Performing Computations with MedeA

Contents

1 Introduction

The *MedeA Environment* includes a computational job management interface, which is a system allowing users to launch multiple jobs (and tasks) at a time, control running jobs, and retrieve data from previous calculations.

- *MedeA GUI*: Building models, submitting jobs, and analyzing and visualizing results
- JobServer: Job control, data pre- and post-processing, and storage/management of all computational results
- TaskServer: Executing computational tasks

During the default installation of *MedeA*, a working configuration of *MedeA*, the JobServer and the TaskServer are created on the local machine. While this configuration is fully functional, you will most likely add additional JobServers and TaskServers and group them into queues.

2 Starting the *MedeA GUI*

2.1 *Windows*

Start *MedeA* from the Start Menu >> Materials Design >> MedeA

2.2 *Linux*

If your desktop is properly configured, you will find a Materials Design folder application folder containing *MedeA* similar to the Start Menu in Windows.

You can always start *MedeA* in `~/MD/Linux-x86_64` with `./MedeA &`.

DOCUMENTATION

3 Launching a Job in *MedeA*

By "Job" we refer to a single or several non-interactive (batch) computational Tasks that are launched from the *MedeA* interface by invoking any of the *MedeA* modules, including *VASP*, *LAMMPS*, *GIBBS*, *MOPAC*, *Gaussian*, *MT*, *Phonon*, *Transition State Search*, *Electronics*, *UNCLE*, etc. Such a Job is controlled by the *MedeA* JobServer. All the above Jobs consist of at least one computational Task. The JobServer distributes all Tasks to TaskServer machines. An exception is the *Interface Builder* Job that runs directly on the JobServer machine.

To submit a Job, click Run in the graphical user interface, select a queue from the pop-up windows, add optional comments and confirm with Submit .

Some Jobs run directly on the JobServer (such as the *Interface Builder*). Here, Run submits directly to the current JobServer without invoking the *Queues* dialog.

Most computational Jobs consist of one or more separate Tasks. A Task is defined as a serial or parallel process (*vasp_std*, *LAMMPS*, . . .) running on one or more cores. A Job can consist of multiple Tasks and may require additional pre/post-processing of the JobServer to complete.

- Jobs with multiple Tasks: E.g., displacement calculations to derive a phonon spectrum with *Phonon*, elastic coefficients with *MT*, band structure calculations with *VASP*, deformation simulations with *Deformation*, cluster expansion calculations with *UNCLE*, transition state search calculations with *TSS*, high through-put calculations with *MT*, and *GIBBS* calculations of an adsorption isotherm.

- Jobs with a single Task: A *VASP* total energy calculation (single point), a *VASP* structure optimization, a single *LAMMPS* stage (regardless of how many sub-stages it has), and a *GIBBS* run with a single thermodynamic condition.

- Jobs without Tasks: *Interface Builder*.

When you submit a Job to a specific queue, *MedeA* figures out how many Tasks need to run to complete the Job. *MedeA* then submits these Tasks to all active TaskServers available to the selected JobServer queue.

Note: If TaskServers are not up and running, connection from the JobServer to the TaskServers is interrupted (e.g., a network problem), or when the TaskServers are fully loaded (reached maximum number of cores or Tasks allowed), the Job in question has the status of *running* (preprocessing and Task setup have started), but is unable to submit Tasks until the TaskServer is up and running, the connection is re-established, or have sufficient cores/Tasks freed up.

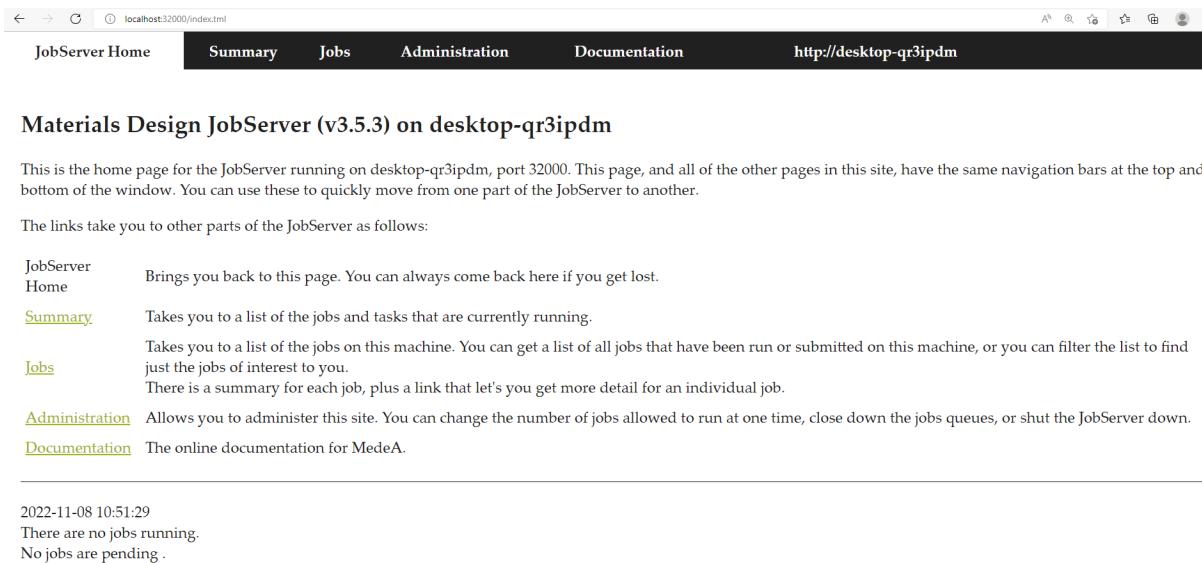In more details, the following happens when launching a Job in *MedeA*:

- *MedeA* collects information on your structure and the requested Job and sends it to the JobServer, including your input on which queue (a group of TaskServers) to run the calculations.

- The JobServer receives and processes these data creating input files for one or several Tasks required for the Job to complete. For example, this step may involve getting *VASP* PAW potentials from the SQL database or setting up several displacement calculations for *Phonon*. The status of the Job is now *running*.

- Pre-processing finished, and the JobServer checks the queue for the availability of TaskServers with free cores. The Task status is now *pending*. As soon as the TaskServer signals availability, the JobServer transfers input data and the Task is started. The Task status is now *running*. If all TaskServers are busy or not available otherwise, the JobServer queues the Tasks for later submission.

- Each TaskServer accepts a predefined number of Tasks depending on its configuration (e.g. single core, multi-core, etc.). All accepted Tasks are executed at once.

- When a Task has completed, the data is sent back from the TaskServer to the JobServer where it is processed and stored. The Task status is now *finished*, while the Job status is still *running*.

- Once the JobServer has received and processed all the required data to complete the Job, the Job status changes to *finished*.

DOCUMENTATION

Typically, the JobServer is installed as a service or a daemon, in other words, it runs as a background process and does not require direct interaction from the user to do its work. The JobServer resides either on the machine running *MedeA* or on a dedicated Windows or Linux server.

MedeA provides a web interface to the JobServer to let you view running or completed Jobs, to change the way Jobs run or to stop or restart Jobs.

4 Monitoring a Running Job

To start the web interface of the JobServer, in the *MedeA* main menu, select Jobs >> View and Control Jobs . The following page comes up in your default web browser:

The JobServer Home page navigation bar (black) has many links:

 JobServer Home : Starting page for Job controller on (default) http://localhost:32000 [1]. The JobServer listens on port 32000 and can run on a different machine than the *MedeA* GUI. Multiple JobServers can be configured to work with one *MedeA* GUI.

 Summary : Job/Task overview page displaying which Jobs are currently running and what are their Tasks. Use the Job and Task links on this page to browse to the Job/Task directory of a given Job/Task.

 Jobs : The Jobs overview page lists all Jobs running or completed on the JobServer. Use filters at the top of this page to narrow down the selection.

 Administration : JobServer configuration page with settings including automatic restart, name, and port of the JobServer machine.

 Documentation: *MedeA* documentation page with user's guide and application notes.

For more details on JobServer/TaskServer administration and configuration, please refer to the JobServer and TaskServer Administration and Configuration section.

5 Hold or Resume a Running Job

 Hold Selected stops the current Job from creating more Tasks. The Task can later be resumed with Resume Selected .

---

[1] http://localhost:32000/

On the Jobs page, select the Job numbers (Job #) with their respective checkboxes on the very left, and at the bottom of the page click Hold Selected . A Job with status *held* will not submit any more Tasks. Select a held Job and click Resume Selected to continue computations.

6 Terminate a Job

 Terminate Selected stops the current Job from creating more Tasks and tries to stop all Tasks unless a queuing system is used. *MedeA VASP* calculations (Tasks) can be stopped in a more nuanced way:

6.1 Stopping a *MedeA VASP* Task:

To stop a running *MedeA VASP* Task, click on the Job number in the Jobs page and then on the Control button next to the Task you would like to stop:

Tasks

Task	Server	Status	Time Submitted	Time Started	Time Finished	Control
204	localhost	running	2021-06-08 14:09:20	2021-06-08 14:09:22	0000-00-00 00:00:00	Control...

Choose one of the following options:

 Stop VASP after this geometry step - VASP will finish the current geometry step and stop. This option provides a valid electronic structure, total energy, and geometry, e.g. an intermediate step in an ongoing structure relaxation. The geometry optimization is not converged, though.

 Stop VASP after this electronic iteration - VASP will finish the current electronic (SCF) step and stop. No valid electronic structure and total energy will be returned, at least not a converged one. The geometry is valid, however, the geometry optimization is not converged.

 Terminate the task immediately - This command has a different implementation and works only on Linux. Moreover, you need to be aware that terminating a Task does not interact with any external queuing systems such as PBS, Slurm, Torque, GridEngine, and LSF.

- Linux TaskServer: Kills the current Task using the Unix *kill* command.

- External queuing systems: We recommend **not using** this command with external queuing systems. Log in to your TaskServer machine instead and use the queuing system specific commands to delete Tasks with e.g. *qdel* or *bkill*.

 Delete the task from the JobServer - Use this option when:

- The TaskServer is unreachable due to network problems.

- The TaskServer cannot notify the JobServer about finished Tasks.

This option returns all the files from the TaskServer to the Job directory and deletes the Task from the JobServer registry. The JobServer will continue to submit remaining Tasks and end the Job with an error due to the deleted Task. You can then restart just the Task in question using the restart function described in the next section.

 Start the task over - Use this option only when a TaskServer is unreachable due to network problems. This option deletes the task from the JobServer registry and notifies the JobServer to start it all over.

Note: Several types of *MedeA* Jobs make use of multiple Tasks, e.g. Jobs launched by the modules *MT*, *Phonon*, *Transition State Search*, or *Electronics*. Also, the *MedeA VASP* user interface launches several

Tasks in case of calculations employing hybrid functionals and for computing response tensors, band structures, the density of states, or optical spectra. On occasion, such individual multi-Task Jobs may get stuck and cannot advance, because one or more Tasks are unable to continue but remain in running mode. This may happen because TaskServers are taken offline, for example, due to network problems, full hard disks, insufficient memory, and so on, or a variety of related hardware issues. Also, atomic configurations generated in the process may prove difficult to converge electronically. In these circumstances, it is possible to enable the *MedeA* modules to make use of the calculated results of such Tasks. So, if you detect such Tasks, it is recommended that you force these Tasks to be retrieved by the JobServer, allowing the *MedeA* modules to proceed. This can be achieved by the function ⬚Delete the Task from the JobServer⬚, which returns all the files from the TaskServer to the Job directory and enables progress of the entire Job.

7 Restarting a Held or Interrupted Job

To restart a Job that has been interrupted by e.g. a network failure or held by a user, you have two options:

First option: Select the Job in the Jobs page and click ⬚Restart Selected⬚ at the bottom of the page.

- The JobServer will retrieve all fully completed Tasks.
- The JobServer will submit all uncompleted Tasks to the queue.

Second option: Click on the Job number (Job #) and then on ⬚Restart⬚ at the top of the page. In the following dialog, you can explicitly choose which Tasks to rerun and which Tasks to attempt to retrieve. This option allows you to rerun specific Tasks that may not have finished properly but were registered by the JobServer as completed.

Note: The ⬚Start Selected Over⬚ deletes all finished output files and restart the job and Tasks from scratch.

8 MedeA Preferences

The ⬚MedeA Preferences⬚ menu can be accessed by selecting ⬚File⬚ >> ⬚Preferences. . .⬚ in the *MedeA GUI*.

The ⬚MedeA Preferences⬚ menu has the following tabs:

8.1 Directories

Specify the location of various inputs and outputs for *MedeA*.

8.2 Programs

Specify external programs to interface with *MedeA*.

8.3 Modules to Load

Specify which modules to load on startup of *MedeA*.

8.4 Job Servers
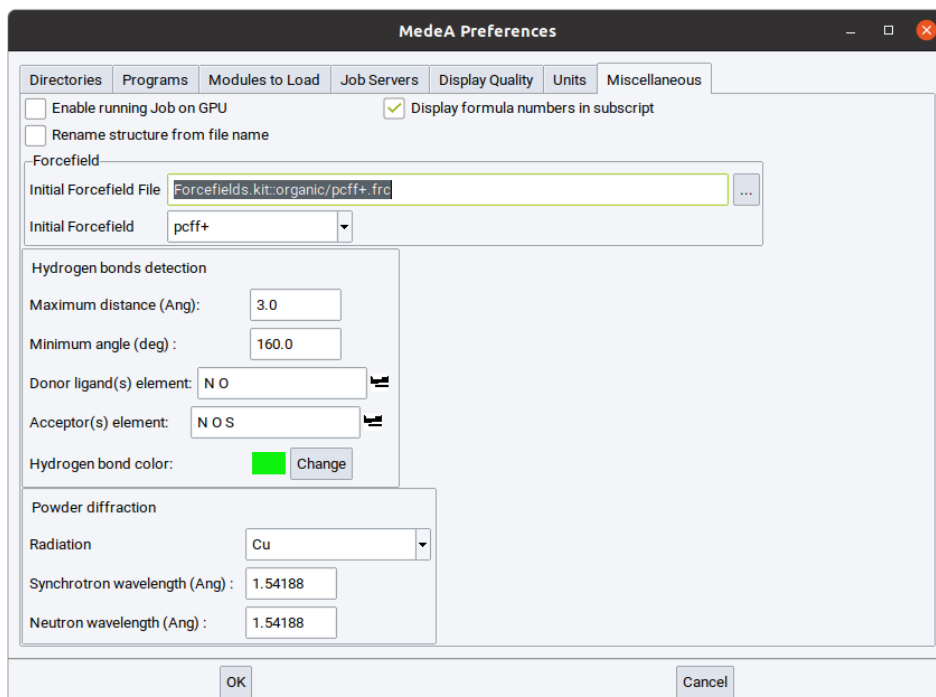
Add JobServers to *MedeA*.

8.5 Display Quality

Change the display Style and Solid Quality based on the number of atoms in the system.

8.6 Units

The default unit for energy (kJ/mol by default).

8.7 Miscellaneous

Many miscellaneous settings including:

- Enable running Job on GPU
- Display formula numbers in subscript
- Initial Forcefield file