



LEARNING VRHOTWIRE'S SCRIPTING LANGUAGE:VRNACULAR

version 1.0

GETTING STARTED SCRIPTING WITH VRNACULAR

In the beginning there is the panorama.

You finally figure out how to make a nice one, and put it on the web.
You make a few, perhaps many. Then you want to add more features
to your movie. Your client has some ideas...

But you're a photographer, not a programmer!
At first you're happy using vrhotwires 'tools' menu items to generate scripts for you.
But code that's generated automatically has limitations.

You want a more custom look and feel to your movie.
Handcrafted in every detail...

After a while you start looking at those auto-generated scripts made by the tools and trying
to understand them.
(by going to 'script window' under views (command r))

and then you start trying to change the odd thing...

you learn how the 'scrip to movie' buton works...

Now you want control over this scripting thing!

That brings you to learning VRNacular.

There's another document in this folder that acts like a dictionary of VRNacular,
here we walk you through the scripting process...

Hope you find it easy!

DESIGN GOALS:

Let's start by writing down what we want our movie to do:

"I want my movie to pan when I press a button at the bottom"

"I want my movie to play the sound of the dog barking three times, then pan over to the bad guy, and play the linear transition of the bad guy doing a flip"

"I have a pano of a park, and I want a girl riding a bike to ride through a couple times every minute"

With that as our basis, we can derive a script that will do what we're trying to achieve.

Eventually scripting languages might evolved that allow you to use descriptions like these... for now we' ll focus on translating these goals in spoken english into the VRnacular language....

VRNACULAR BASICS

Before we try to solve one of these goals, let's learn how to use vrh scripting at all!

Open vrhotwires

drag your vr movie onto the white window to open it.

Press 'command R' or choose 'script' from under the views window.

Copy and paste this script into that window:.

```
On Idle  
  SetPanAngle 3 delta  
end
```

Now press the 'script to movie' button.

Your movie should be autopanning. (if it isn't make sure it has a qtvr track. If not, update it to qtvr 2.0 using conVRter from vrtools.com)

Now, do you know how to speed it up? Make it spin faster?

Make this the script: (change the 3 to a 7)

```
On Idle
    SetPanAngle 7 delta
end
```

Press "Script to movie" to see the increase in speed of panning.

If you press the "Movie To Script" button you'll get...

```
On Idle
    QTVRSetPanAngle 7 delta
end
```

This is because the real call is QTVRSetPanAngle but a few commands can be typed in using abbreviated names... like SetPanAngle SetTiltAngle SetVar etc.

If the middle line was shortened to read:
SetPanAngle 7
the movie would go to 7 degrees and stay there.

But we've said 'delta' which means
'add 7 to the existing pan angle'.

By increasing the delta we make the pan faster.
and how do I make it go the other way?

```
On Idle
    SetPanAngle -3 delta
end
```

Hey, it doesn't work right. The movie stops panning when it gets to zero!!

That's because qtvr doesn't want negative numbers. You need to make it wrap.

Try this instead:

```
On Idle
    SetPanAngle neg3 delta wrap min=0 max=360
end
```

Hmm, min and max huh? so does that mean I can make it just go half way and then start again from zero?

```
On Idle
    SetPanAngle neg3 delta wrap min=0 max=180
```

end

cool....

Now what's this other window here?

SCRIPT SYNTAX WINDOW

The window on your left when you are scripting is the script syntax document.

The first thing you see is a list of triggers:

Most of them work on sprites and hotspots:

On MouseDown Hotspot 9
On MouseDown Sprite 1

As you scroll down you see the actions, operands and operators.
These will be great reference items as we learn to script.

----- glossary:

Those three words, 'actions,operands and operators' scared me a bit when I first started wiring, along with another 'expression'.

Here's a quick glossary:

action: Something your movie does, like panning, or moving a sprite.

trigger: a way to trigger an action like On MouseDown. the Idle trigger just does the action over and over...

operand: a special number qt knows, like the current pan angle, or the current time, or the value of a variable.

operator: one of the math symbols like + - * /

expression: a 'phrase' in vrnacular. like variable[1]+variable[2] it is usually made up of the grammar: operand-operator-operand

You'll notice that 'SetPanAngle' isn't on the list of Actions.

The official list is strictly based on apple code. We find it easier to have a standard version that way.

But we also allow for synonyms. That is, an easier way of typing something in:

For example, instead of saying

SpriteTrackSetVariable you can just say

setvar

When the script writes out it will say

SpriteTrackSetVariable but it makes it easier to type in. Also, the original vrhotwires language used SetPanAngle, so we accept it...

(did you know you can add your own synonyms! see the synonyms folder for more info. That kind writes out the way you set it too!)

Solving for goal 1:

"I want my movie to pan when I press a button at the bottom"

Let's just use the watermark tool to add the sprite right now, as the interface is still changing...
Add the sprite...

Now, first let's try these sprite click things:

copy this script:

```
On MouseDown Sprite 1
```

```
    SetPanAngle 99
```

```
end
```

hmm that seems to work pretty well. When I click on the sprite, I pan to 99. When I click again, nothing happens because I'm already at 99.

If I press command 'h' and look at the realtime info in vrhotwires, I can see I'm at 99.

But my sprite is a little arrow. I wanted it to pan while the little arrow is clicked. Then when I release the mouse it can stop panning.

Well this calls for a variable!

On MouseDown Sprite 1

```
    SpriteTrackSetVariable 1 1 tracktype=sprite  
end
```

On MouseUp Sprite 1

```
    SpriteTrackSetVariable 1 0 tracktype=sprite  
end
```

On Idle

```
    if variable[1]=1  
        setpanangle 3 delta  
    endif  
end
```

You don't have to define variables in VRNacular before you use them.

Here, we toggle the value of variable 1 when the mouse is clicked or not clicked.

Then on idle, we check to see what the variable is set to before we decide to pan.

EXPRESSIONS

I remember when I was a kid and I used some new phrase I learned at school my father would say 'what an odd expression'.

Well expressions are kind of phrases in vrnacular too.

The script syntax doc gives a list of actions...

and each of those actions has parameters.

So far we've only handed in constants(numbers) as parameters (like in our SetPanAngle 3 call)

But anywhere that takes a number can take an expression too!

Try this one:

```
On Idle  
    SetPanAngle thePanAngle+1  
end
```

This should get your pano spinning too!

And thePanAngle is an operand. You used an operand !

Operands can be thought of as reserved values. Imagine that quicktime has this great big room full of cubbyholes where it keeps certain values. Like the current date, or the current pan angle, or a sprite's position...

There's a list of operands in the script syntax document.

Important: In vrnacular 1.0 you cannot have any space in your expressions.

ie: you must say `SetPanAngle theMovieTime+37` and not `SetPanAngle theMovieTime + 37`

VARIATIONS ON GOAL 1

so what are some variations on this panning we've been doing when we click our sprite?

Well, wherever we've used pan we can replace it with FieldOfView. This turns our button into a zoom button:

On MouseDown Sprite 1

```
SpriteTrackSetVariable 1 1 tracktype=sprite  
end
```

On MouseUp Sprite 1

```
SpriteTrackSetVariable 1 0 tracktype=sprite  
end
```

On Idle

```
if variable[1]=1  
    QTVRSetFieldOfView 1 delta  
endif  
end
```

And what about something with the sprite, let's give it a down state and up state...
When we click it, we'll make it shrink to 7/10ths of it's usual size...

On MouseDown Sprite 1

```
SpriteTrackSetVariable 1 1 tracktype=sprite  
SpriteScale 0.7 0.7 tracktype=sprite spriteindex=1
```

end

On MouseUp Sprite 1

```
SpriteTrackSetVariable 1 0 tracktype=sprite  
SpriteScale 1 1 tracktype=sprite spriteindex=1
```

end

Note that vrnacular seems to not like just .7 right now, but prefers if you enter 0.7

Or we could move it a bit when we click down:

On MouseDown Sprite 1

```
SpriteTrackSetVariable 1 1 tracktype=sprite  
SpriteTranslate 5 5 false tracktype=sprite spriteindex=1
```

end

On MouseUp Sprite 1

```
SpriteTrackSetVariable 1 0 tracktype=sprite  
SpriteTranslate neg5 neg5 false tracktype=sprite spriteindex=1
```

end

Hmm... that one doesn't look too good, the track is too small.
You can resize your track from the movie info window.

We'll come back to sprite motion...

although you might want to try:

On Idle

```
SpriteRotate 5 tracktype=sprite spriteindex=1
```

end

just to see that crazy spinning....

There are over 100 wired actions and dozens of operands and operators... learning to script is a fairly long and complex task. But vrhotwires gives you a special advantage in that you can analyze the script of any existing wired movie.