

Get-View and New-Object - how does it ...



rsj 9 posts since

Jan 21, 2008 I'm a bit desperate now - i have been trying to figure the framework options out, using get-view commands, and the API SDK reference site.

As an exercise for myself, i set out to learn how to change linkspeed on a pnic, using the get-view commands. My problem is that i have no idea about where to find those informations, im not sure how to navigate the SDK, and when to use new-object.

Here is an example of how far i get:

```
$HS = Get-View -ViewType "HostSystem" -Filter @{"Name" = "ESXSERVER"}.ID
$mod = $HS.ConfigManager.NetworkSystem
$networkConfig = New-Object VMware.Vim.HostNetworkConfig
$networkconfig
```

The \$networkconfig turns out empty (I would expect to see som kind of info on what is included in the HostNetworkConfig), and im not sure how to drill down to choosing the pnic i want to make the changes to.

When writing \$HS value to the screen, i can drill down thru the different options, but how do i know when to use New-Object to define settings?

Im pretty much grasping at straws here...

I would like to get an understanding of the methods beyound using cmdlets, but im kinda stuck...



halr9000 813 posts since

Jun 7, 2007 1. **Re: Get-View and New-Object - how does it work?** Oct 9, 2008 4:45 PM

Wow, big question. It is probably deserving of a blog post (or hey, a book! 😊).

First, briefly:

Cmdlets are the highest level of interface with .NET objects which PowerShell provides. VMware knows more or less what VI admins need to do their jobs. They have a goal of enabling admins by creating cmdlets for as many usage scenarios as they can. Time and money of course are limiting factors, and so version 1 came with a certain set of cmdlets. It was seen that this wasn't good enough, and that there needed to be a way to access the full SDK to do the low level stuff for which they did not yet have time to fill out with cmdlets. Thus, Get-View.

Get-View deals with server-side objects represented by the VI web service (and documented in the SDK). With it, you can do anything that the VI Client can do, and more. It's more complicated to use, of course, and in fact, it's not a lot simpler than writing a "full" program using a language like C#. But it's not bad, relatively speaking.

You can pass several different kinds of "automation objects", those which live in your PowerShell session, to Get-View and voila, you now have a new object which has a richer set of methods and properties. However, this only works for "managed objects" (and not directly for all of them), which are only one type of object represented in the SDK. This leaves "data objects" totally left out, and they are important, too.

Get-View and New-Object - how does it ...

New-Object has nothing to do with the VI Toolkit, it's a cmdlet which comes with PowerShell. Its used to give you the ability to access a .NET (or COM) object which is stored in an assembly (fancy name for a DLL). One thing you can do with it is to create these data objects. The data objects are often used as specifications to methods. This basically works like a giant bundle of arguments to perform a single action.

And that's all I have time to say right now. 😊

Author of the upcoming book: [Managing VMware Infrastructure with PowerShell](#)

Co-Host, PowerScripting Podcast (<http://powerscripting.net>)



[alanrenouf](#) 582 posts since

Feb 28, 2006 2. **Re: Get-View and New-Object - how does it work?** Oct 10, 2008 12:40 AM

👤 in response to: [halr9000](#) Nice explanation Hal, what is the best way to start learning how to use the SDK information ?



[hugopeeters](#) 155 posts since

Jan 10, 2008 3. **Re: Get-View and New-Object - how does it work?** Oct 10, 2008 1:17 AM

👤 in response to: [alanrenouf](#) Through examples of course!

I just completed a blog post that used quite a bit of searching the API reference. It's about changing the License Server setting in Virtual Center through the VI Toolkit. Read it here:

<http://www.peetersonline.nl/index.php/vmware/changing-your-vmware-license-server/>

Hugo



[alanrenouf](#) 582 posts since

Feb 28, 2006 4. **Re: Get-View and New-Object - how does it work?** Oct 10, 2008 1:20 AM

👤 in response to: [hugopeeters](#) Looks like a back read of your articles will help me get started 😊 Great work !



[hugopeeters](#) 155 posts since

Jan 10, 2008 5. **Re: Get-View and New-Object - how does it work?** Oct 10, 2008 5:09 AM

In addition to what Hal said, here's some steps for figuring out how to use a method:

1. Finding the method you need: Use Get-Member on an object you obtained by using Get-View. As al said, Get-View returns an object with many methods and properties. Get-Member will show all of these. E.g.:
`$VMHostView = Get-VMHost HOSTNAME | Get-View ; $VMHostView | Get-Member`
2. Finding the syntax of the method: Type it without brackets, e.g.: `$VMHostView.Rename` ; It will show some info including the OverloadDefinition, e.g.: `System.Void Rename(String newName)`
Ignore the System.Void, note that String is the type of the parameter and newName is the name of the parameter. You could search for the method in the API reference for more info on the parameters e.g. their properties.

Get-View and New-Object - how does it ...

3. Create the parameter objects. In the example, a string is easy to create, just use double quotes.(\$param = "newname") In other methods it might state LicenseSource or any other kind. For VI Toolkit stuff, use \$param1 = New-Object VMware.Vim.TypeName (e.g.: VMware.Vim.LicenseSource). When looking at your new object, it will already have all properties, but with no values. Set the values you want and then use the object as input for your method (e.g.: \$VMHostView.Rename(\$param))

If you use the wrong type of object or the incorrect amount of parameters, read the error message carefully. It will tell you what went wrong.

Hope this helps.

Hugo

www.peetersonline.nl



rsj 9 posts since

Jan 21, 2008 6. **Re: Get-View and New-Object - how does it work?** Oct 13, 2008 12:50 PM

👤 in response to: [hugopeeters](#) Thx halr and hugo! Im still learning to navigate the SDK, but the understanding of the get-view and its uses are getting closer.

I have to say that the examples in your blog are great hugo!

Also im looking forward to the book halr 😊

One more question:

Im trying to use the set-harddisk to increase vmdk capacity. My problem is that i can't return the correct value for the "-Harddisk" parameter.

This is the error i get:

Cannot convert value "1" to type "VMware.VimAutomation.Types.HardDisk"

Im guessing i have to retrieve the harddisk value somewhere, in a format that corresponds to the Types.HardDisk, but how do i know where to get that value in the correct format? I cannot seem to find the Types.HardDisk in the SDK.

Again thx for quick and on the spot feedback!