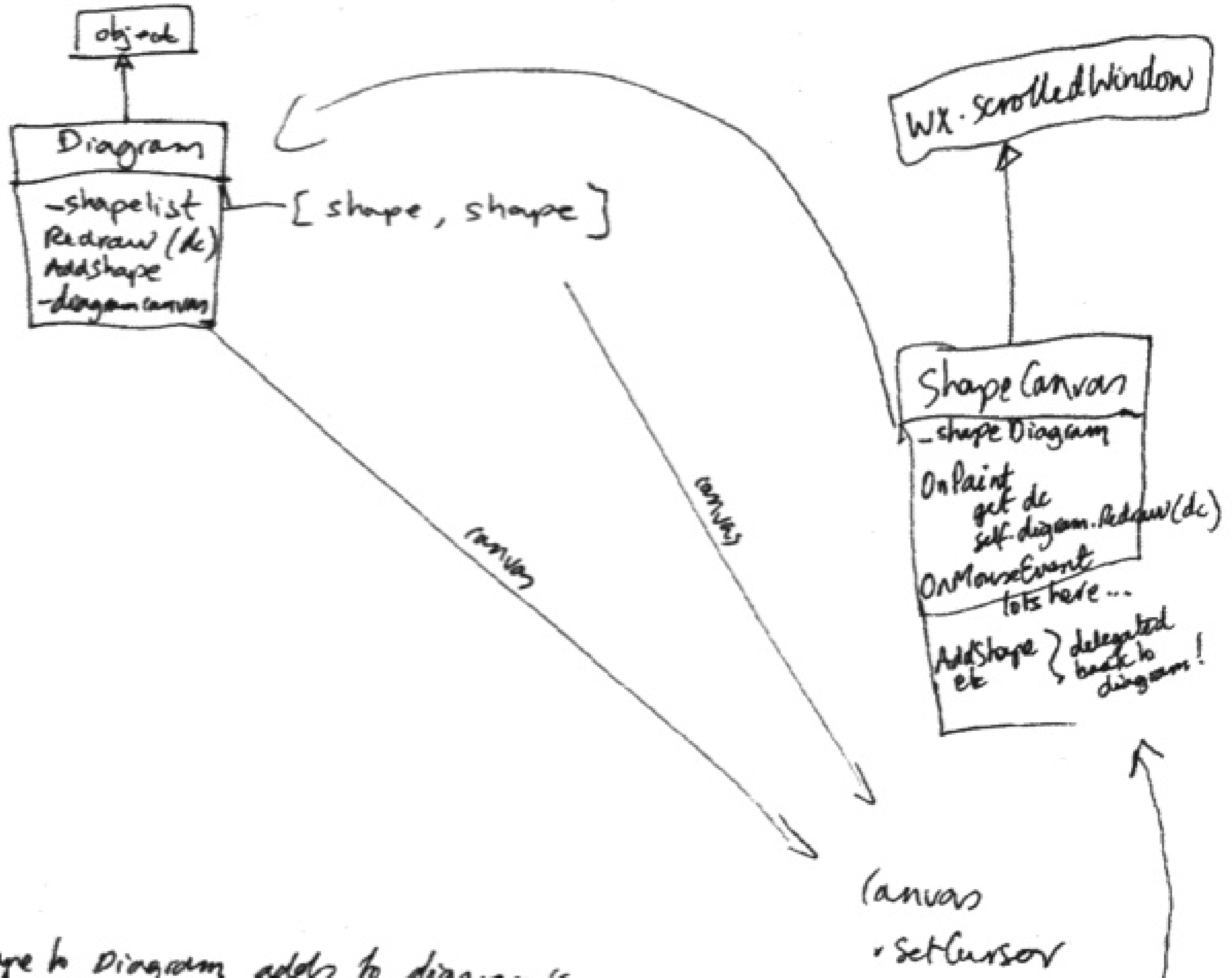


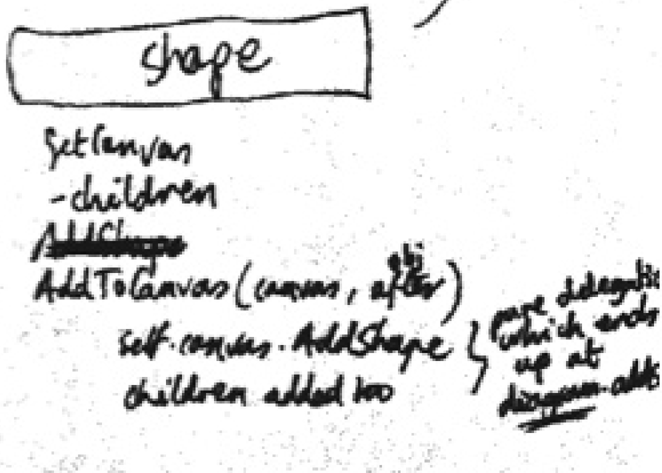
- Adding shape to Diagram adds to diagram's shape list

- + sets shapes's canvas to be same as diagrams.

- Diagram Redraw(dc) loops each shape.Draw(dc)
  canvas not involved.

- Diagram has associated $\overset{\text{\_diagramcanvas}}{\text{ShapeCanvas}}$ which doesn't do anything from Diagram's point of view (except setting cursors).

- setting shape to a $\overset{\text{shape}}{\text{canvas}}$ loops through shape's children + sets their shapecanvas as well.


shape
SetCanvas
-children
~~AddShape~~
AddToCanvas(canvas, after obj)
self.canvas.AddShape
children added too
more delegation which ends up at diagram.add

Home Trees Index Help wxPython 2.8.9.2

Package wx :: Package lib :: Package ogl :: Class ShapeCanvas [frames | no frames]

Type ShapeCanvas

```
object --+
         |
    Object --+
             |
    EvtHandler --+
                 |
            Window --+
                     |
                 Panel --+
                         |
            ScrolledWindow --+
                             |
                            ShapeCanvas
```

GetSize() ✓ no getWidth()

Method Summary

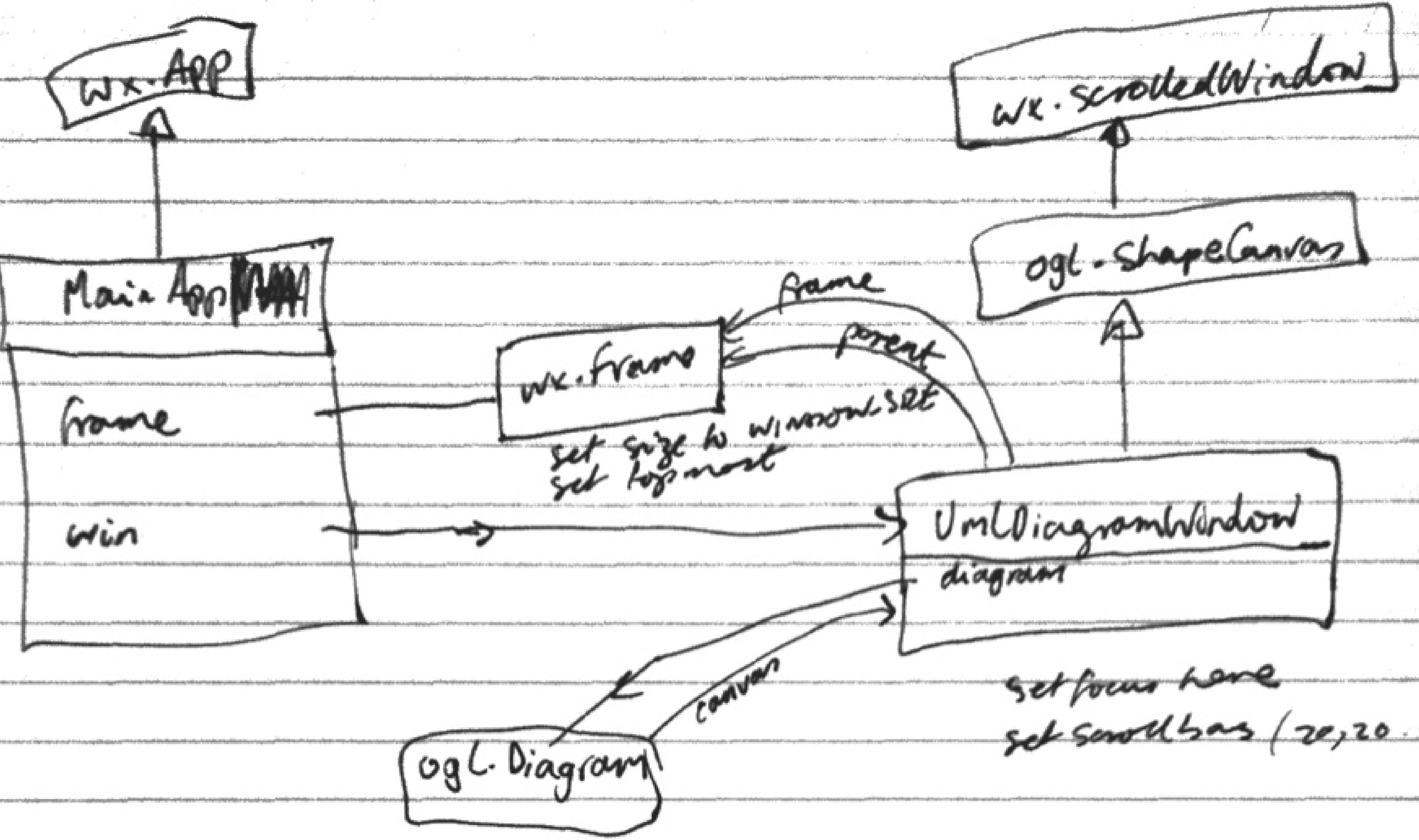
	`__init__(self, parent, id, pos, size, style, name)`
	`AddShape(self, object, addAfter)`
	`FindShape(self, x, y, info, notObject)`
	`GetDiagram(self)`
	`GetQuickEditMode(self)`
	`InsertShape(self, object)`
	`OnBeginDragLeft(self, x, y, keys)`
	`OnBeginDragRight(self, x, y, keys)`
	`OnDragLeft(self, draw, x, y, keys)`
	`OnDragRight(self, draw, x, y, keys)`
	`OnEndDragLeft(self, x, y, keys)`
	`OnEndDragRight(self, x, y, keys)`
	`OnLeftClick(self, x, y, keys)`
	`OnMouseEvent(self, evt)`
	`OnPaint(self, evt)`
	`OnRightClick(self, x, y, keys)`
	`Redraw(self, dc)`
	`RemoveShape(self, object)`
	`SetDiagram(self, diag)`
	`Snap(self, x, y)`

Property Summary

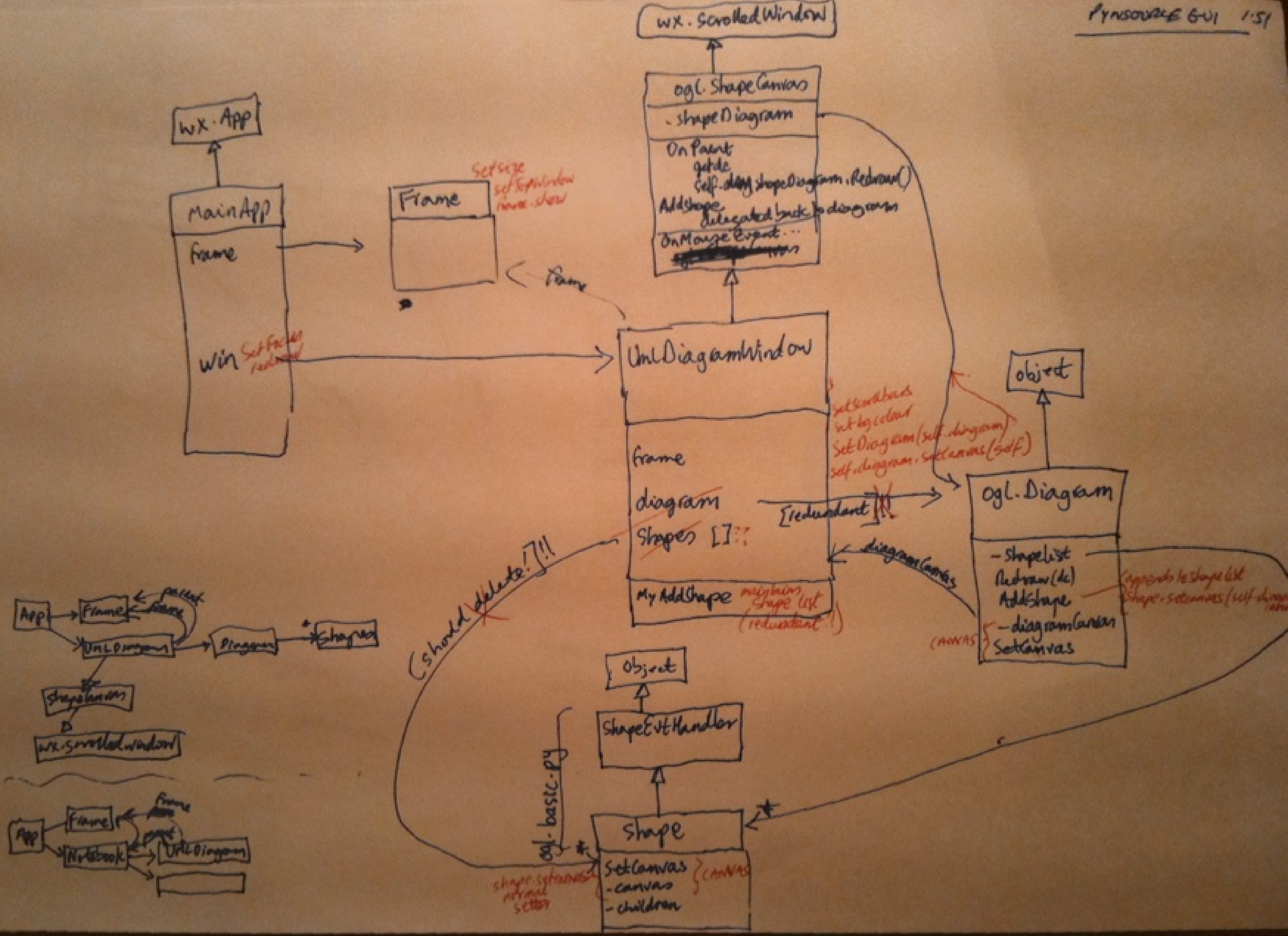
Home Trees Index Help wxPython 2.8.9.2

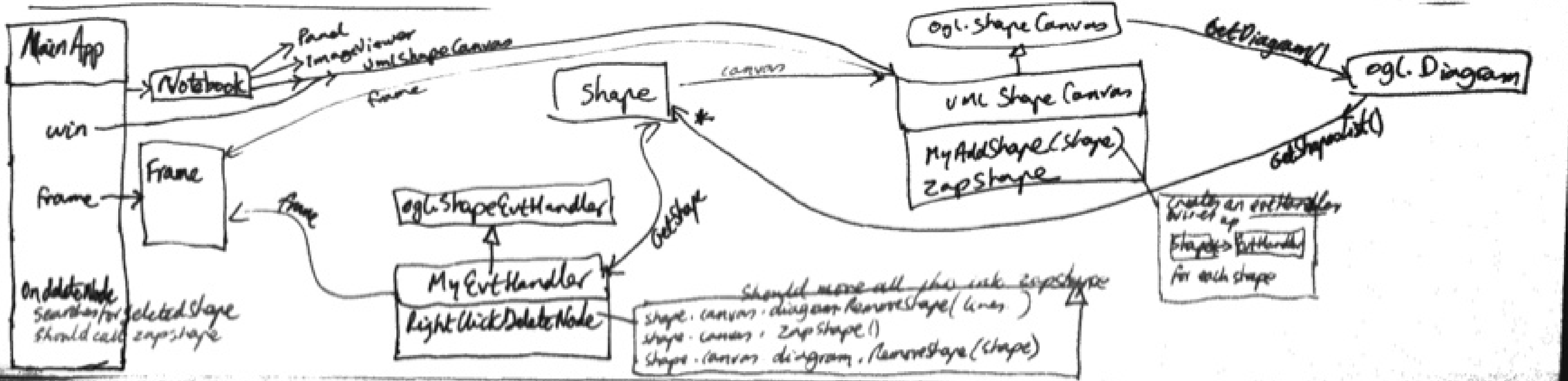
Generated by Epydoc 2.1.20050511.rpd on Mon Feb 16 12:52:24 2009 http://epydoc.sf.net

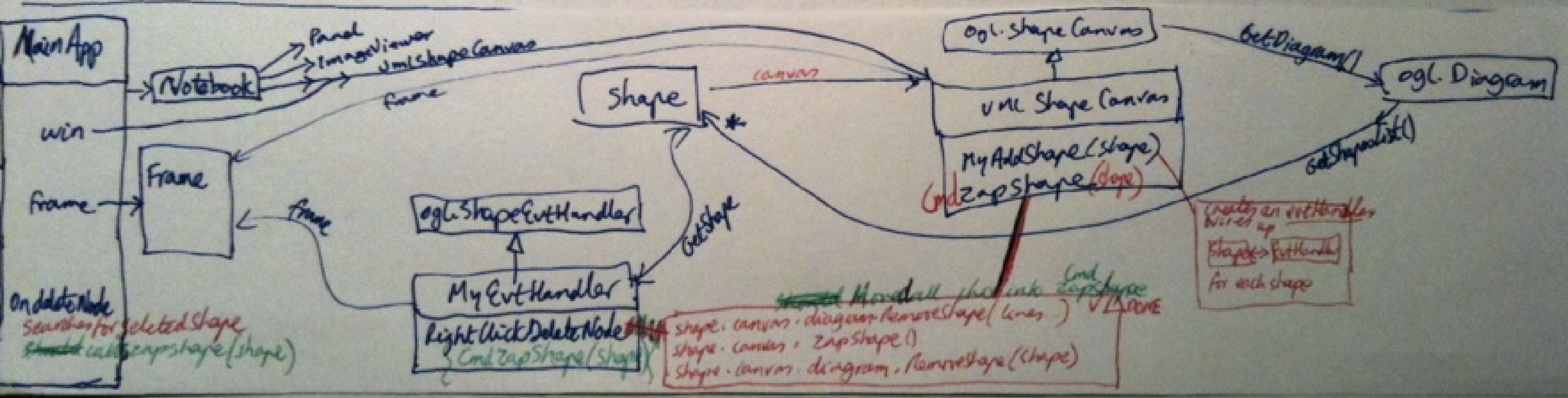
1·4c no GetSize() } in ShapeCanvas
1·5 or getwidth() }

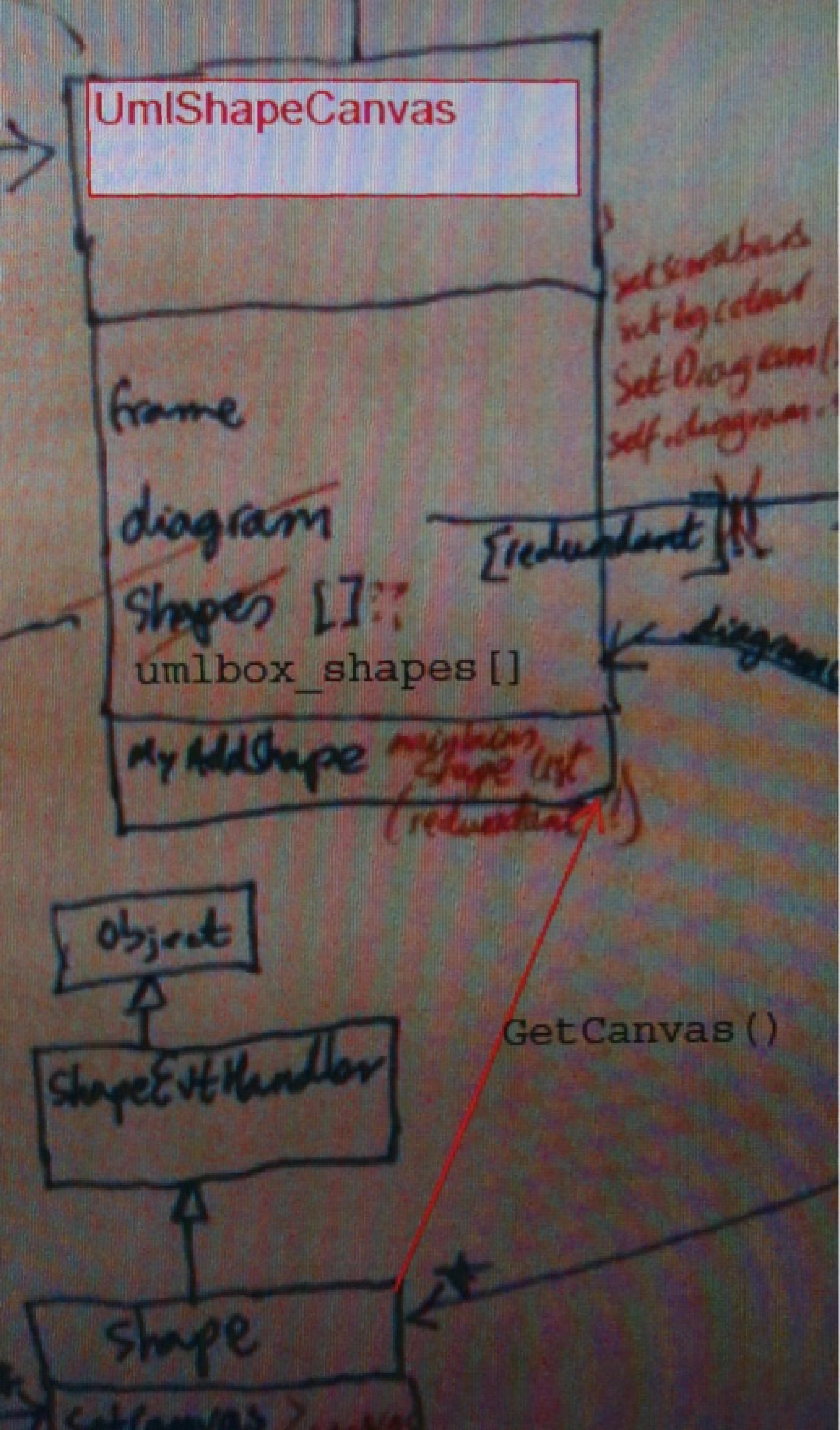

wx.App
wx.ScrolledWindow
ogl.ShapeCanvas
MainApp
frame
win
wx.frame
frame
parent
set size to window.set
set topmost
UmlDiagramWindow
diagram
canvas
ogl.Diagram
set focus here
set scrollbars (20,20.

Pynsource GUI 1:51



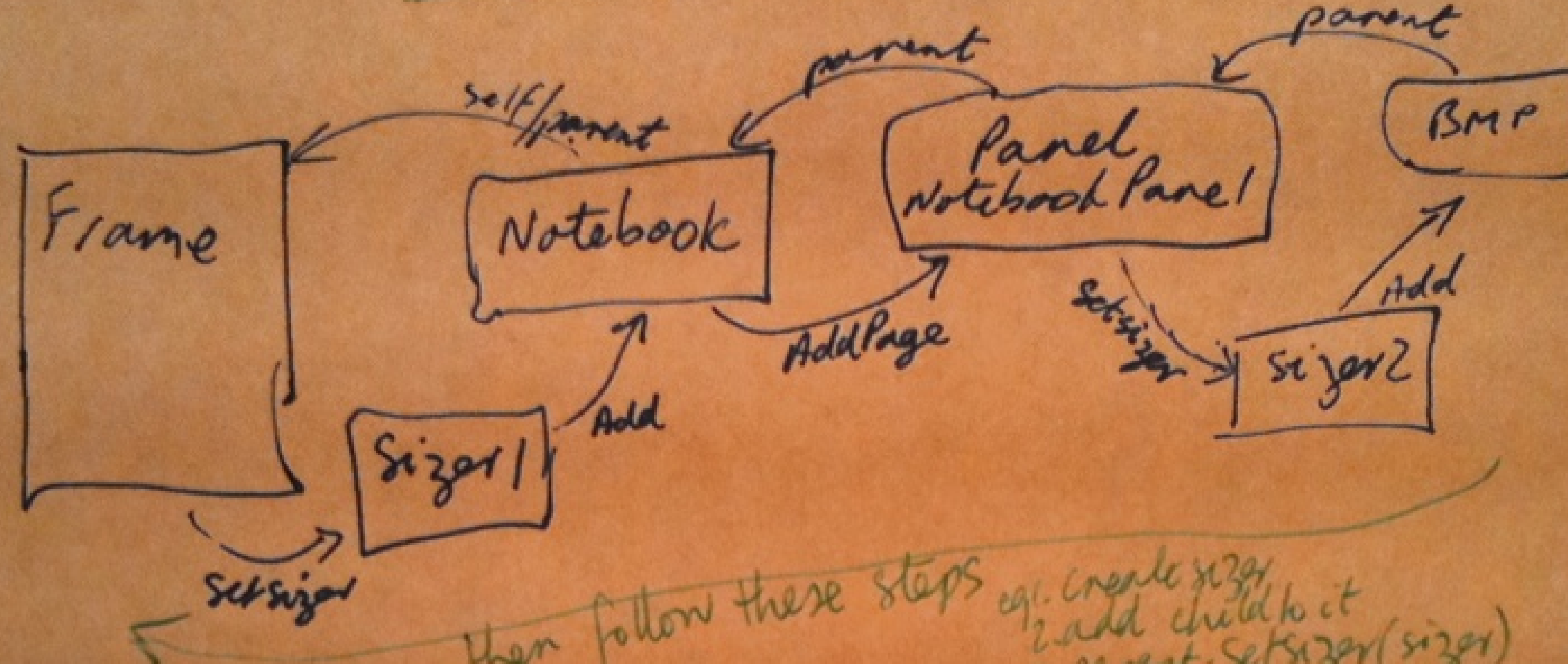

MainApp
win
frame
OnDeleteNode
Searches for selected shape
should call zapshape
Notebook
Panel
ImageViewer
UmlShapeCanvas
Frame
Frame
Frame
Shape
canvas
*
GetShape
ogl.ShapeEvtHandler
MyEvtHandler
RightClickDeleteNode
Should move all this into zapshape
shape.canvas.diagram.RemoveShape(lines)
shape.canvas.ZapShape()
Shape.canvas.diagram.RemoveShape(shape)
ogl.ShapeCanvas
UmlShapeCanvas
MyAddShape(shape)
ZapShape
Creates an evtHandler
wired up
Shape → EvtHandler
for each shape
GetDiagram()
ogl.Diagram
GetShapeList()


MainApp
win
frame
OnDeleteNode
searches for selected shape
~~should~~ call zapshape(shape)
Notebook
Panel
ImageViewer
UmlShapeCanvas
frame
Frame
Shape
canvas
*
GetShape
frame
ogl.ShapeEvtHandler
MyEvtHandler
RightClickDeleteNode
{ CmdZapShape(shape) }
shape.canvas.diagram.RemoveShape(lines...)
shape.canvas.ZapShape()
shape.canvas.diagram.RemoveShape(shape)
~~should~~ Move all this into Cmd ZapShape
✓ DONE
ogl.ShapeCanvas
UmlShapeCanvas
MyAddShape(shape)
CmdZapShape(shape)
GetDiagram()
ogl.Diagram
GetShapeList()
Creates an evtHandler
wired up
Shape -> EvtHandler
for each shape


UmlShapeCanvas
set scrollbars
set bgcolour
SetDiagram(
self.diagram.=
frame
~~diagram~~
[redundant]!!
~~Shapes []~~ ??
umlbox_shapes[]
diagram
MyAddShape
maintain shape list
(redundant!)
Object
ShapeEvtHandler
GetCanvas()
*
Shape
SetCanvas

first create chain of objects with parenting

wxglade technique

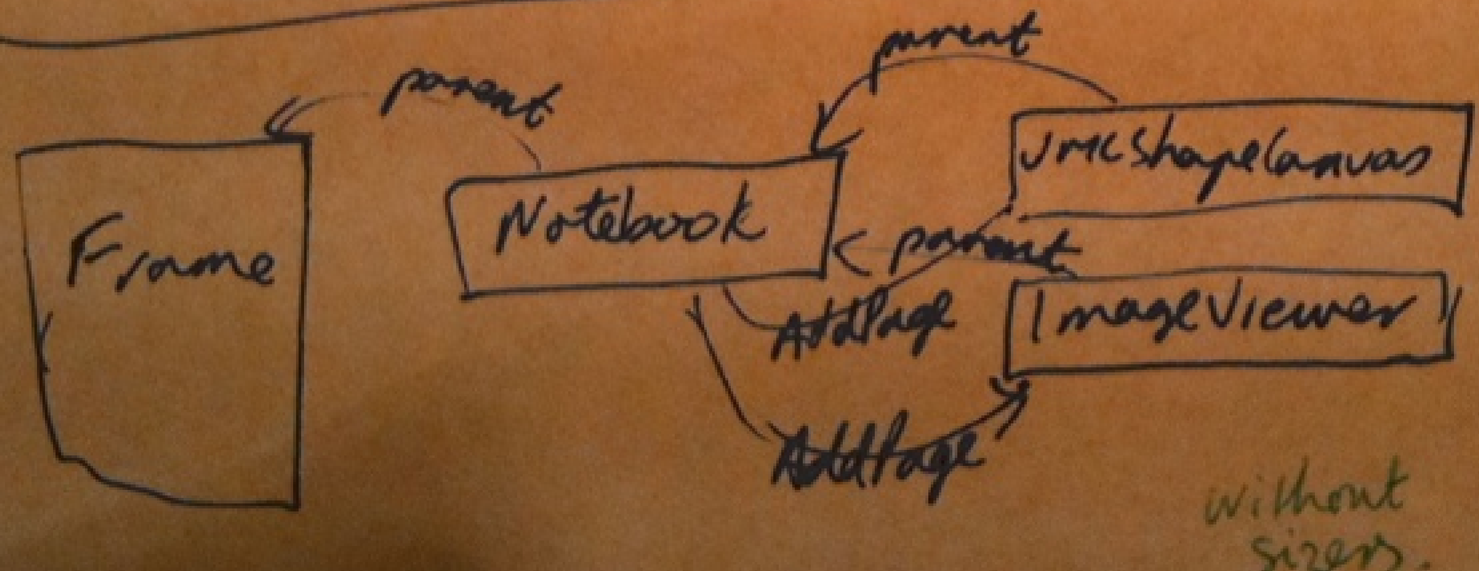


then follow these steps

eg. create sizer
2 add child to it
3 parent.SetSizer(sizer)

Sizer1.Fit(self/frame)
frame.Layout().

Pynsource GUI



without sizers.