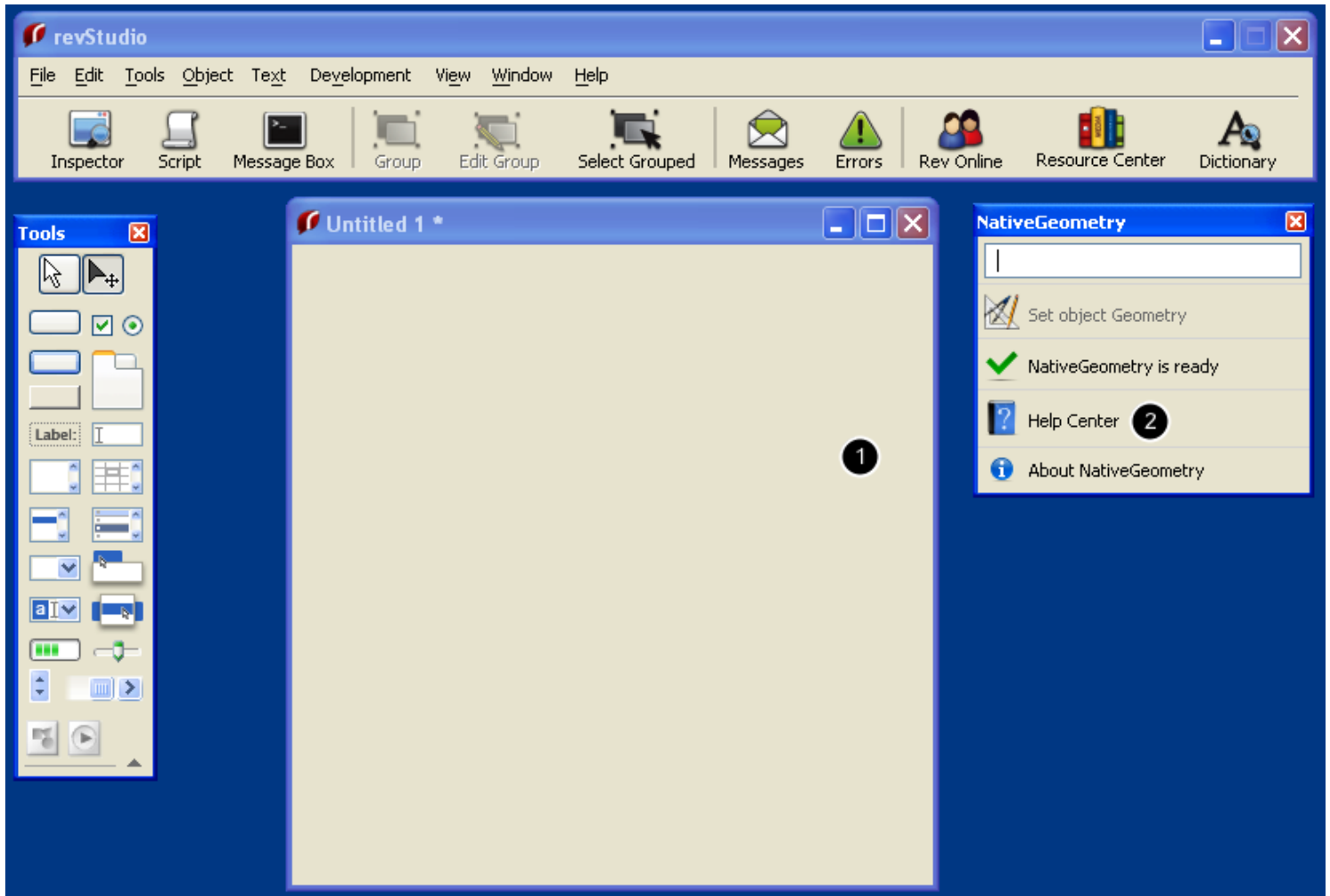


## NativeSoft NativeGeometry Tutorial – 1

---

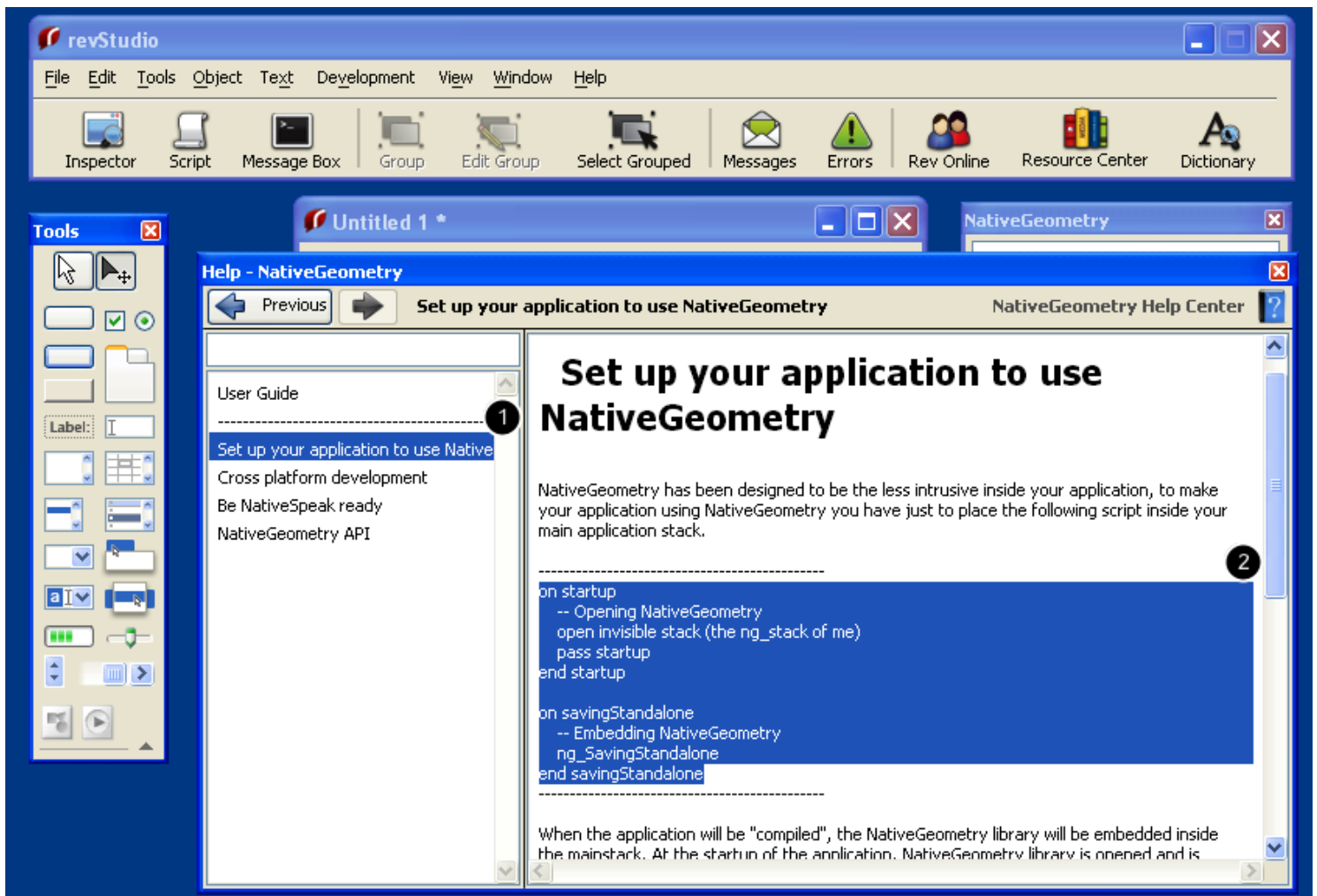
NativeGeometry is a complex technology, its core contains thousands of lines just to do one simple thing: The entire geometry management of your application. This tutorial will show you how to get started with NativeGeometry. After completing this tutorial you will be able to do basic operations with NativeGeometry.

### Make an Application Using NativeGeometry

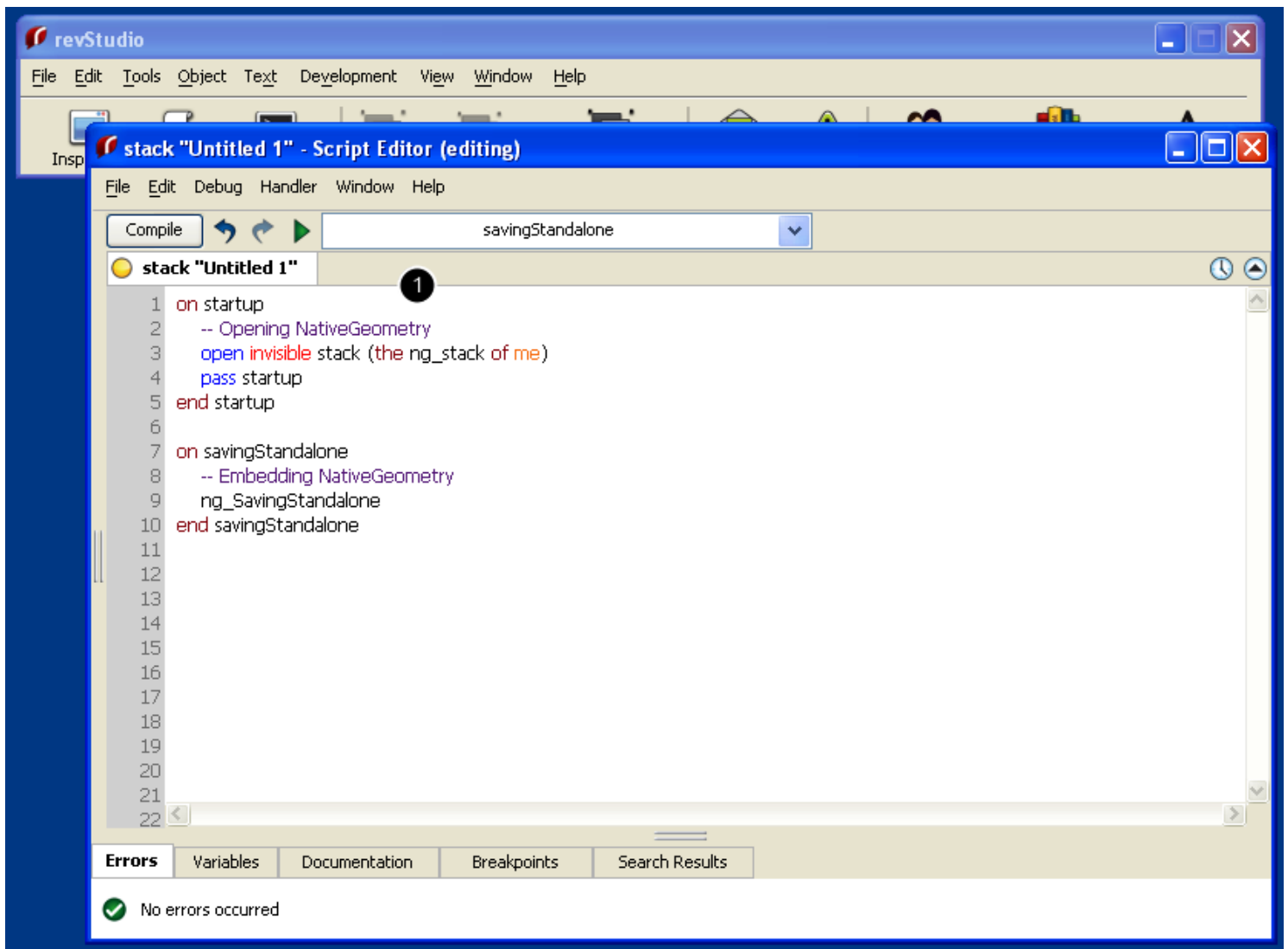


The implementation of NativeGeometry is straightforward.

First, create an empty stack (1), then open the NativeGeometry help center (2).



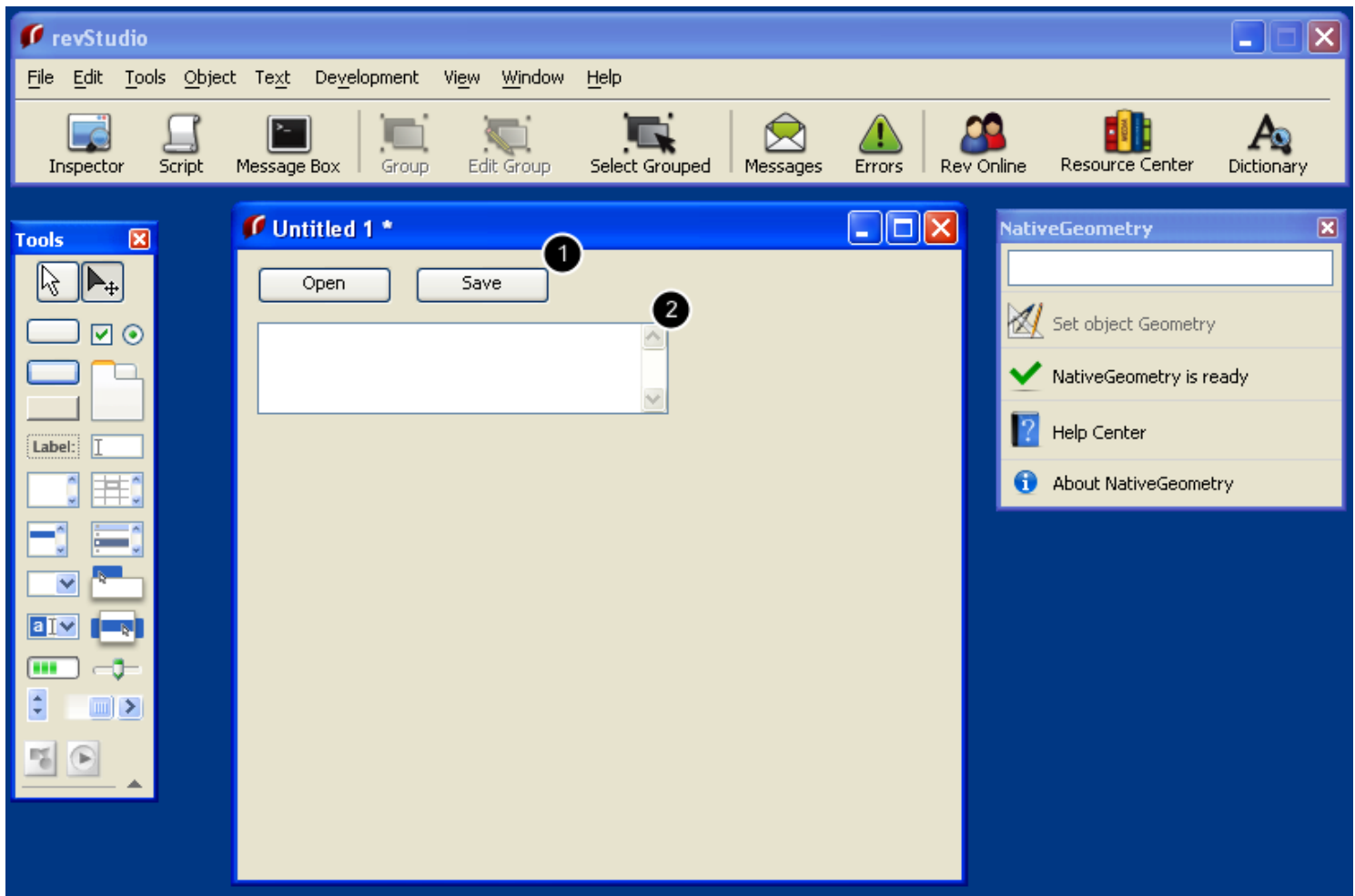
In the help center, click "Set up your application to use NativeGeometry" (1), then copy the highlighted code snippet (2) by pressing "Ctrl+C" or "Command+C" .



Finally, edit the script of the newly created stack, and simply paste the copied code snippet using "Ctrl+V" or "Command+V" to paste.

Now your application has NativeGeometry embedded and you will be able to use it.

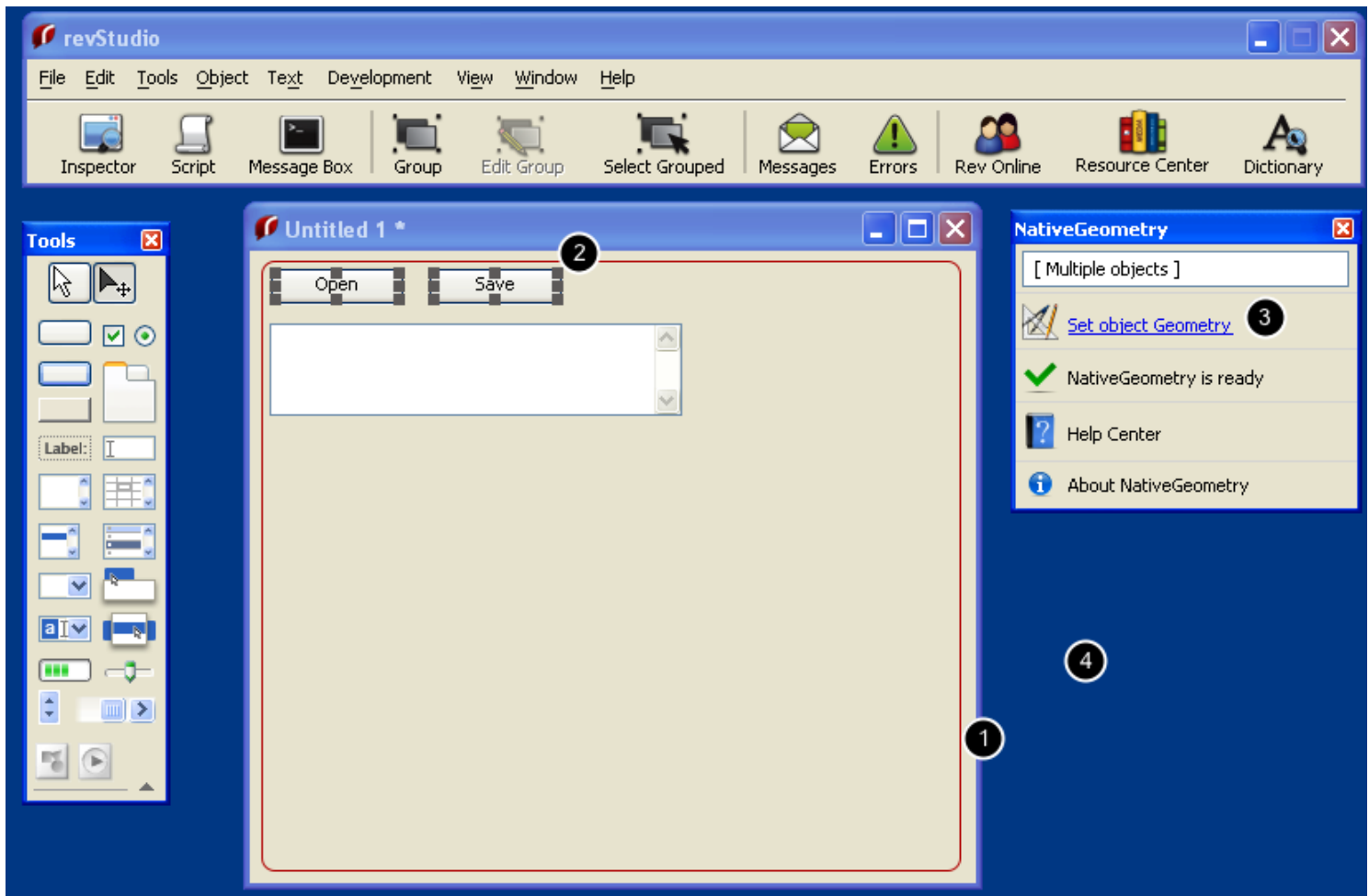
## A Simple Application



We will write a really simple application, which you can complete within the NativeGeometry trial limitations if you do not already have a license for NativeGeometry.

Simply place two buttons (1) and a scrolling field (2) in the newly created stack.

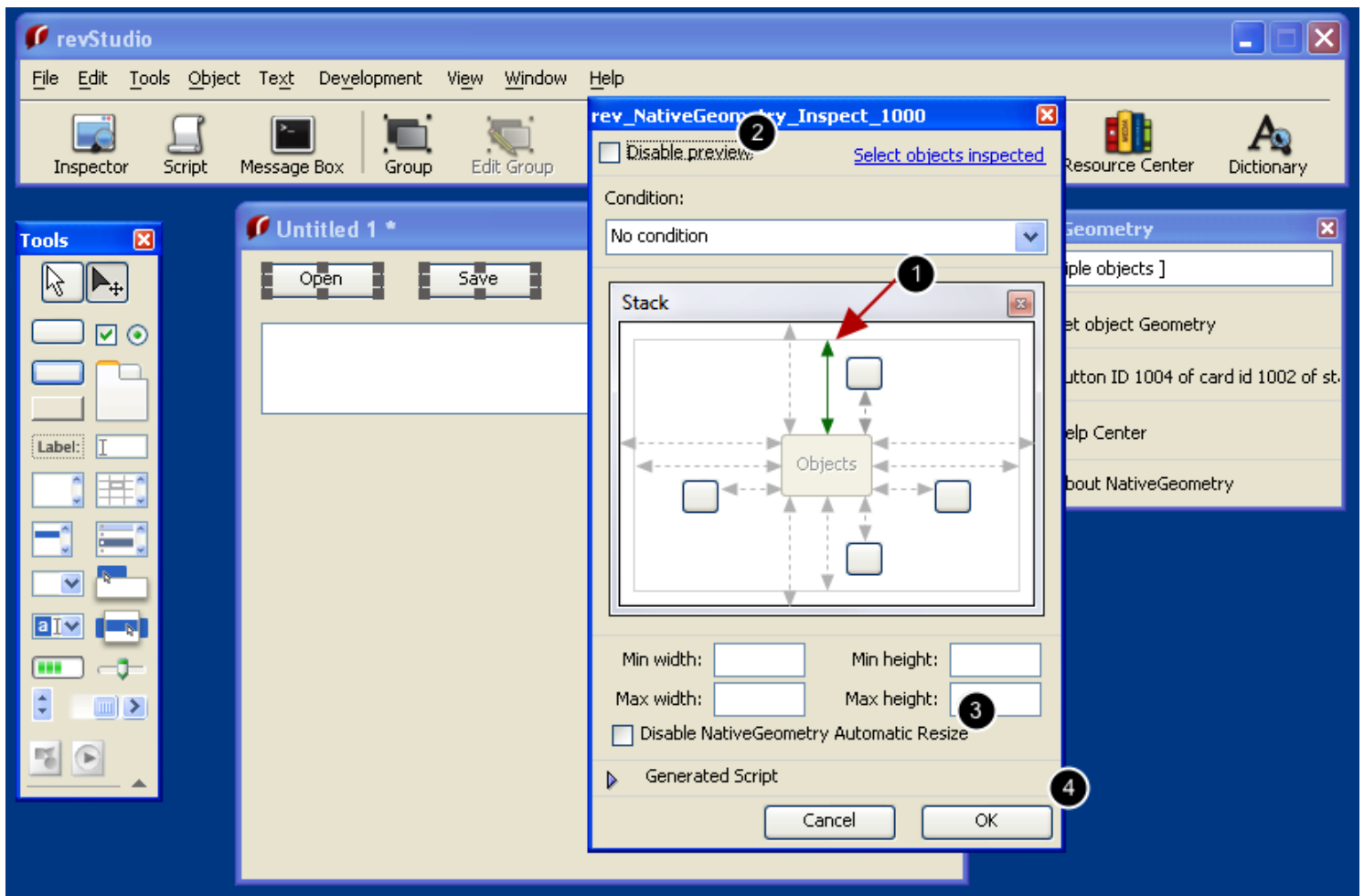
## Setting the geometry of the "Open" and "Save" buttons



Now, we want to have the "Open" and "Save" buttons at the top "padding" of the stack, so that they look good on all platforms.

The padding is a virtual property that depends on the platforms where the applications will be launched. This property avoids objects colliding with each other and prevents you having to set fixed sizes. The red rect represent the padding property. (1)

Now, select the "Open" and "Save" buttons (2), then click "Set object Geometry" (3).



The NativeGeometry inspector is a complete palette that enables you to define the geometry of your application quickly and easily .

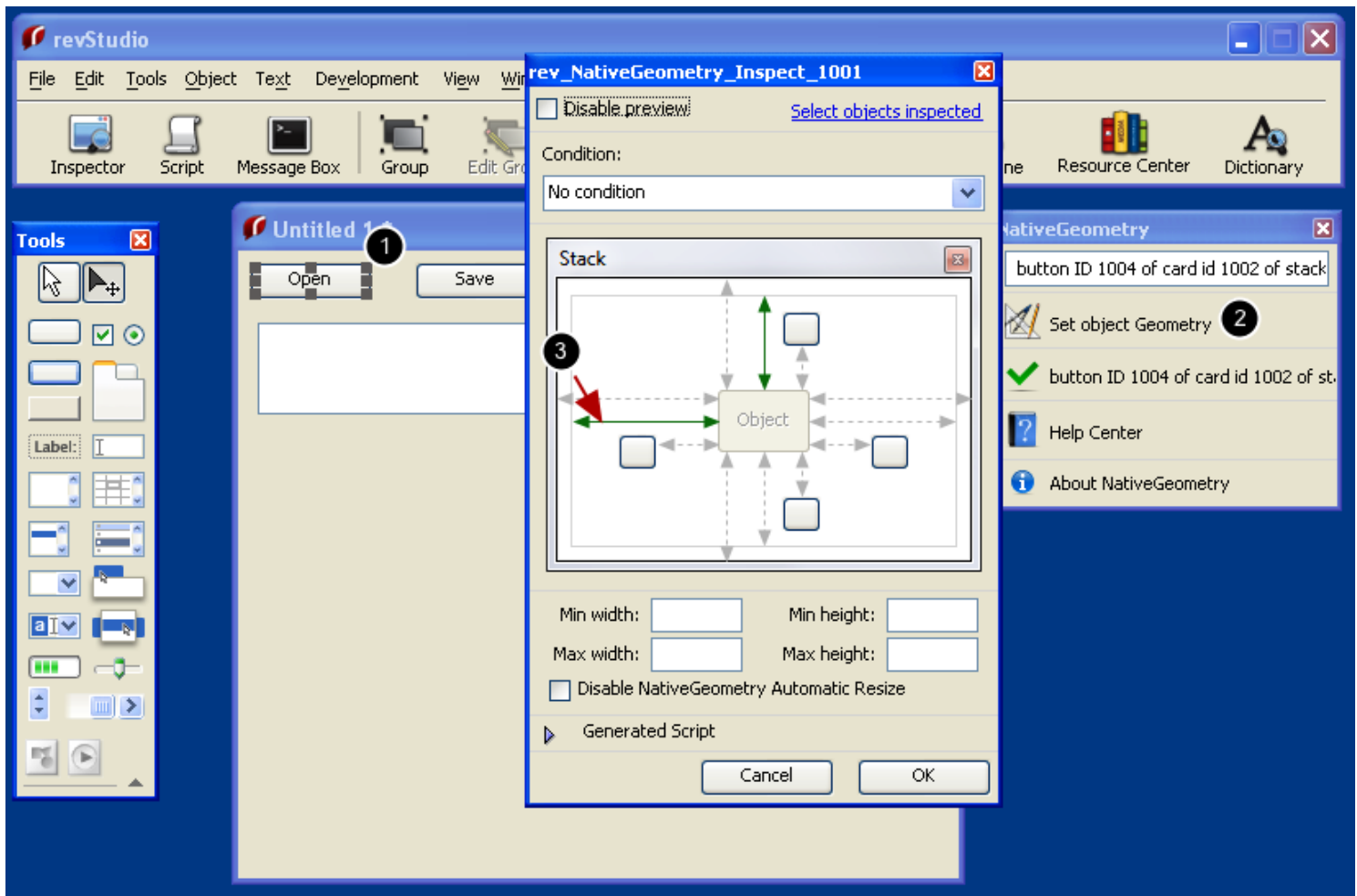
To set the "Open" and the "Save" object to the top padding of the stack, simply click "1". You will see that instantly your objects have been placed and resized!

Don't worry, the NativeGeometry palette by default let's you preview all your modifications, if you click "Cancel" your objects will have their original size and positions restored. To disable the preview option, click "Disable preview" (2). Also you will notice that your objects have been resized, this is normal; NativeGeometry automatically adapts the objects' sizes depending on their content and the platform. You can turn off this option using "Disable NativeGeometry Automatic Resize" (3).

If you want to have an application localizable that looks good on all platforms, do not activate "Disable NativeGeometry Automatic Resize" (3), unless you do not want to have NativeGeometry resizing some objects.

To apply the new geometry, click "OK" (4).

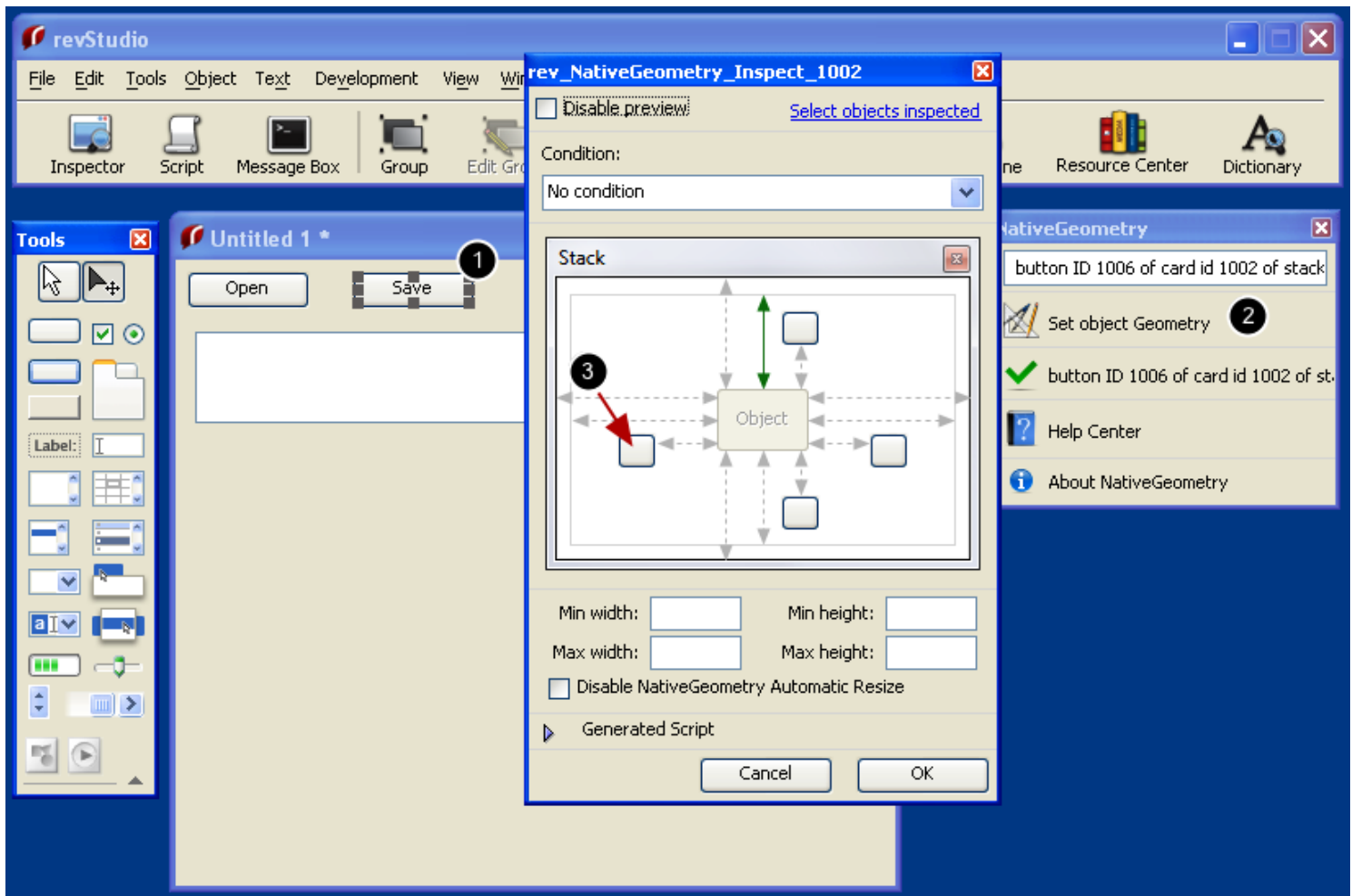
## Set the Geometry of the "Open" Button



Select the "Open" button (1), click "Set object Geometry" (2) then set its left to the left padding of the stack (3).

That's all, the "Open" button is now strictly placed.

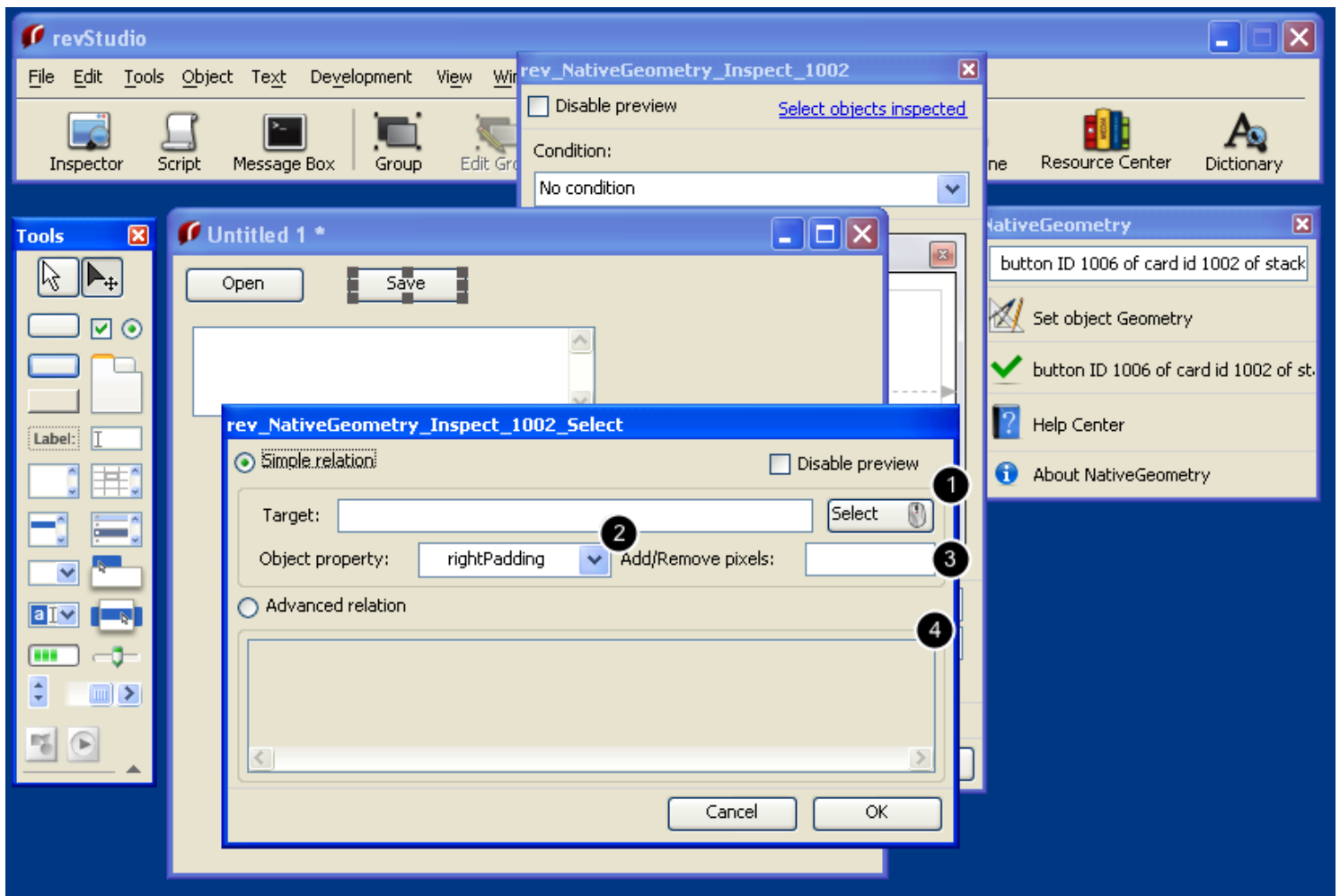
## Set the Geometry of the "Save" Button



We want to have the left of the "Save" button to be at the right of the "Open" button, with a space between them.

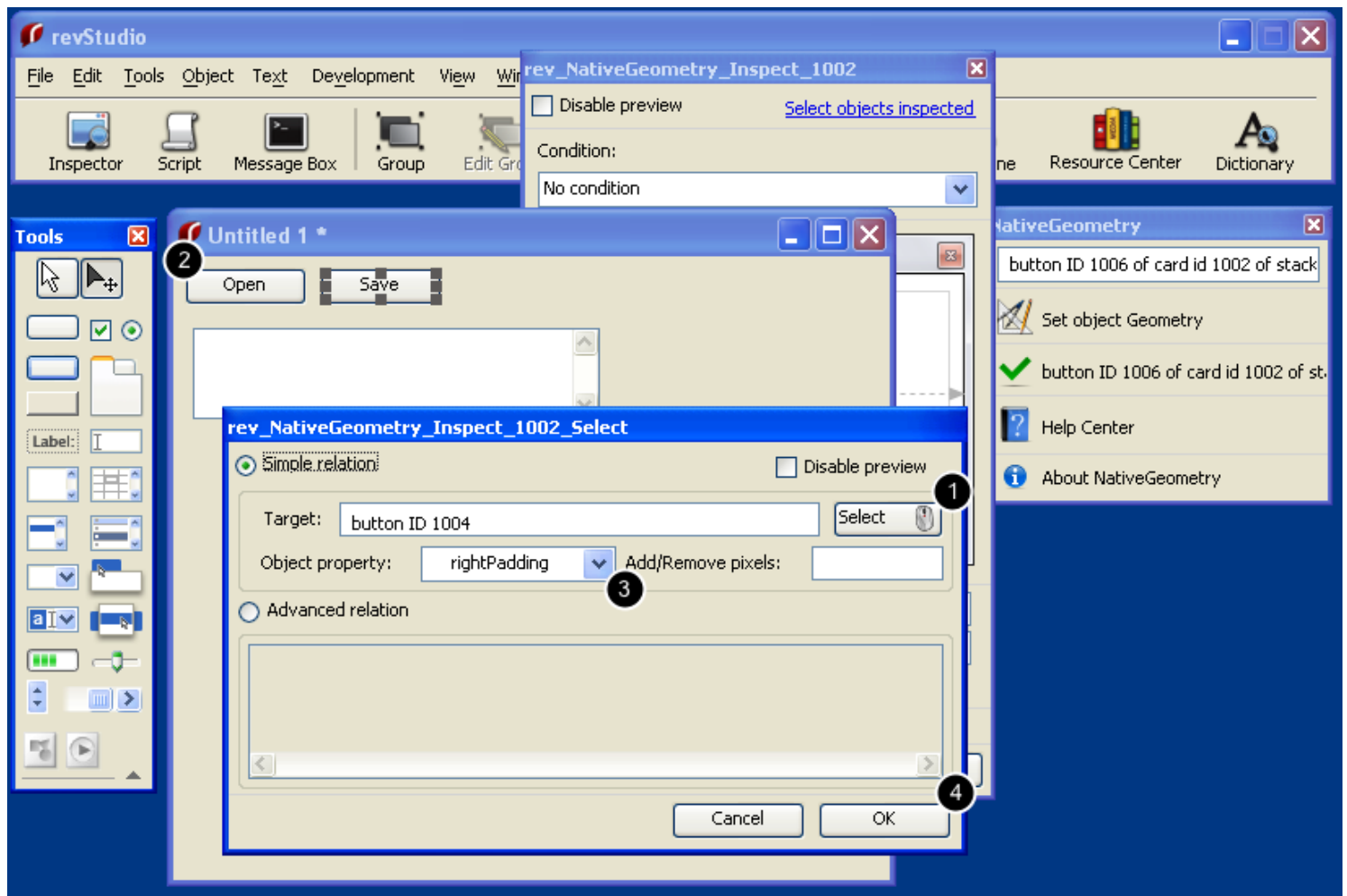
So, select the "Save" button (1), then click "Set object Geometry" (2), then click the button (3).





Now you have a window that enables you to do one of two things:

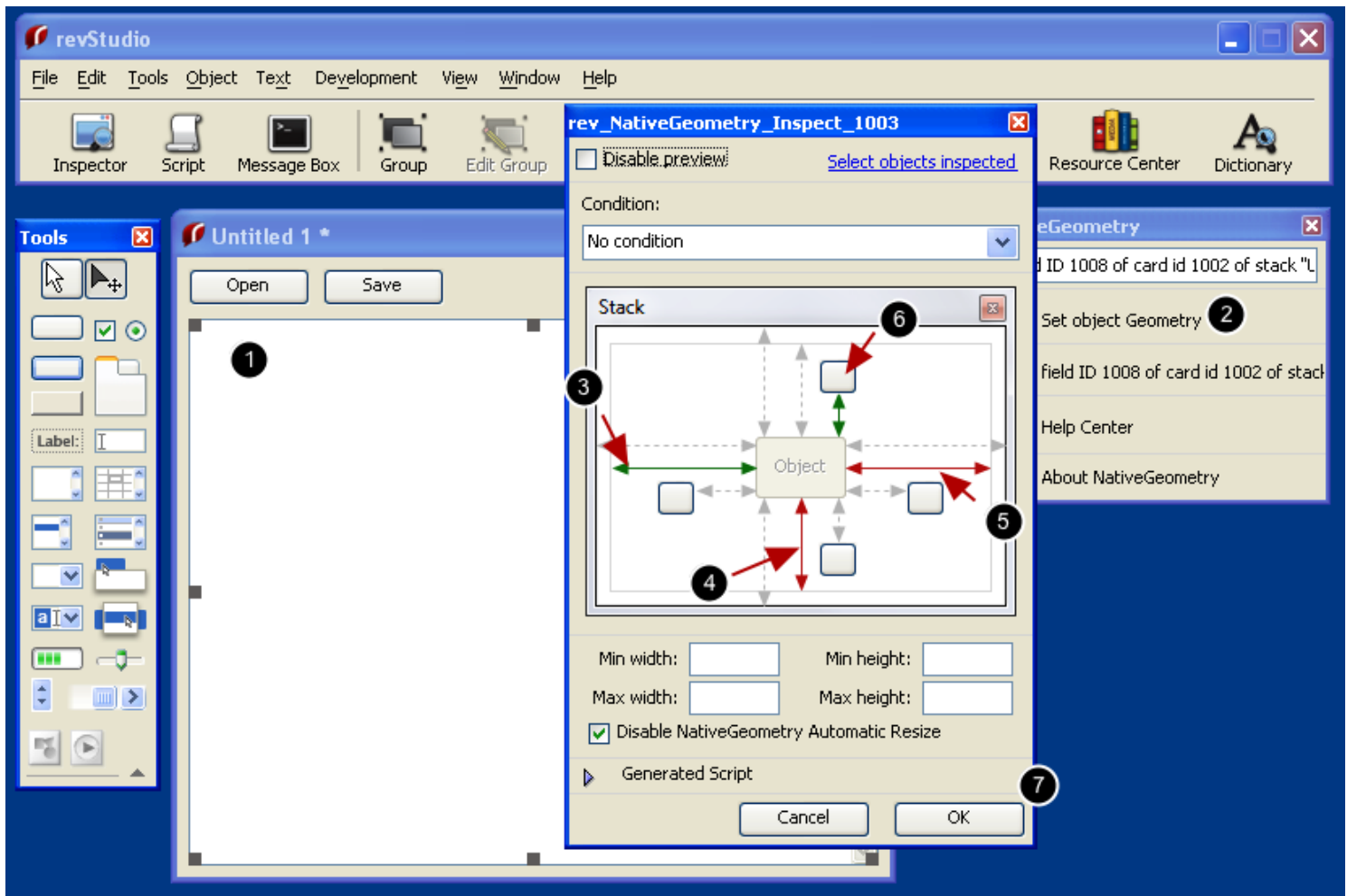
- Select an object (1), read a property of the selected object (2) and add/remove pixels manually. (3)
- Or write directly an advanced geometry relation, for example, 'the height of this stack / 2 + the left of button "toto" + 6'. This will be covered in another tutorial.



Click "Select" (1) then click the button "Open" (2), check the property read (3), finally click "OK" to close this window and click "OK" also in the NativeGeometry inspector.

Now, you have the "Open" and the "Save" buttons that have their geometry correctly defined. Let's set the geometry of the field now.

## Set the Geometry of the Field

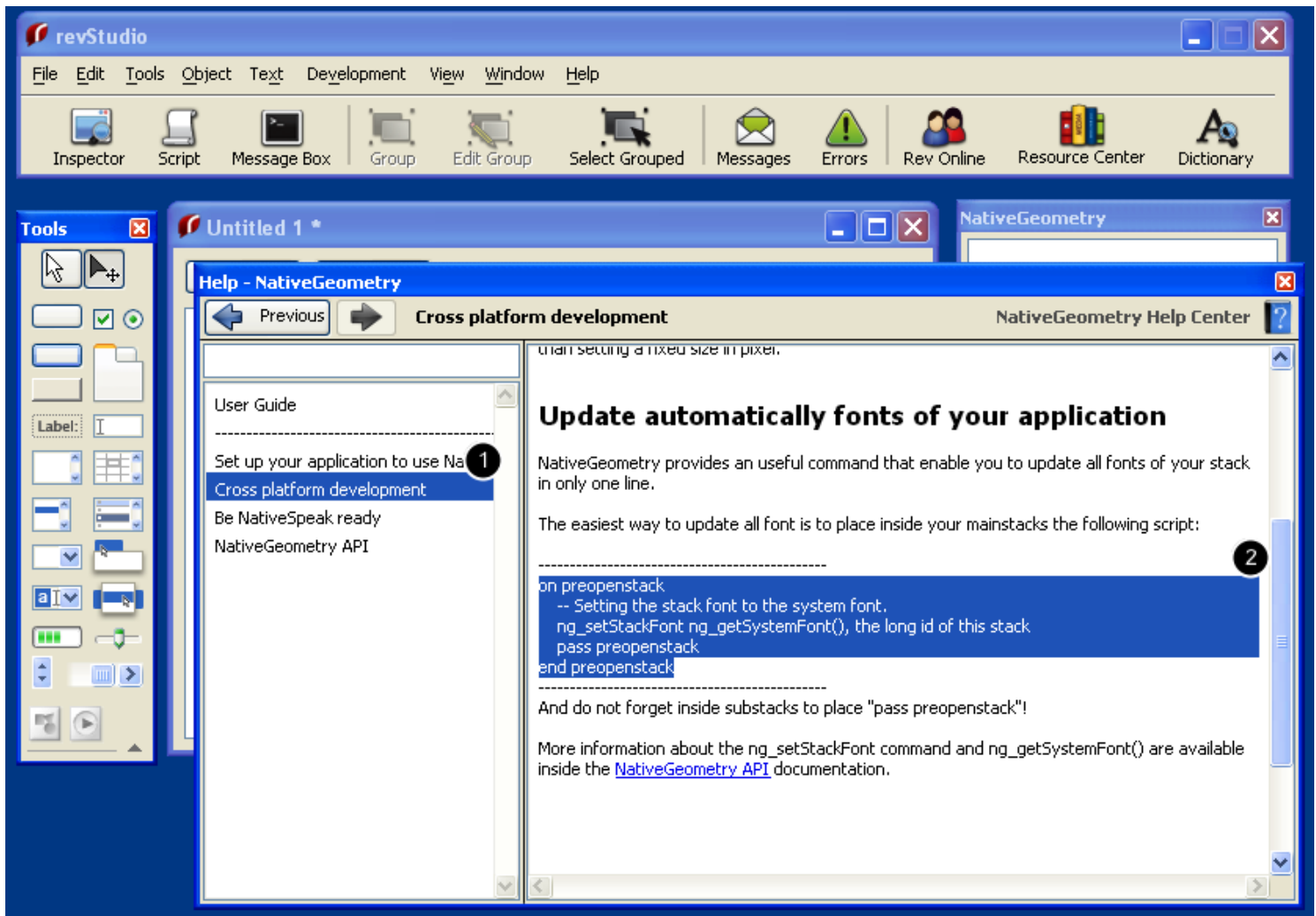


Now that you know how the property inspector works, it should be a straightforward quick task.

- Select the field (1)
- Click "Set object Geometry" (2)
- Set the left of the field to the left padding of the stack (3)
- Double click (4) in order to set the bottom of the field to the bottom padding of the stack
- Do the same for the right (5)
- Click (6) and select the button "Open"
- Click "Ok" (7) to apply the new geometry.

That's all, the field is now resized!

## The Last Touch to have a Perfect Application

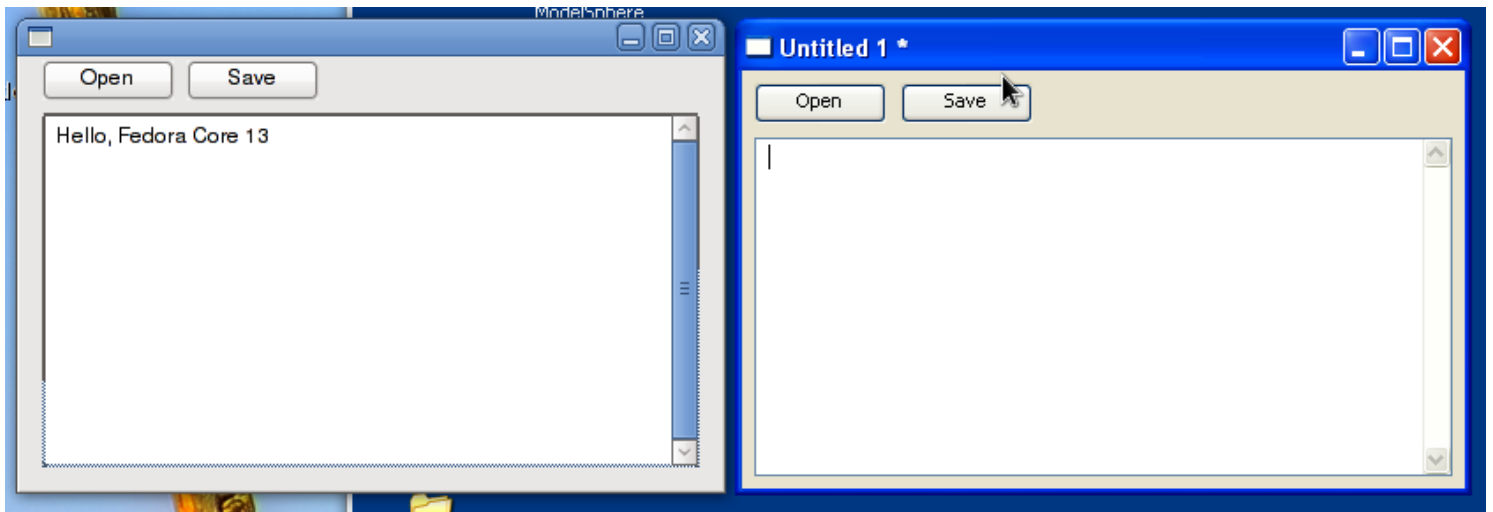


In order to have a perfect application, NativeGeometry enables you to automatically set the font of your application.

To enable that feature, open the NativeGeometry help center, then click "Cross platform development" (1), then copy the highlighted code snippet (2) and paste it inside your main application stack. ("Untitled 1" in this tutorial).

Now, no matter what the platform, your application will always have a good font, and objects will be good looking.

## The Final Application



Save the stack, configure the standalone builder then simply build your application and launch it.

This is the end of the first NativeGeometry tutorial, it showed you how to use the NativeGeometry inspector in order to make quickly a good looking GUI with NativeGeometry.

The next tutorial will be focused on the NativeGeometry API.

Kind Regards,

Damien Girard

<http://www.nativesoft.fr>