Control-M: Creative notification options in BMC Control-M

Posted by Joe Goldberg Mar 18, 2012
-by Joe Goldberg, Lead Technical Marketing Consultant, Control-M Solutions Marketing, BMC Software Inc.

When jobs fail, BMC Control-M provides a variety of notification options such as sending email or opening Remedy incidents.

However, events don't have to be only errors and DO MAIL, DO REMEDY and WHEN OK are not your only alerting options.

Shout Destinations give you almost unlimited options for what you can do when "events" occur in Control-M jobs.

The Shout Destination table allows you to define your own logical destinations and even to modify these on the fly. So let's say you want to send an email to the on call programmer by using destination ONCALL but you wish to change who that person is from time to time (maybe shift to shift or day to day). You can create multiple tables and switch them at the beginning of each shift (run the "ctmshtb" utility for Control-M/Server or the Modify command with the NEWDEST parameter for Control-M for z/OS).

ctmshtb SHIFT001   Run this at 7:00 to activate the Shift 1 Destination Table that defines
    ONCALL as Joe_smith@mycompany.com

ctmshtb SHIFT002   Run this at 19:00 to activate the Shift21 Destination Table that defines
   ONCALL as jack_brown@myafterhoursserviceprovider.com

What if you want to run a program or a script when a job ends? However, the action is not one that is directly part of the business process. Or perhaps you want to perform an action that Control-M does not natively support? You can define a custom logical destination and SHOUT to it. If you need to pass job attributes to the program or script, you can imbed them in the shout message using auto-edit variables.

Let's take a look at a simple example. You have a data validation application that runs on a regular basis and returns a checksum value for each execution that is important to the Finance Department. These users need to see 10 to 20 results for a particular date and time and would prefer to keep a running log file rather than searching through dozens of emails. The value we want is returned as the completion status of the application. We can see the SHOUT, which includes the %%COMPSTAT AutoEdit variable, in the job definition as follows:

In this example, we have created a custom destination called LOG2FILE. When a shout to this destination is performed, a script is run and the SHOUT message text, which includes the %%COMPSTAT AutoEdit variable, is included as a parameter.

This destination is added to the active Shout Destination Table via the Control-M Configuration Manager (CCM). Right click on a Control-M/Server and select Shout Destinations Manager to view thHere ise dialogs below.
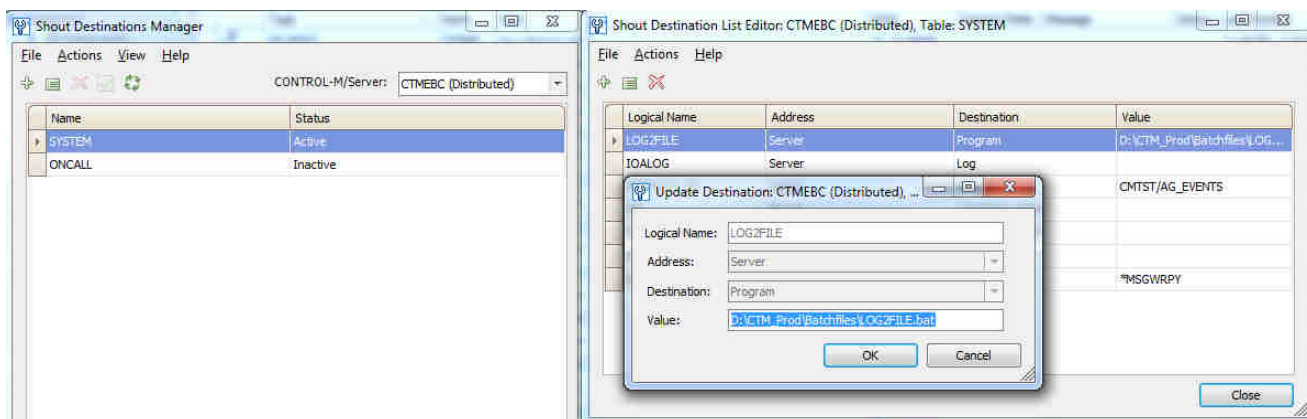
The script specified as the "value" is listed here:
@echo off

echo %2 >> D:\Users\Logs\OurFile.txt

Shout destinations have been used to open incidents in help desk tools, to collect information as we saw above and to "annotate" job execution in the Control-M log (use the "Log destination type).
Please share your experiences of how you have used Shout Destinations in your environment.

13974 Views

Vanz Vanderbilt *in response to* Clint Adams *on page 3*
Aug 20, 2015 1:50 PM
Thanks for this additional info Clint.
I would surely try out in our environment to add value to the business.

Vanz Vanderbilt *in response to* Mike Landis *on page 3*
Aug 20, 2015 1:48 PM
Hi Mike,

Thanks for easy to understand details for filling the shout destination table.
However it seems that the Control-M is still unable to send outgoing emails for a notification.

It seems that problem lies with the smtp relay of the server itself. I will figure this out with the server team.

Thanks again.

Clint Adams *in response to* Vanz Vanderbilt *on page 3*
Aug 20, 2015 12:26 PM
We do something similar to this. We have a month end process that has to be run by 5pm and it is a stream run adhoc from a user via Self Service. One month they forgot so we now have a dummy job that if it hasn't run by 5pm, it will launch a script via the shout destination table that uses ctmorder and schedules out the job and runs it.

Mike Landis *in response to* Vanz Vanderbilt *on page 3*
Aug 20, 2015 10:32 AM
Vanz
have you tried a notification before Job Completion?
WHEN: Job Not Submitted by <time>   The use Time: Specific Time and set it for 0815.

Destination: use the Logical Name from your shout table.
Message: %%JOBNAME is late

Shout Destination table

Logical Name: Whatever you want
Address: Server
Destination: Mail
Value: your@email.com

Vanz Vanderbilt

Aug 18, 2015 11:24 PM

We have three jobs that we would like to shout an email to users that it has not started at a certain time and that is late.

Will somewhat similar steps can be applied for such scenario?

Exectime 0815 if job does not trigger, perform an email notifying that the job is late.

How do I define the variable's within the shout table? Can I have a sample please?

Laurie Kenley

Mar 18, 2012 9:29 PM

Some customers have jobs that can only run for a certain amount of time before their execution time impacts the downstream jobs, so they're routinely killed. Operators used to monitor these jobs and kill them by hand. Using a shout destination, you can automate this by shouting to a script to auto-kill a job if it runs longer than its set execution time. Very handy.