MedeA Command-Line Tools: Execute Workflows Through the Command-Line

Contents

1 Introduction

While *MedeA* is primarily intended to be used through a graphical user interface (GUI), there are instances for which access to a command-line interface would be helpful. In particular, executing existing *MedeA* Flowcharts without human intervention or the use of a GUI can be helpful for continuous integration and development efforts. While most *MedeA* users use *MedeA* itself as a workflow manager, some users may want to incorporate *MedeA* as a single step in a multi-physics workflow involving several other software packages. For these reasons, basic functionality for executing, but not constructing, *MedeA* Flowcharts through a command-line interface is provided.

2 Commands and Options

In order to submit a new Job, the syntax is the following if you run from e.g. a bash console:

```
./bin/Linux-x86_64/mdlauncher cmd jobsubmit --help

jobsumit options:
-dir value           The directory to save files to if the -wait option is set.␣
↪Creation of directory will be attempted if it does not exist.  <>
-flow value          The workflow file (computational protocol). <>
-queue value         The JobServer queue to submit to. If omitted, the first␣
↪available queue is used. <>
-title value         The Job title. <>
-description value   The Job description. <>
-nproc value         The requested number of processors/cores. If omitted, default␣
↪queue value is used. <>
-priority value      The requested priority. If omitted, the default queue value is␣
↪used. <>
-structure value     The initial structure file. <>
-server value        The JobServer to submit to. <local>
-v                   Verbose mode, print information to standard output.
-wait                Wait for job completion.
```

For example, a *MedeA VASP* job can be started like this:

```
./bin/Linux-x86_64/mdlauncher cmd jobsubmit -flow ~/MedeA/Flowcharts/simple_vaspAG.
↪flow -structure ~/MD/Structures/Elements/Ag.sci -nproc 2 -priority 8 -title
↪'Command\ line\ submitted\ VASP\ on\ Ag'
```

DOCUMENTATION

Note: If the server is not specified, the server is the last one used from *MedeA* (loaded from MedeA.settings). If the server is specified, the -server value must be a registered name in MD/servers.dat (if not it will display the list of available servers).

The last forcefield used in *MedeA* is used (loaded from MedeA.settings). A “Set Forcefield” stage can be added to the Flowchart to ensure that a specific forcefield is used.

If you do not include -wait, the command will just submit and return. The -wait option adds more flexibility on how to submit several jobs and wait/process output.

To check whether a Job is finished, the “jobfinished” command and options may be used:

```
./bin/Linux-x86_64/mdlauncher cmd jobfinished --help

jobfinished options:
-job value            The Job number ( > 0) to query is a mandatory option. <>
-server value         The JobServer to query. <local>
-v                    Verbose mode, print information to standard output.
```

Example output of the “jobfinished” command is as follows:

```
./bin/Linux-x86_64/mdlauncher cmd jobfinished -job 1066
finished
```

To retrieve files from a Job in a local directory for further processing and inspection, the “jobget” command may be used:

```
./bin/Linux-x86_64/mdlauncher cmd jobget --help

jobget options:
-job value            The Job number ( > 0) to query is a mandatory option. <>
-dir value            The directory to save files to. Creation of directory will be␣
↪attempted if it does not exist.  <.>
-server value         The JobServer to query. <local>
-v                    Verbose mode, print information to standard output.
```

Example output of the “jobget” command is as follows:

```
./bin/Linux-x86_64/mdlauncher cmd jobget -job 1066 -dir another_run_AG
24 files retrieved from Job 1066 in directory another_run_AG.
```

DOCUMENTATION