

How to create a Script-based 2-State Monitor

Leveraging the scripting runtime for state-based monitoring in System Center Operations Manager and Essentials 2007

Author:

Pete Zerger, MCSE (Messaging) | MCTS (SQL 2005) | MVP-MOM

Version History:

Version: 1.1

Published September 2009

Version: 1.0

Published August 18, 2008

Some Rights Reserved: You are free to use and reference this document and it's, so long as, when republishing you properly credit the author and provide a link back to the published source.

Introduction

This tutorial describes how to create a Timed Script Two State Monitor in step-by-step detail, using sample scripts available for download from System Center Forum. The intended audience is Operations Manager 2007 administrators with at least a basic understanding of the Operations console.

We hope this tutorial serves to improve your understanding of how to implement script-based monitors in Operations Manager 2007 to monitor the availability, performance and configuration of your monitored systems.

Enjoy!

Implementation Steps

This section details the implementation steps for a Timed Script Two State Monitor in Operations Manager 2007 or Essentials 2007

1. Browse to the Monitoring space. Right click **Monitors** and select **Create a Monitor → Unit Monitor**.
2. On the **General Properties** screen, browse to **Scripting → Generic → Timed Script Two-State Monitor**.

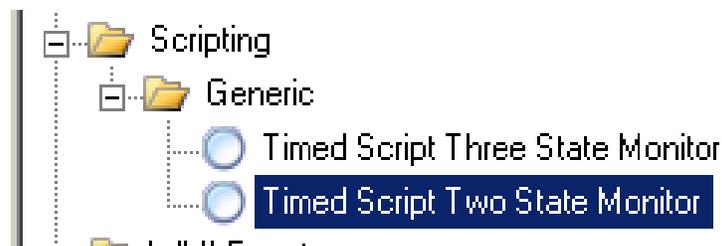


Figure 1 - Timed Script Two State Monitor

3. In the **Select a destination management pack** dropdown, choose the custom management pack where the monitor should be stored. Click **Next**.

TIP | *MS best practices suggest never storing your custom monitoring in the Default Management Pack.*

4. In the **Name** textbox, enter the following name for the monitor: '2-State Process Monitor'.
5. To the right of the **Monitor Target** box, click the **Select** button. Type or browse to **Windows Server 2003 Operating System**.
6. In the **Parent Monitor** dropdown, accept the default of **Availability**.
7. De-select the **Monitor is Enabled** checkbox.

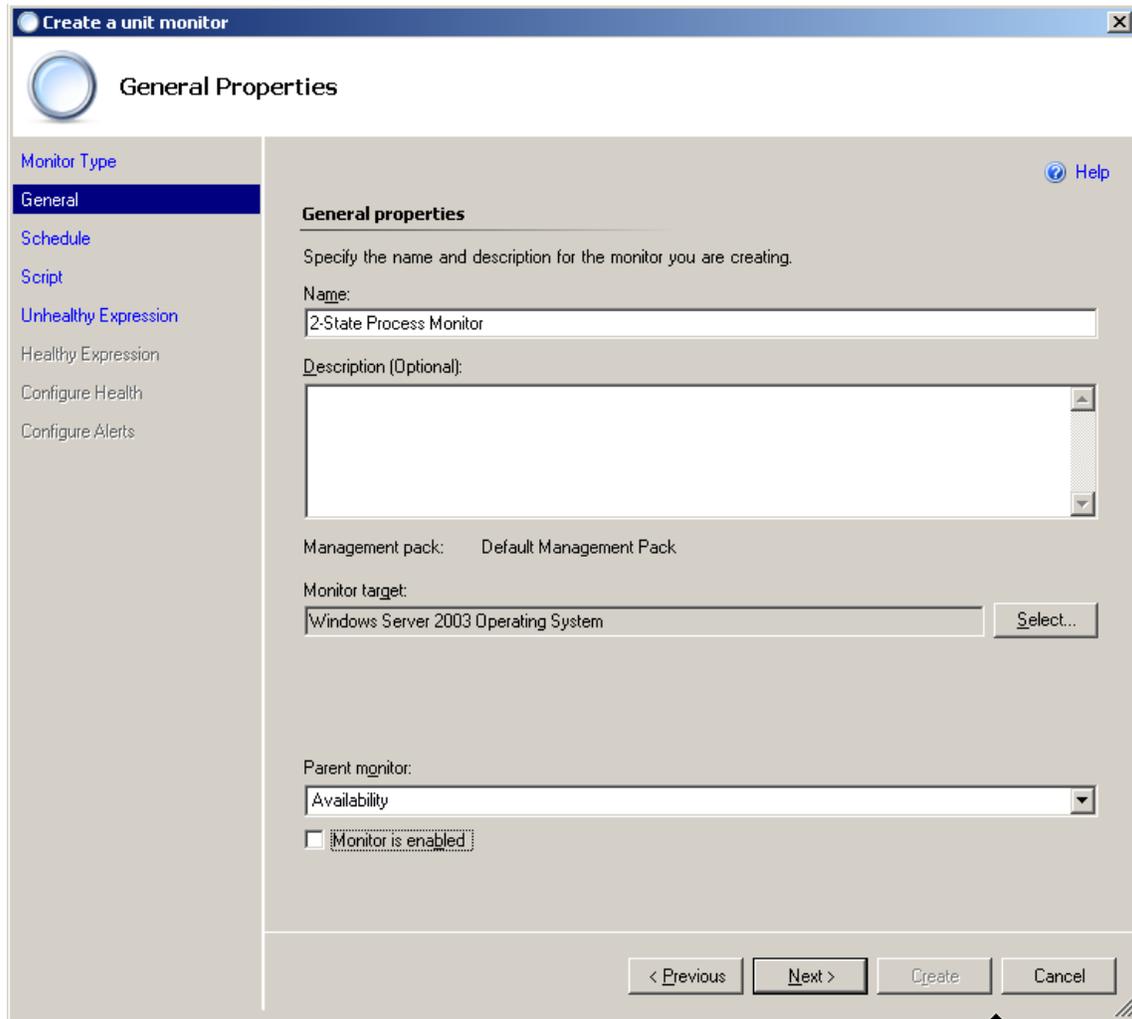


Figure 2 - General Properties of Unit Monitor Wizard

8. On the **Schedule** screen, set the desired run interval for the script and click **Next**. While every 2 minutes is great for testing, a run interval of between ever 5 and 15 minutes is probably more suitable for your production environment.

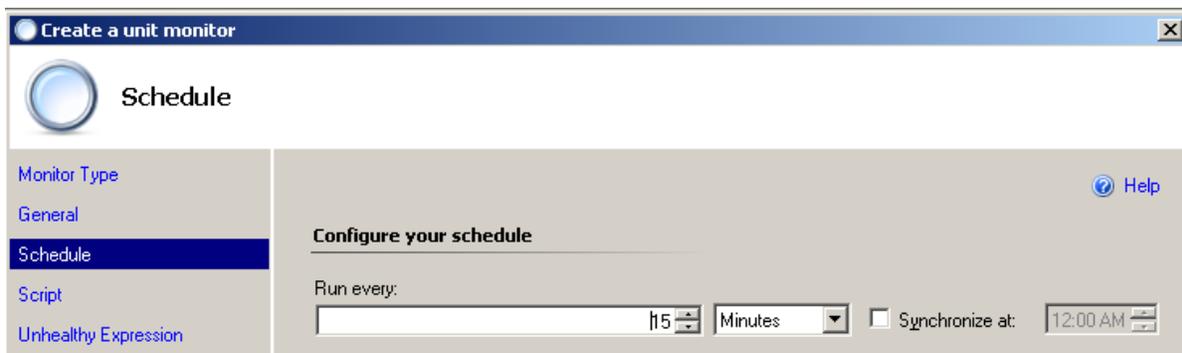


Figure 3 - Schedule Screen in the Unit Monitor Wizard

9. On the **Script** screen, paste in your script into the Window provided as shown in **figure 4**. In this case, you can use any of the sample 2-state monitor scripts at this URL:

<http://www.systemcentercentral.com/BlogDetails/tabid/143/IndexID/19269/Default.aspx>

You'll find a script for checking a registry value, and another for checking for a running Windows process. ***In this example, we will use the script that checks for a running Windows process (click [HERE](#) to retrieve).*** This sample script looks for Notepad.exe, so be sure to read the instructions in the script header and set the process name to the value of your choice.

If the process is found, the script returns a value of GOOD, and if not, returns a value of BAD.

2-State Monitor Scripts and the PropertyBag

2-state monitor scripts use an element of the Operations Manager 2007 scripting runtime called the **PropertyBag**. A PropertyBag is an object that holds name value pairs that can be used to indicate state (as in this example), hold event data, or even create custom performance data that can be represented in a Performance View in the Monitoring space. *Find these values below in the sample process monitoring script to familiarize yourself with the how to use the PropertyBag!*

'Instantiate OpsMgr scripting runtime and create PropertyBag

Set oAPI = CreateObject("MOM.ScriptAPI")

Set oBag = oAPI.CreateTypedPropertyBag(StateDataType)

'Adding a GOOD or BAD value to the PropertyBag

Call oBag.AddValue("State","BAD")

Call oBag.AddValue("State","GOOD")

For more information on the ways you can use a PropertyBag in your OpsMgr runtime scripts, click on the hyperlink below for following article at System Center Forum:

[Using PropertyBags with Custom Scripting In Operations Manager 2007](#)

10. Be sure to give the script a name that is unique to this monitor and descriptive of its function, as shown in **figure 4**. Should you later have problems script execution on an agent managed computer, the name will appear in an event in the Operations Manager Event Log.
11. Ignore the **Parameters** button in this case, as it is unnecessary in our example. When complete, click **Next**.

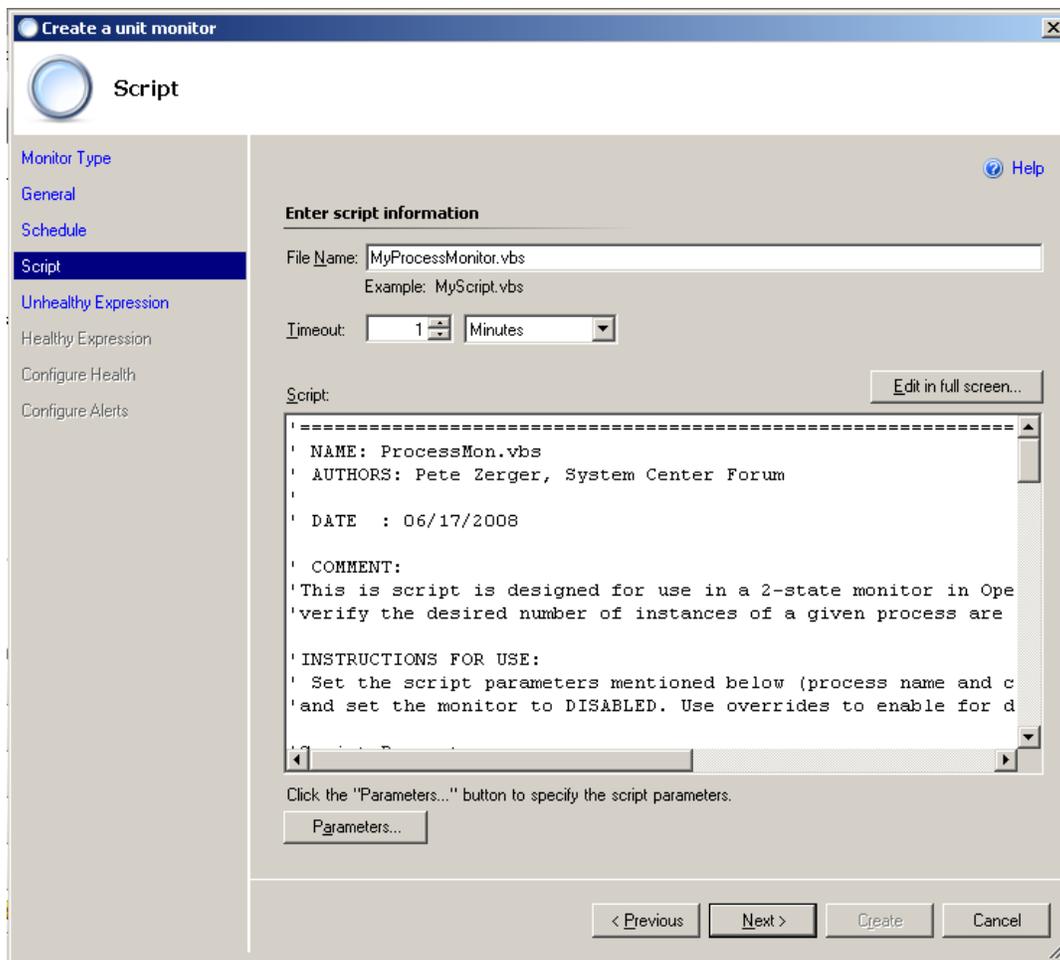


Figure 4 - Script Screen in the Unit Monitor Wizard

- On the **Build Event Expression – Unhealthy Expression** screen, click the **Insert** button to insert an expression for determining when the script has identified that the monitor is in an unhealthy state as shown in **figure 5**. Enter the values below into the expression fields.

Parameter: Property[@Name='State']

Operator: Equals

Value: BAD

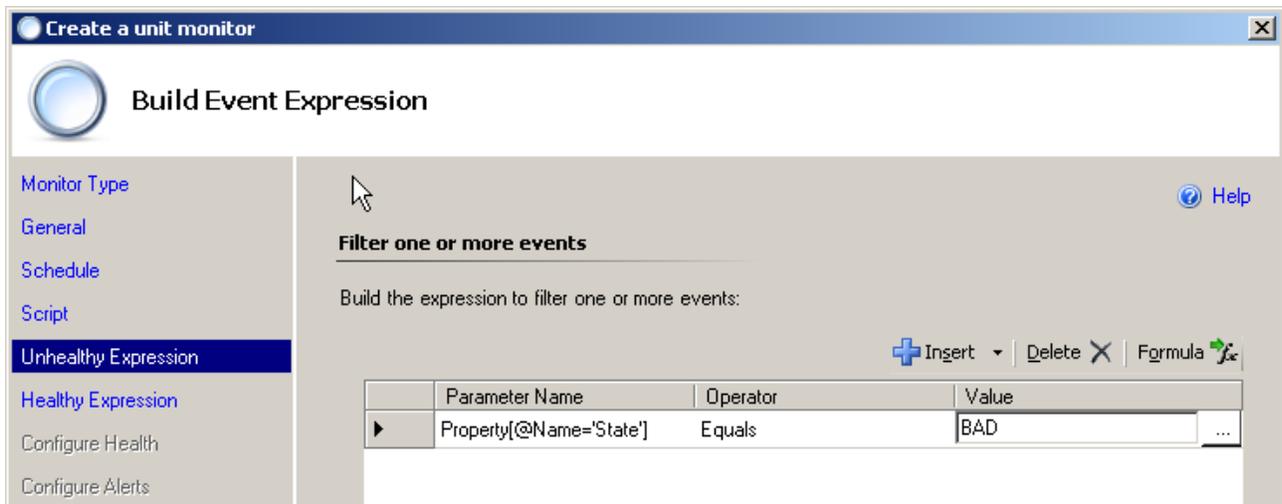


Figure 5 - Unhealthy Expression Screen in Unit Monitor Wizard

- On the **Build Event Expression – Healthy Expression** screen, click the **Insert** button to insert an expression for determining when the script has identified that the monitor is in a healthy state as shown in **figure 6**. Enter the values below into the expression fields.

Parameter: Property[@Name='State']

Operator: Equals

Value: GOOD

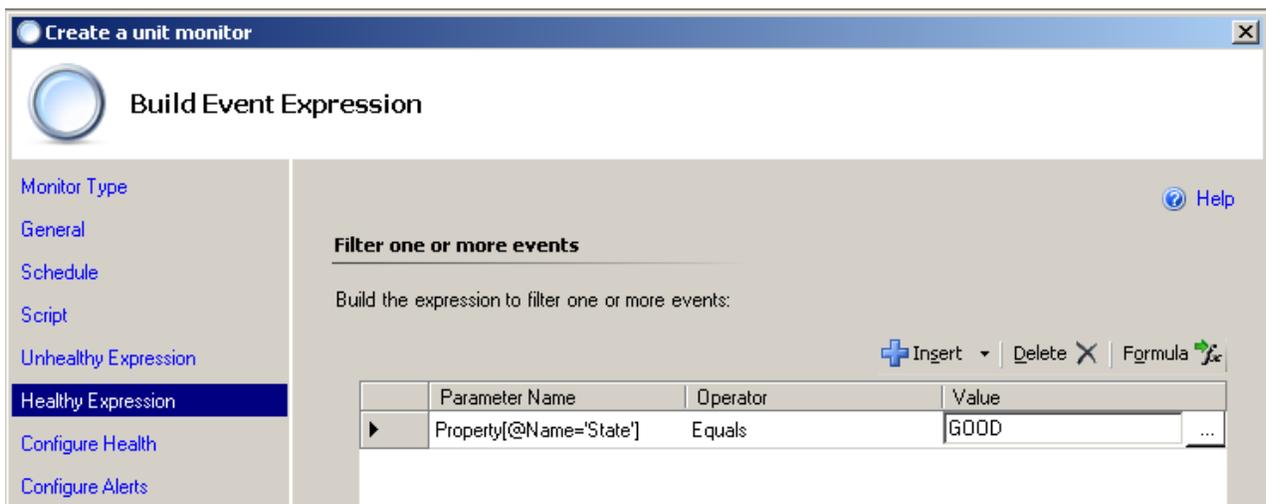


Figure 6 - Unhealthy Expression Screen in Unit Monitor Wizard

14. On the **Configure Health** screen, we define the severity of the unhealthy state. This will be either Warning or Critical (using the dropdown provided), depending on the severity of the situation for which you are monitoring. When complete, click **Next**.

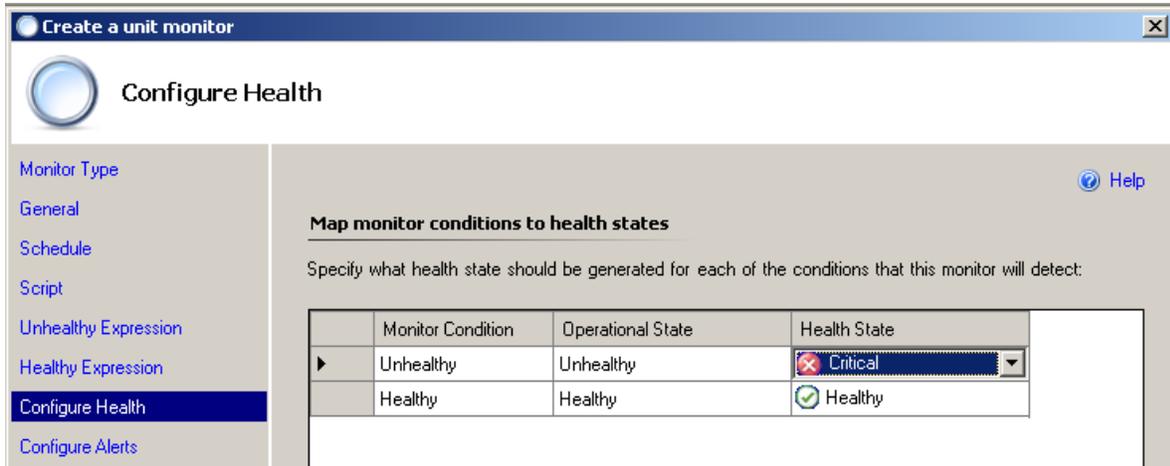
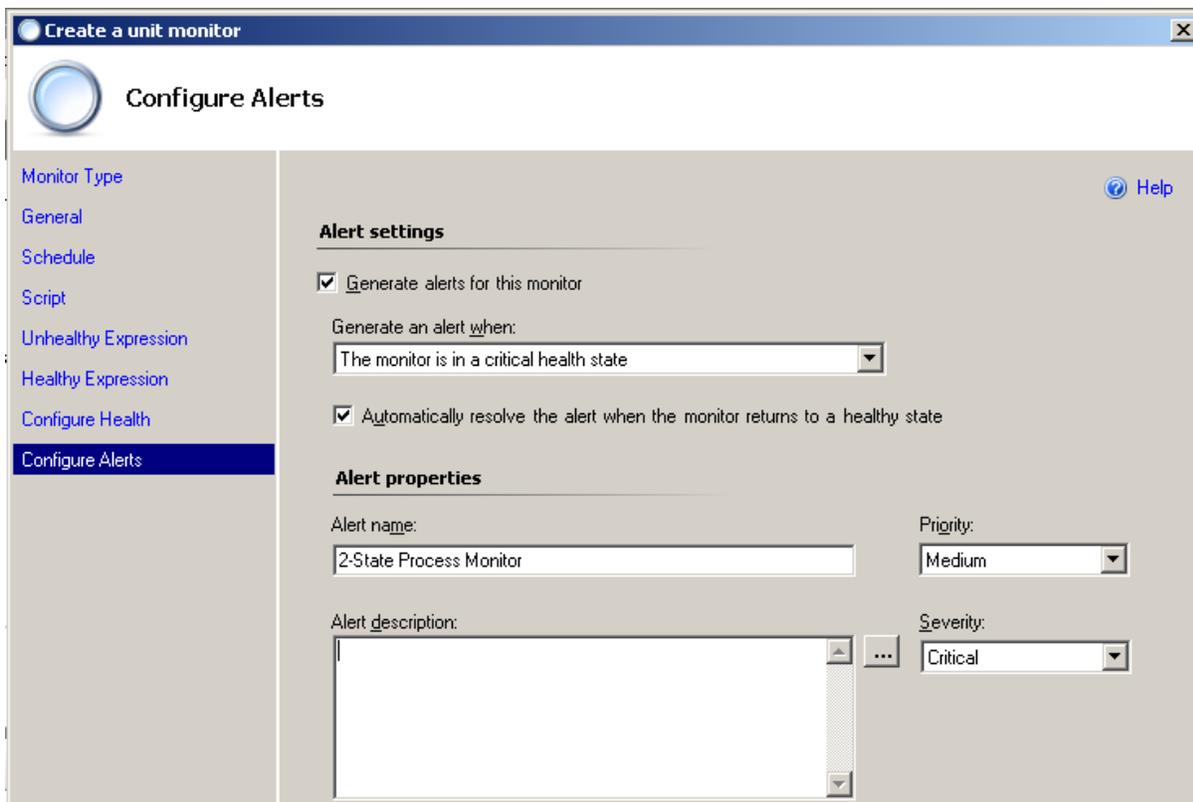


Figure 7 - Configure Health Screen in Unit Monitor Wizard

15. On the **Configure Alerts** screen, check the Generate alerts for this monitor box, if you do want to generate alerts when this monitor goes into an unhealthy state. You may want to add some custom text in the **Alert description** field.



16. When complete, click the **Create** button.

17. Now, create an override to enable the monitor for the agent managed computers that should execute the script.

This completes the necessary configuration for your 2-State Script-based Unit Monitor. We need simply wait for the updated management pack containing the monitor to be received by the agent.

Result

The script should generate an alert whenever the notepad.exe process is not detected, and return to a healthy state when it is detected. Simply launching Notepad on the target agent computer will cause the alert to be resolved and the monitor to return to a healthy state. In this case, unless you already have an instance of Notepad running on the target system, the monitor may begin in an unhealthy state.

You can find a few examples of additional 2-state monitor scripts in the sample management packs on the System Center Central website.

Feedback

Your comments and feedback are welcome at administrator@systemcentercentral.com