

# Initialize And Finalize Scripts: How do I run code before the initialize script runs?

*How do I run code before the initialize script runs?*

By installing custom code into the init script of the initrd of a software image.

So, rephrasing:

*How do I insert custom code into the init script of the initrd of a software image?*

Typically, you want a custom piece of code to get executed before the node installer is started and identifies the compute node on which it is running. Which means before the 'initialize' script is run. (See the Bright Admin manual for what an initialize/finalize script is). Running custom code in the initrd can be useful when troubleshooting or addressing hardware-related issues very early in the system's boot process.

Here's a walkthrough of modifying the init script for Bright Cluster Manager 6.x for a specific software image. It assumes that your software image is using the redhat6 based linux distribution, but similiar steps are also possible for other distributions. The init script (under `/cm/node-installer/scripts/`) to change for the software image varies depending on the distribution and version of the image. Eg:

software image distro/version and init script file name	RHEL 6	initrd_linuxrc.r
	RHEL 5	initrd_linuxrc.
	SLES11	initrd_linuxrc.

It would be wise to start the modifications by adding a simple code snippet to test whether the whole process has been done correctly. E.g. code that pauses the boot process for 10 seconds and outputs a distinct message to the console.

## 0) make a copy of your `/cm/node-installer/scripts/initrd_linuxrc.redhat6` file

```
cp /cm/node-installer/scripts/initrd_linuxrc.redhat6{, _back}
```

# Initialize And Finalize Scripts: How do I run code before the initialize script runs?

## 1) edit the initrd init script on your headnode

```
/cm/node-installer/scripts/initrd_linuxrc.redhat6
```

When editing the code, note that there are some branches of the code which get executed only for cloud compute nodes. If you want your code to be executed on regular (non-cloud) compute nodes, then do not add it into those branches.

Also, note that the init script is a "sh" script, not a "bash" script. So, do not use bashisms.

## 2) recreate the initrd for the software image used by your nodes:

```
[root@headnode scripts]# /cm/node-installer/scripts/mkinitrd_cm.redhat  
/cm/images/default-image/
```

*(in Bright 6.x use 'mkinitrd\_cm.suse' for SLES software images, and 'mkinitrd\_cm' for rhel5-based software images)*

This step will, among other things, put the init script which you have just modified, inside of the initrd of the software image used by the compute nodes.

## 3) Restore the original init script in the /cm/node-installer/scripts/ directory using the copy you made earlier [in step 0]

Note: the modified init script at this point is already inside the regenerated initrd of the software image. If you do not restore the original init script you risk unintentionally reusing this modified init script for a different software image. For example, when adding a kernel module to a different software image, the modified init script might get reused when recreating the software image's initrd.

That's all. You can now reboot the compute node and test your modified init script.

---

### Tips:

# Initialize And Finalize Scripts: How do I run code before the initialize script runs?

To reboot the compute node from within the init script execute the following code snippet:

```
echo "1" > /proc/sys/kernel/sysrq  
echo "b" > /proc/sysrq-trigger
```

Unique solution ID: #1106

Author: Piotr Wachowicz

Last update: 2013-03-27 10:51