Guide to Running ***SWAN***

What is SWAN

SWAN is a wave model to compute random, short-crested wind-generated waves in coastal regions and inland waters. This user guide provides instructions on how to run a standard SWAN test case on the Supercomputing Wales systems.

Step 1 - Log in

The examples used in this guide is configured to run on the Cardiff Skylake *Hawk* cluster. Connect to *hawklogin.cf.ac.uk* with your Supercomputing user credentials using your preferred method (e.g. PuTTY from a Windows machine or ssh from any Linux terminal, thus

```
ssh –l your_username hawklogin.cf.ac.uk
```

The steps below involve typing commands (**`in bold font`**) in the terminal window.

Step 2 - Load a SWAN module

In common with most of the application guides in this series, we assume at the outset that the module of choice would be selected from those originally available on HPC Wales. Thus a necessary first step would be to gain access to that module set by issuing the commands

```
module purge
module load hpcw
```

Note that the “module purge” command is required to prevent any unintentional collision with pre-existing modules. Subsequently issuing the module command

```
module avail hpcw
```

will provide visibility of and access to the entire set of HPCW modules.

A number of SWAN binary packages are available.

- List pre-installed SWAN versions:

  ```
  module avail SWAN
  ```

Note this command also returns SWAN-unstructured modules, which will be discussed in step 6.

- Load your preferred version (41.01 is used in this tutorial):

  ```
  module load SWAN/41.01
  ```

- Confirm the loaded modules (note the dependent compiler, MPI, and I/O library are loaded automatically):

  ```
  module list
  ```

Step 3 - Create a directory

From your home directory, create a directory to hold SWAN user files:

```
cd ~
mkdir -p SWAN
```

Step 4 - Obtain a test case

SWAN has a number of predefined test cases. For this example, we are using the academic test case 'a11refr', a copy of which is at:

```
/app/environment/SWAN/41.01/sb/intel-13.0/intel-4.1/example
```

Now copy the SLURM script and input files to your local directory by the following:

```
cd ~/SWAN
cp /app/environment/SWAN/41.01/sb/intel-13.0/intel-4.1/example/SWAN.SLURM.q .
cp /app/environment/SWAN/41.01/sb/intel-13.0/intel-4.1/example/a11refr* .
```

Step 5 - Submit a job

Now the current directory should contain all required files to run the SWAN 'a11refr' test case. Inspect the jobscript *SWAN.SLURM.q* – parallel SWAN jobs are launched using the *swanrun* script supplied by the SWAN installation (rather than using *mpirun* directly). Note the use of *$SLURM_NTASKS* and *srun*, the former an environmental variable set by the SLURM job scheduling software. This helps set the number of cores used by the job while *srun* helps generate the *machinefile* containing the list of computing nodes running the job, both required by *swanrun* to launch the job.

- Submit the job using: `sbatch SWAN.SLURM.q`
- Check the job queue using: `squeue`
- When completed, screen output can be found in a file called SWAN.Hawk.o.<JOBID> (where <JOBID> is the unique job ID generated by the queuing software). A number of other files should be created containing the results of the run.
- For more information, check the SWAN user manual in the reference section below.

Step 6 - Using SWAN-unstructured

SWAN can better represent real-world complex geometries when using unstructured grid. These variants are provided as separate SWAN-unstructured modules on the HPC Wales systems. Note that SWAN-unstructured is not available by default for all users. Please contact the helpdesk to ask about getting access to the software.

Once you have been registered to use SWAN-unstructured please follow the steps here to run an unstructured test case called F32:

- Repeat step 2, but load a SWAN-unstructured module instead:

```
module purge
module load hpcw
module load SWAN-unstructured/41.01
module list
```

- Create a working directory and copy the F32 test case:

```
cd ~
mkdir -p SWAN-unstructured
cp -r /app/environment/SWAN-unstructured/41.01/sb/intel-13.0/intel-4.1/example/* .
```

- Repeat step 5 to run the case. This time the test case is serial.
- This job will take a couple of minutes to complete. Examine the output to confirm everything is OK.
- For more information, check the unstructured SWAN tutorial in the reference section below.

References

- SWAN website: http://swanmodel.sourceforge.net/
- SWAN user manual: http://swanmodel.sourceforge.net/online_doc/swanuse/swanuse.html
- Unstructured SWAN, a tutorial: http://swanmodel.sourceforge.net/unswan/unswan.htm