
How to : TESTIMAGE

- What is testimage?

The command Shell *testimage* is a *SH* program which helps you to test an environment for nodes of the cluster. You can install any software and then check if the new environment is always reliable. After decided that the program must be into nodes of the cluster you can commit it. But if you aware of the environment is not stable, *testimage* allow you to delete the change by a simple roll-back operation.

This document aim to make you understand how the command *testimage* can help you. If you want more details, you have the list of options at the end of this document.

- Syntax

The syntax of this command is quite simple: ***testimage*** [OPERATION] [OPTION]

There are many operations available like the edition, the commit, the roll-back. You can also have the list of the image knew and clear the test image that you do not want any more.

Let's see each operation that we have :

- List

The command *testimage* is abled to list all system images available on the machine. When you edit an image, the program create a new system image in order to not modified with the original. Then, the commit operation copy all changes from the test image to the original.

When you edit an image, you can see into the list a new image named with the suffix “.new”. For the list of all images do :

```
$ testimage --list
```



Any options are ignored.

- Edit

The first step for testing software is to install it to a system image. So you must edit an image:

```
$ testimage --edit --name <animage> --node <anode>
```

You specify the name of the image in which you want to install the software. If the test image already exists you can edit it. So that erase the old test image before edition, you must use the option *--force-copy*. Then you have to tell if you want to update a node after the edition by giving the name of nodes concerned. You can make the update to a set of nodes by using the *--range* option. Thus the program uses the node option value and the range value to generate a set of nodes. For example:

```
$ testimage --edit --name workimage --node wali --range 11-13
```

This command line updates “wali11”, “wali12” and “wali13” after editing the image. It's possible to make a complete installation with partitioning, and installation of all the system through the option *--force-install*.



*You can also use the option *--no-update* instead of the *--node* if you don't want to update.*

- Commit

The commit operation is used to copy the change of the new environment to the current. Thus all nodes will have the new software after the next update. And the test image becomes the current. This is done by the command line:

```
$ testimage --commit --name <animage> --node <anode>
```

If you want to install the change, either you specify a node or you use all nodes which are linked to the image. If you want to update later, you must use the option *--no-update*.



*In order to know the nodes which are linked to any images, you can use the command *mksimachine*.*

- Roll-back

The program permit to recover the last environment thanks to roll-back operation. You can specify a set of nodes for updating (*--range*). By default, all nodes linked to the image are updated. The option *--no-update* is also available with this operation.

```
$ testimage --rollback --name <animage> --node <anode>
```



The current environment is lost after a roll-back operation.

- Other options

--postinstall-script : Use this option if you should execute a script after the update of nodes. The script must be in all nodes that you want to update.



You can rename an image by using `si_mvimage` which is a command line from SystemImager package.