getexample User Guide

Pawsey Supercomputing Centre

I. Getexample Outline: What Is Getexample and Why Is It Created?

Getexample is a tool designed for Pawsey Supercomputing Centre resources which provides working examples for serial and parallel programming for the users of Pawsey.

As Pawsey Supercomputing Centre has many resources which are set up differently from each other, it is usually difficult for the users to find sample codes which could be run on Pawsey's supercomputers without any modification. Even if the users manage to find well-written Bash scripts, they may not know how to submit these examples to the supercomputers they wish to use. Therefore, the *getexample* displays various examples to perform specific tasks for two particular supercomputers of Pawsey which are Magnus and Zeus. Hence, it aims to help the users in many areas listed below:

- It provides serial and parallel programming examples such as MPI, OpenMP and hybrid jobs which is a combination of OpenMP and MPI. Hence, it demonstrates the users what the main differences are between these examples and shows how to run them with various compilers and program environments.
- It supplies information about how many nodes are required for different tasks and which particular commands to utilise to submit them to Magnus and Zeus such as *aprun, srun* or *omplace*.
- It shows how to specify the partition queue and runs all the jobs on the *workq*.
- It displays how to run tasks on the scratch and store the results in the group directories effectively.

Thus, the *getexample* hopes to be a self-directed learning tool, helping the users develop and perform their own projects utilising the applicable commands from the *getexample* based on their preferences.

II. Files Included in the Getexample

Each individual example on the *getexample* occupies a directory that is reserved for them with consisting of two files listed below:

- SLURM (Simple Linux Utility for Resource Management): This allows the users to submit batch jobs to the supercomputers, check on their status and cancel them if needed. It contains the necessary information about the name of the executable, the results directory, the name of the output file and the jobID. It also allows the users to edit how many nodes the code requires to run on Magnus and Zeus, the duration of the task that it takes, which partition to be used and their account name. The SLURM initially creates a scracth directory for the example to run in and the results are outputted to a log file. Then, it creates a group directory in which the results directory is located for that example. Once the job finishes running within the scratch, the output file is then moved to the results directory located in the group directory and then the scratch directory gets removed.
- README: This file is an executable Bash script which can be read and run by the users. It provides the path to the source code and includes details about what the source code does, how to compile the source code depending on the program environment such as Cray, Intel, GNU or compilers like PGI, what can be modified in the SLURM directives, and a set of instructions on how to submit the SLURM to Magnus and Zeus including which specific commands to use for that particular supercomputer. It can be executed by simply typing ./README which then uses the path to copy the source code into the directory of the example, then it compiles the source code and submits the batch job to the chosen supercomputer.

III. How to Use the Getexample on Pawsey Supercomputing Centre Resources

The *getexample* tool currently provides examples for Magnus and Zeus which are easily accessible from both of the resources. However, it only provides examples that are applicable to the supercomputer in use. Therefore, if you log in from Magnus, you can only display the Magnus examples. To view the Zeus examples, you need to log in with your Zeus account.

To use the *getexample*, follow the steps listed below:

1) To have access to the examples, log in from either Magnus or Zeus, using the following commands respectively.

```
ssh -X myusername@magnus.pawsey.org.au
ssh -X myusername@zeus.pawsey.org.au
```

   If you are using a Windows machine, you need to use ssh -X to get a reliable X connection. If you are a Mac user, you need to specify ssh -Y (a relaxed security mode) instead.

2) To view the examples within the *getexample*, simply type:

```
getexample
```

   This lists all the examples available for the supercomputer you are utilising.

3) To require one of the examples, simply type:

```
getexample <name of the example>
```

You will see a feedback after entering this command which tells you that it creates a directory with the same name as the example on where you access the *getexample* from and copies all the files of the example into this directory.

4) To see the example, change your directory from your current directory to the new directory created by typing:

```
cd <name of the example>
```

5) To list the files located in this example directory, type:

```
ls
```

You will see 2 files as mentioned previously. We recommend you to read the descriptions located within the README and the SLURM scripts to understand what the example does and how it is performed on the supercomputer that you are using.

6) To run the example, type:

```
./README
```

This command copies the source code into the directory of the example, then compiles the source code and submits the SLURM script to the supercomputer you are currently logged in.
Once you run the README, you will see some feedback on the terminal screen. This tells you how to check the status of the job submitted, and where to find the results of the job. You can follow the information appeared on your screen to view results.

IV. Abbreviations Used in the Getexample

If you come across any abbreviations that you do not know in the examples of the *getexample*, you can refer to this page as they are provided below:

- HPC–High Perfomance Computing
- ALPS–Application Level Placement Scheduler
- SLURM–Simple Linux Utility for Resource Management
- MOM–Message Oriented Middleware
- LAMMPS–Large-scale Atomic/Molecular Massively Parallel Simulator
- GROMACS–GROningen MAchine for Chemical Simulations
- MPI–Message Passing Interface
- NUMA–Non-Uniform Memory Access
- OpenMP–Open Multi-Processing
- SGI–Silicon Graphics Inc.
- SSH–Secure SHell

V. Further Information

The *getexample* tool produced by Pawsey Supercomputing Centre is mainly implemented to make the introduction to supercomputing easier for the new users to help them understand how to run serial and parallel computing codes on Magnus and Zeus using the commands specific to each individual supercomputer. It also tries to instil into the users the behaviour of running jobs on the scratch directory, but storing the results in the group directory instead.

Therefore, Pawsey Supercomputing Centre hopes that this userguide is not only helpful and beneficial for the beginners, but also for the other users. Should you require further information on how to run jobs on Magnus and Zeus including how many nodes to specify for a particular task and which compilers to use, do not hesitate to visit Pawsey Supercomputing Centre's website. The links to Magnus and Zeus user guides are provided below:

- For Magnus:
  1) https://support.pawsey.org.au/documentation/display/US/Compiling+on+a+Cray
  2) https://support.pawsey.org.au/documentation/display/US/Running+on+a+Cray
- For Zeus:
  https://support.pawsey.org.au/documentation/pages/viewpage.action?pageId=2162999