AuthorMPs.com

Sample MP

Console Tasks

Steve Wilson

3/27/2007

This guide details the contents of the sample console task management pack (AuthorMPs.Demo.ConsoleTasks).

Contents

Introduction

This management pack shows you how to create console tasks. There are three types of console tasks available and I cover each one:

- Monitoring Object based task
- Alert based task
- Event based task

Monitoring object based tasks are targeted to a specific class. Alert and event based tasks are not target specific and will be available against any alert or event.

Console tasks execute on the machine you are running the Operations Console on. They are not sent to the agent to execute like agent tasks are. Typically they are used to launch some interactive application using the context of the object you run the task against.

Discovery

The MP defines a single class:

- AuthorMPs Demo - Application X

To discover an instance on this class on a monitored server you must create the following registry key:

- HKLM\AuthorMPs\ApplicationX

Also there must be two string values in this key:

- Version: Set to something like 3.0
- Path: Set to something like C:\ApplicationX

You can check that an Application X instance is discovered correctly using the state view that I provide.

Views

There are three views in this management pack:

- Application X State – a state view showing all instances of Application X
- Application X Alerts – an alert view showing all alerts for all instances of Application X
- Application X Events – an event view showing all events for all instances of Application X

Rules

I have provided two rules in this management pack, one to generate an event and one to generate an alert. These are both targeted to the Application X class and allow you to populate the database with some event and alert data to try out your console tasks.

To generate an event against an Application X instance type the following at a command prompt on a server that you have discovered Application X on:

EventCreate /ID 501 /T information /D “This is some event description”

To generate an alert:

EventCreate /ID 502 /T information /D “This is some event description”

Console Tasks

There are four console tasks included in this MP showing a variety of principles.

AuthorMPs.Demo.ConsoleTasks.ApplicationX.ComputerManagement

Target: Windows Server

This task is targeted at Windows Servers. It will run the computer management MMC and connect to the server you launch the task against.

You can run this task from the Computers view in the Operations Console. Switch to this view, locate a Windows Serve you are monitoring, right click and select **Windows Computer Tasks, AuthorMPs – Computer Management**. This will launch the computer management snap in. Notice in my definition how I use pass the network name of the computer as context:

```
<Application>%windir%\system32\mmc.exe</Application>
<Parameters>
      <Parameter>compmgmt.msc</Parameter>
      <Parameter>/computer:$Target/Property[Type="Windows!Microsoft.Windows.C
omputer"]/NetworkName$</Parameter>
</Parameters>
```

Notice that I set the **RequireOutput** attribute to false. You should do this when you are launching an interactive application so Operations Manager does not leave an output window open.

Notice also that I set the **Category** attribute to **MonitoringObject**. This states that the task will run against a monitoring object not an alert or event.

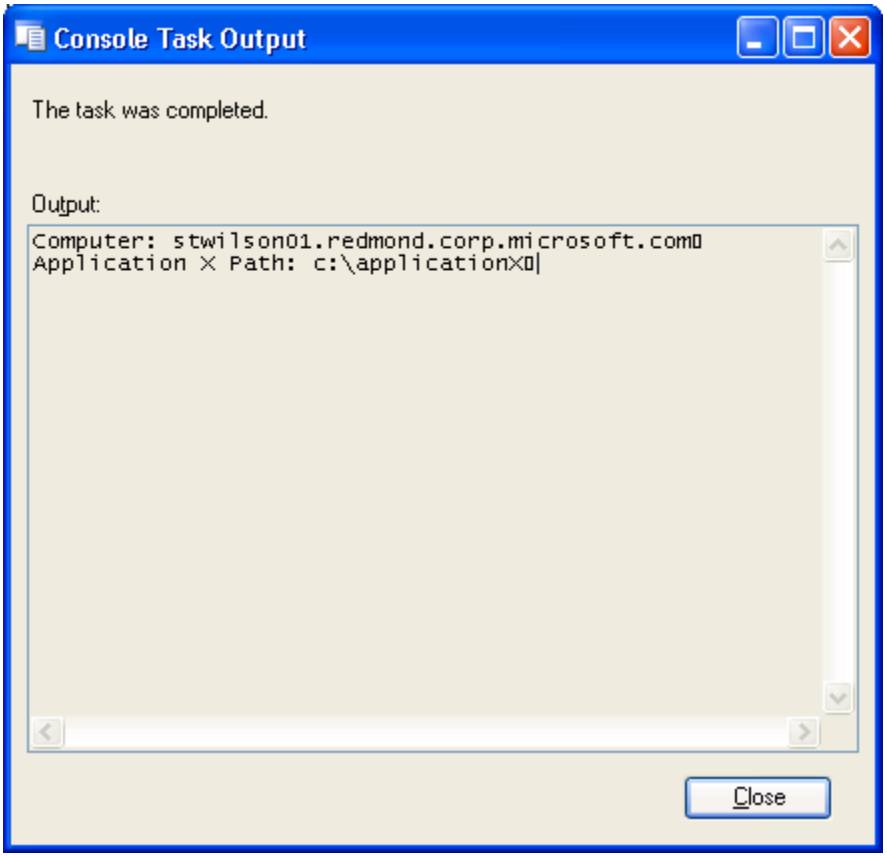
AuthorMPs.Demo.ConsoleTasks.ApplicationX.EchoObjectProperties

Target: Application X

This task is a really basic one that echo’s out a property of the Application X instance and also a host property to show you that you can use any target property or any property on you host to pass as configuration.

To run this task, go to the Application X state view that I provided, right click on an instance and select **AuthorMPs Demo Application X Tasks, AuthorMPs – Echo App X Properties.** You will see a command

output window with the computer name this instance is hosted by and the path property of this instance of Application X:



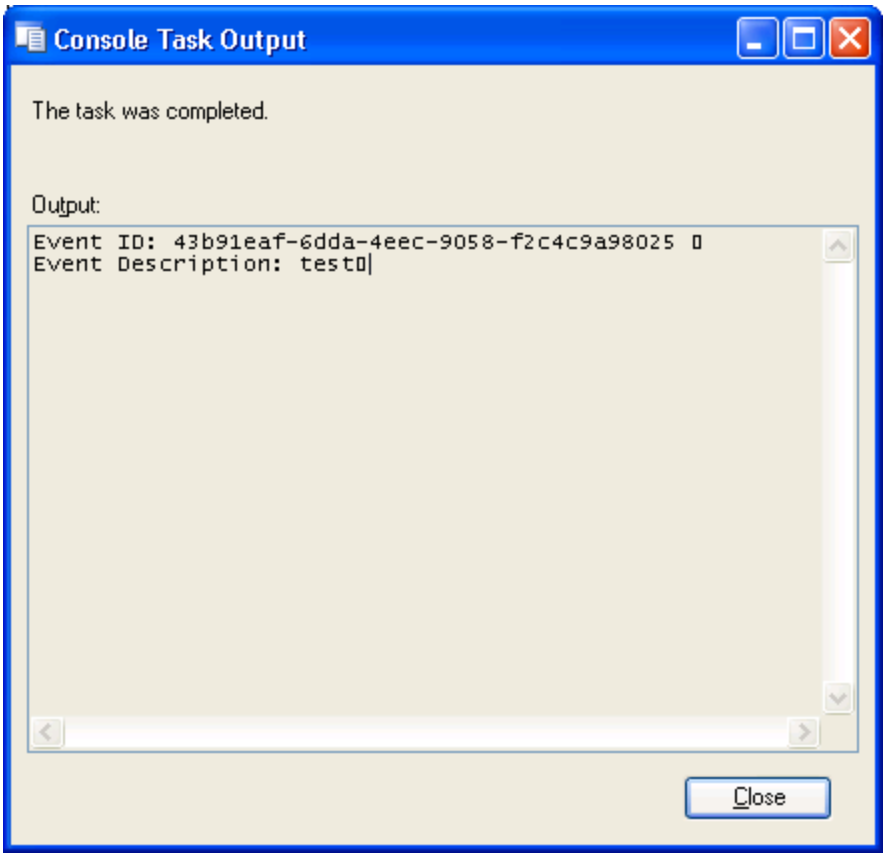
Notice that I set the **RequireOutput** attribute to true this time. You should do this when you are launching a non-interactive command that returns some output so Operations Manager leaves the output window open.

AuthorMPs.Demo.ConsoleTasks.ApplicationX.EchoEventDescription

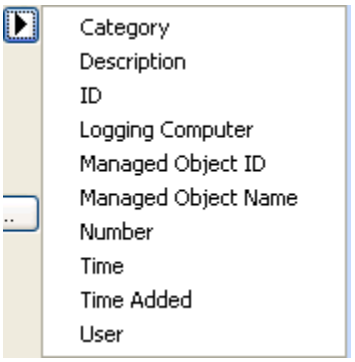
Target: Entity

This task is an event based console task. The task will only show up in an event view when you have selected an event. This is done by setting the **Category** attribute to **Event.**

To see this working generate an event against the Application X instance (see previously in this guide for how to do this). Now go to the event view I provide, select an event and launch the **AuthorMPs – Echo Event Properties** task from the actions pane. You will get some output like the following:



I only use the ID and description field of the event in my task – there are more fields you can use. You can see the list from the console task properties dialog:



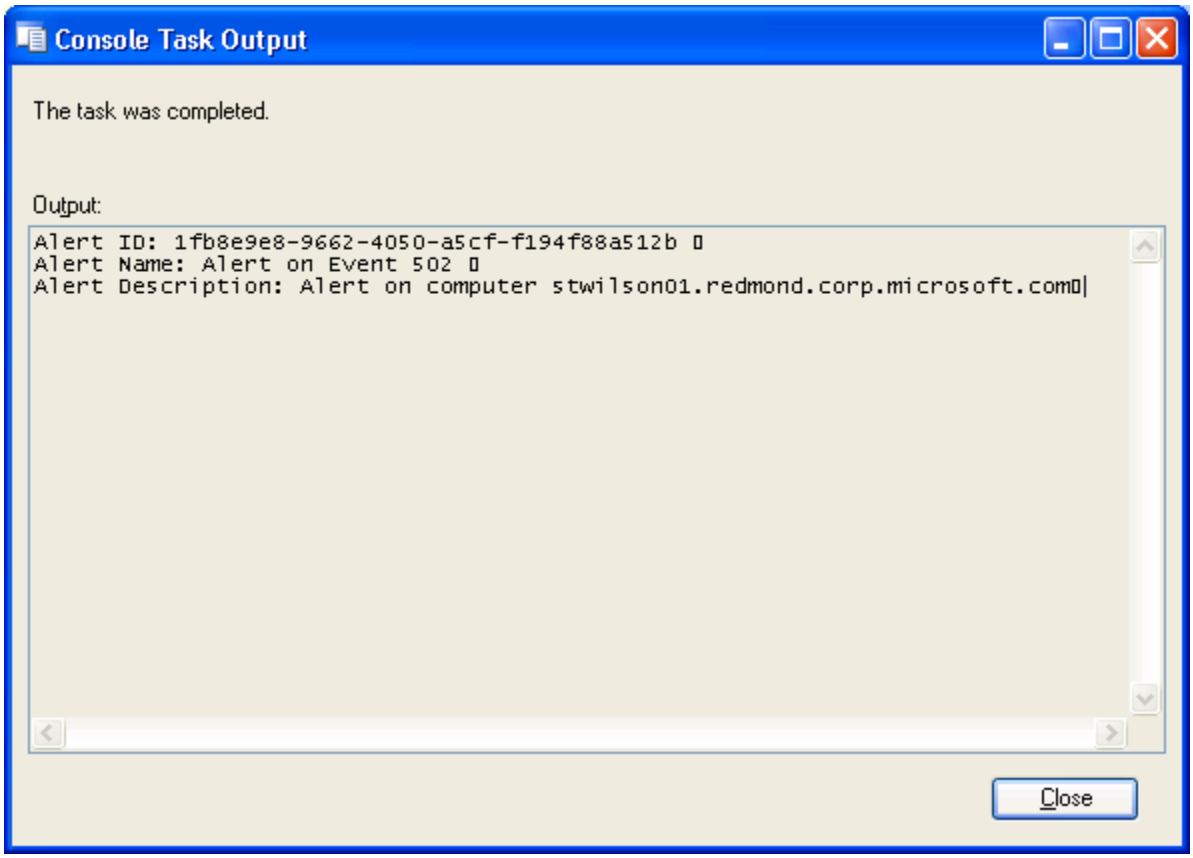
Note that all event based console tasks should be targeted to System.Entity in the System.Library management pack.

AuthorMPs.Demo.ConsoleTasks.ApplicationX.EchoAlertDescription

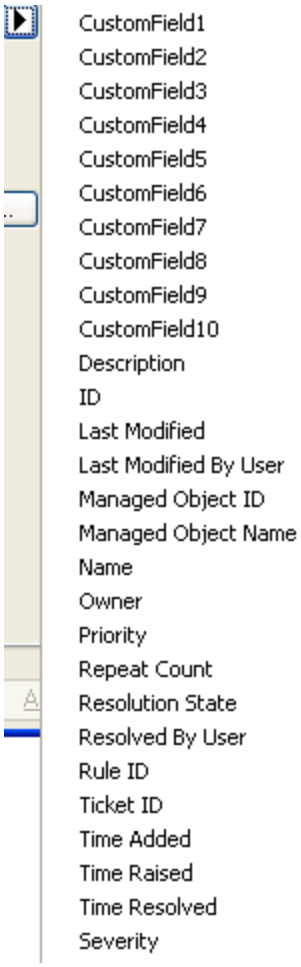
Target: Entity

This task is an alert based console task. The task will only show up in an alert view when you have selected an alert. This is done by setting the **Category** attribute to **Alert.**

To see this working generate an alert against the Application X instance (see previously in this guide for how to do this). Now go to the alert view I provide, select an alert and launch the **AuthorMPs – Echo Alert Properties** task from the actions pane. You will get some output like the following:



I only use the ID and name and description field of the alert in my task – there are more fieldsyou can use. You can see the list from the console task properties dialog:



Note that all alert based console tasks should be targeted to System.Entity in the System.Library management pack.