

Genesis II FUSE User's Manual

Mark Morgan
Global Bio Grid Group
Department of Computer Science, University of Virginia

Preface

As of Genesis II version 1.2.5, Genesis II now ships with a Java-based FUSE file system implementation. With this file system driver, users can mount the Grid namespace directly into their UNIX environment. Once this is done, the file system works just like any other mount on your UNIX system. In addition to the normal file and directory operations supported by Genesis II, users can also copy JSDL documents into BES containers, or Genesis II Queue resource, thereby creating or submitting new jobs into the grid. This functionality makes it much easier for legacy applications and users to write programs or shell scripts that automatically submit jobs into the grid.

In order to use the Genesis II FUSE file system driver, FUSE must already be installed on the target machine (though this is the default for most Linux installs).

Installing Genesis II FUSE

Genesis II FUSE is installed by default when a user installs Genesis II. The only pre-requisites are that the FUSE kernel driver must already be installed on the target machine (which is the default for Linux boxes) and it's use must be permitted by the current configuration of that machine.

In order to verify that FUSE is installed on your machine (and that you have permission to use it), first check that the fuse driver exists at `/dev/fuse` and that the intended user has read/write permissions on it (this is often accomplished by putting that user into the fuse group on that machine). Finally, verify that the user has access to the **fusermount** binary and that that binary is in the user's path.

Assuming that Genesis II is correctly installed and that fuse is installed, no other steps are required.

Running Genesis II FUSE

In order to mount the Genesis II FUSE module, you must first create a directory to mount it against (this is true for mounting any file system in a UNIX operating system). Then, a user simply runs the Genesis II shell command¹ (located in your Genesis II install and called **grid**), logs into the grid as some grid user, and issues the fuse command.

```
USAGE: fuse { --unmount <mount-point> | --mount <mount-point> [--uid=<uid>] [--daemon] }
```

¹ At the moment, Genesis II FUSE can only be mounted from the Genesis II command shell.

The *umount* version of the command allows a user to unmount a previously mounted FUSE file system (although the UNIX *fusermount -u <mount-point>* command will also work). The *mount* version is used when the user wants to mount a new mount point on their local machine. In both cases, mount-point refers to the local directory (not the grid directory) to mount the Genesis II namespace into.

By default, Genesis II FUSE uses the current user id as the user id for all file ownership mappings (in other words, all grid files are reported as “owned” by that user id (though they may not all be readable/writable/executable by that user). Further, all file permissions are reported relative to the grid user that one is logged in as when one mounts a Genesis II FUSE file system. If you wish to alter the user id that grid files are reported relative to, then use the *--uid* option.

Finally, Genesis II FUSE supports the option of being run in a background thread² (thereby freeing the Genesis II console up for other uses). If the **fuse** command is given with the *--daemon* option, then Genesis II FUSE will start in a separate Java thread and run asynchronously with respect to the grid shell. Note however that exiting the grid shell will also exit the fuse driver.

A Note About Genesis II FUSE Output

By default, the Java FUSE driver is quite verbose in its output and as a result, tends to litter the console where the driver is run with significant output. At the moment, this is being left in the driver to promote debugging. On occasion, this output will contain Java stack traces. However, this should not be considered alarming. Most often, these are caused by attempts to lookup paths that do not exist (this can occur for various reasons including mkdirs and file creates as well as some operating systems which attempt to find various hidden folders or files in directories like .thumb files, etc.).

² This is an interesting option, but of dubious use. The Genesis II FUSE driver is typically quite verbose and tends to fill the console from which it is loaded with a lot of random output.