ASPECT INSTALLATION - N8 POLARIS CLUSTER v1.0
Phil Heron, Durham University
July 2017
email: philip.j.heron@durham.ac.uk

Setup for ASPECT 2.0.0 and dealii 8.50 using candi.

STEP 1 - downloading Candi

Download candi

```
git clone https://github.com/dealii/candi
```

———————

STEP 2 - setup modules

Setup modules

```
module unload intel/12.1.5.339 openmpi/1.6.5
```

```
module load gnu/5.3.0 openmpi mkl
```

The module list we have is therefore:

```
  1) licenses        3) bit/64        5) user         7) openmpi/
1.6.5
  2) sge             4) durham        6) gnu/5.3.0    8) mkl/
10.3u11
```

We then export the mpi set up:

```
export CC=mpicc; export CXX=mpicxx; export FC=mpif90; export FF=mpif77
```

We should also find the path to MKL:

```
module show mkl
```

which will show many lines, but this is the one we need:

```
prepend-path  LD_LIBRARY_PATH /apps/developers/libraries/mkl/
10.3u11/1/bit-64-gnu-5.3.0/mkl/lib
```

We will use the path /apps/developers/libraries/mkl/10.3u11/1/bit-64-

```
gnu-5.3.0/mkl/lib later on in the build.
```

————————

STEP 3 - editing Candi setup

As candi can download a great number of packages that deal.ii can work with, we only need to download the ones that are directly relevant to ASPECT. As a result, we can can comment out the packages that we do not need (which are open cascade, PETSC, SLEPC).

```
cd candi


vi candi.cfg

line 19 needs to be edited to:

INSTALL_PATH=//home/polaris_dur1/*username*/candi

We need to edit the following:

PACKAGES="load:dealii-prepare"
#PACKAGES="${PACKAGES} once:opencascade"
#PACKAGES="${PACKAGES} once:parmetis"
#PACKAGES="${PACKAGES} once:superlu_dist"
PACKAGES="${PACKAGES} once:hdf5"
PACKAGES="${PACKAGES} once:p4est"
PACKAGES="${PACKAGES} once:trilinos"
#PACKAGES="${PACKAGES} once:petsc"
#PACKAGES="${PACKAGES} once:slepc"
PACKAGES="${PACKAGES} dealii"
```


In the same file, we need to look into specifying the MKL libraries. Therefore, towards the end of the file we need to edit their path:

```
MKL=ON
MKL_DIR=/apps/developers/libraries/mkl/10.3u11/1/bit-64-gnu-5.3.0/mkl/
lib
#BLAS_DIR=
#LAPACK_DIR=
```


————————

STEP 4 - running Candi

From here we can run candi. However, we need to specify the platform we wish to run on, which in this case is centos 7.

```
./candi.sh --platform=deal.II-toolchain/platforms/supported/
centos7.platform -j8
```

(where j8 is the number of processors you have available to run the configuration and installation of deal.ii and it's dependencies). I would recommend running -j8.

You are asked to review the set up of candi once the installation begins. Press enter for these parts unless you notice an error. The program will make sure you have the relevant external libraries installed and then make sure the compilers are correct: c++ etc.

Build finished in 3315 seconds = 55 minutes

————————

STEP 5 - testing dealii

Head to the build directory to test:

```
cd /home/polaris_dur1/durph/deal.ii-candi/tmp/build/deal.II-v8.5.0
```

```
make test
```

This should run 9 tests, they all should pass —>

```
[durph@login1.polaris deal.II-v8.5.0]$ make test
Running quicktests...
Test project /home/polaris_dur1/durph/deal.ii-candi/tmp/build/deal.II-
v8.5.0/tests/quick_tests
    Start 1: step.debug
1/9 Test #1: step.debug ......................    Passed    20.10 sec
    Start 2: step.release
2/9 Test #2: step.release ....................    Passed    18.94 sec
    Start 3: affinity.debug
3/9 Test #3: affinity.debug ..................    Passed    17.40 sec
    Start 4: mpi.debug
4/9 Test #4: mpi.debug .......................    Passed    19.95 sec
    Start 5: tbb.debug
5/9 Test #5: tbb.debug .......................    Passed    16.85 sec
```

```
    Start 6: p4est.debug
6/9 Test #6: p4est.debug ......................    Passed    18.67 sec
    Start 7: step-trilinos.debug
7/9 Test #7: step-trilinos.debug ..............    Passed    17.45 sec
    Start 8: lapack.debug
8/9 Test #8: lapack.debug .....................    Passed    16.67 sec
    Start 9: umfpack.debug
9/9 Test #9: umfpack.debug ....................    Passed    18.59 sec

100% tests passed, 0 tests failed out of 9

Total Test time (real) = 164.63 sec
Built target test
```

————————

STEP 6 - building ASPECT

Wherever you would like ASPECT to be downloaded,

```
 git clone --recursive https://github.com/geodynamics/aspect.git
```

Once downloaded, we can begin to build and then configure. For the build we need to let ASPECT know where deal.ii is, and from there deal.ii is configured and linked to trilinos and p4est. The example I give below is using my path:

```
cd aspect

cmake -DDEAL_II_DIR=/home/polaris_dur1/durph/deal.ii-candi/tmp/
build/deal.II-v8.5.0 .
```

Once the cmake is complete, you need to make the executable:

```
make -j8
```

Again, I would recommend using a number of processors for this make, as it can take a while.

————————

STEP 7 - testing ASPECT

Once complete, run a make test to check to see if the executable is working correct:

```
cd tests

make test

Running tests...
Test project /home/polaris_dur1/durph/aspect/tests
    Start 1: quick_mpi
1/1 Test #1: quick_mpi ........................   Passed    0.55 sec

100% tests passed, 0 tests failed out of 1

Total Test time (real) =   0.56 sec
```

————————

STEP 8 - release version of ASPECT

I would then make a copy of the debug release:

cp aspect aspect.dgb

and then rerun the make

make release

to get a release version of the executable.

cp aspect aspect.rel