

NEOFIT PRACTICAL GUIDE

VisualNEO Win – Adapting publications to different resolutions

A from-scratch experimentation manual

Objective: understand NeoFit by playing with a brand-new, very simple publication before applying it to a large application. The idea is to observe what happens when you shrink and enlarge the window, and to learn which option works best for each type of object.

1. The core idea of NeoFit

NeoFit uses the size at which you originally designed the publication as its reference. Later, when the window changes size, it can reposition, stretch, keep fixed, or scale the objects according to the rules you register.

Example of a design size:

```
NeoFit_SetDesignSize "1280" "720"
```

This means: “my interface was originally designed at 1280 × 720.” The design size does NOT necessarily have to match the current monitor size.

2. Basic routine

The general structure we are going to test is:

```
:FitPage  
NeoFit_Clear  
NeoFit_SetDesignSize "1280" "720"  
  
NeoFit_RegisterObject "ObjectName" "STRETCH" "STRETCH" "L" "T" "0" "0"  
  
NeoFit_Apply  
Return
```

3. What each command does

NeoFit_Clear

Clears any previously registered objects. This is useful before setting up the rules for the page you are about to adapt.

```
NeoFit_Clear
```

NeoFit_SetDesignSize

Specifies the original width and height used to design the publication.

```
NeoFit_SetDesignSize "1280" "720"
```

NeoFit_RegisterObject

Registers an object and defines how it should behave when the available size changes. In the initial tests we will focus on three modes: FIXED, STRETCH, and UNIFORM.

```
NeoFit_RegisterObject "Object" "HORIZONTAL_MODE" "VERTICAL_MODE"  
"H_ANCHOR" "V_ANCHOR" "0" "0"
```

NeoFit_Apply

After registering the objects, this command applies the calculations and updates their positions/sizes.

```
NeoFit_Apply
```

4. Modes to experiment with

Mode	What it does	Where to try it
FIXED	Keeps the object's size.	Buttons, icons, controls you don't want to deform.
STRETCH	Lets the object stretch with the available space.	Backgrounds, panels, bars, wide fields.
UNIFORM	Scales while keeping the aspect ratio.	Logos, images, and elements that must not be deformed.

5. Anchors / positions

Besides the size, NeoFit needs to know which area the object should stay anchored to.

Letter	Horizontal	Vertical
L / T	L = Left	T = Top
C / C	C = Center	C = Center
R / B	R = Right	B = Bottom

Example: "R" "B" = keep the reference toward the bottom-right corner.

Example: "C" "C" = keep the reference toward the center.

6. First practice from scratch

To learn it without risking a real application, create a new 1280 × 720 publication and place only a few objects:

- A large rectangle as the background.
- A button at the top left.
- A button at the top right.
- A button at the bottom right.
- A logo or image in the center.
- A wide text field.

7. Test A – fixed object

Create a button named BotonPrueba and check that it keeps its size in the top-left corner.

```
NeoFit_RegisterObject "BotonPrueba" "FIXED" "FIXED" "L" "T" "0" "0"
```

Run the publication, shrink the window, and then enlarge it. Watch whether the button keeps its size and its relationship to the corner.

8. Test B – fixed to the right

```
NeoFit_RegisterObject "BotonDerecha" "FIXED" "FIXED" "R" "T" "0" "0"
```

The goal is to verify that a button can keep its size while staying anchored to the top-right area.

9. Test C – bottom-right corner

```
NeoFit_RegisterObject "BotonSalir" "FIXED" "FIXED" "R" "B" "0" "0"
```

This test is very important for buttons such as Save, Exit, Back, or any controls you want to keep near an edge.

10. Test D – stretching background

```
NeoFit_RegisterObject "Fondo" "STRETCH" "STRETCH" "L" "T" "0" "0"
```

Try it with a large rectangle. When the window size changes, it should be a good candidate for taking up more or less space.

11. Test E – proportional image

```
NeoFit_RegisterObject "Logo" "UNIFORM" "UNIFORM" "C" "C" "0" "0"
```

The purpose is to observe whether the image changes size without ending up squashed or unevenly stretched.

12. Test F – mixing horizontal and vertical

NeoFit lets you think about horizontal and vertical behavior separately. This is especially useful for bars, fields, or panels. Experiment first with simple combinations and write down the results.

```
NeoFit_RegisterObject "CampoPrueba" "STRETCH" "FIXED" "L" "T" "0" "0"
```

This combination aims to let the object vary in width while keeping its height.

13. Complete routine for a test page

```
:FitPage  
NeoFit_Clear  
NeoFit_SetDesignSize "1280" "720"  
  
NeoFit_RegisterObject "Fondo" "STRETCH" "STRETCH" "L" "T" "0" "0"  
NeoFit_RegisterObject "BotonIzquierda" "FIXED" "FIXED" "L" "T" "0" "0"  
NeoFit_RegisterObject "BotonDerecha" "FIXED" "FIXED" "R" "T" "0" "0"  
NeoFit_RegisterObject "BotonSalir" "FIXED" "FIXED" "R" "B" "0" "0"  
NeoFit_RegisterObject "Logo" "UNIFORM" "UNIFORM" "C" "C" "0" "0"  
NeoFit_RegisterObject "CampoPrueba" "STRETCH" "FIXED" "L" "T" "0" "0"  
  
NeoFit_Apply  
Return
```

14. When to call FitPage

The idea behind the NeoFit example is to run the routine when the page is entered and also when the window changes size. In VisualNEO this lets you recalculate the interface while trying out different sizes.

```
PgEnterAction=GoSub "FitPage"
```

```
ResizedAction=GoSub "FitPage"
```

IMPORTANT: before copying these lines literally into a real publication, verify which VisualNEO section each action belongs in. For learning purposes, use a test publication first.

15. NeoFit_PinObject

The plugin also provides NeoFit_PinObject. Its purpose is to link or “pin” one object to another, which is useful when two elements must stay together as the resolution changes.

Conceptual examples: label + field; icon + button; photo + name; panel + related element. Since its exact parameters must follow the plugin documentation, it is best to practice it after mastering RegisterObject and Apply.

16. What to watch while playing with the window

- Does the object keep its size when you use FIXED?
- Does STRETCH deform it, or does it behave as you expected?
- Does UNIFORM preserve the image's aspect ratio?
- Does R correctly keep the object on the right side?
- Does B correctly keep the object at the bottom?
- Does C center it the way you expected?
- What happens when you shrink the window a lot?
- What happens when you enlarge it beyond the original size?
- Which combinations work best for fields, buttons, panels, and images?

17. Table for recording your tests

Object	Horizontal	Vertical	H Anchor	V Anchor	Result

18. When we move on to a large application

In an application with many pages, it is not a good idea to start by registering hundreds of objects before defining a strategy. First we must learn which rules work for each class of object. Then we can organize routines per page and, wherever possible, automate the generation of the registrations.

Example organization: :Fit_Principal, :Fit_Propietarios, :Fit_Inquilinos, :Fit_Legajo, etc. Each page can have its own rules.

19. Golden rule for experimenting

Do NOT start with the final application.

First create a small publication, save it as a test, and change only one rule at a time. That way you will be able to see exactly what FIXED, STRETCH, UNIFORM, and each anchor do without getting confused.

20. Recommended learning sequence

1. Create a 1280 × 720 publication.
2. Add a single button and test FIXED.
3. Add another button and test R/T.
4. Add another one and test R/B.
5. Add a background and test STRETCH.
6. Add an image and test UNIFORM.
7. Add a field and mix horizontal STRETCH with vertical FIXED.
8. Try window sizes smaller than 1280 × 720.
9. Try sizes larger than 1280 × 720.
10. Write down which combination works best for each type of object.
11. Only then apply the strategy to a real page.

END OF THE TEST GUIDE