Compilers on NERSC Systems

Michael Stewart
NERSC User Services

September 10, 2013
Compilers on NERSC Systems

- **Crays (Hopper and Edison)**
  - PrgEnv modules provide links to MPI and math library libraries and includes.
  - Invoke compilers with wrapper commands and the loaded PrgEnv module will invoke the proper compiler: ftn (Fortran), cc (C compiler), and CC (C+ +).
  - Available compiler modules: pgi (only on Hopper), intel, cray, and gnu (gcc).
  - Default PrgEnv modules.
    - Hopper - pgi
    - Edison - intel

- **Carver**
  - openmpi wrapper provides links to MPI libraries and includes, (but not to the math library MKL) depending on the loaded compiler module.
  - Wrapper commands: mpif90, mpicc, and mpiCC
  - Available compilers: pgi (default), intel, and gcc.
Building and Running SW on Hopper and Edison

- **Configure scripts.**
  - Use wrapper commands for compilers, e.g.
    - `./configure FC= ftn CC=cc CXX=CC`
  - Executables should be statically linked:
    - `make LDFLAGS="-static"`

- **MPI and Math library and include paths are included by default as part of the wrapper, so leave these Makefile fields blank.**

- **Batch scripts do not run on compute nodes by default.**
  - Only commands launched with aprun will run on compute nodes which are dedicated to the batch job.
  - All other commands run on a “mom” node, a type of interactive node which is shared and subject to overloading.
Compilers on Edison

- **Intel**
  - Default on Edison.
  - Uses Intel MKL math library by default and not Cray libsci.
  - Very well optimized code with option “-fast -no-ipo”.

- **Cray (cce module)**
  - Uses Cray libsci math library by default.
  - Very well optimized code with default optimization “ “.

- **Gnu (gcc module)**
  - Uses Cray libsci math library by default.
  - Best optimization: “-Ofast”

- **Recommendations**
  - Performance: Intel or Cray.
  - Portability: Gnu or Intel
Compilers on Hopper

- **PGI**
  - Default on Hopper.
  - Best optimization: “-fast”.
- **Intel**
  - Very well optimized code with option “-fast -no-ipo”.
- **Cray (cce module)**
  - Very well optimized code with default optimization “ “.
- **Gnu (gcc module)**
  - Best optimization: “-Ofast”.
- **All compilers use the Cray libsci math library by default.**
- **Recommendations**
  - Performance: Intel or Cray.
  - Portability: Gnu, PGI, or Intel.
  - PGI is less picky about standards, and any given code is more likely to compile with PGI than with the others. PGI performance has improved recently.
Compilers on Carver

● **PGI**
  ○ Default on Carver.
  ○ Best optimization: “-fast”.

● **Intel**
  ○ Very well optimized code with default level of optimization, no optimization arguments.
  ○ Option “-fast” does not work on Carver for MPI codes.

● **Gnu (gcc module)**
  ○ Best optimization: “-Ofast”.

● **Intel MKL is the only high performance math library on Carver**

● **Recommendations**
  ○ Performance: Intel.
  ○ PGI is less picky about standards, and any given code is more likely to compile with PGI than with the others.
# Useful Compiler Options

<table>
<thead>
<tr>
<th>Feature</th>
<th>Intel</th>
<th>Cray</th>
<th>PGI</th>
<th>Gnu</th>
</tr>
</thead>
<tbody>
<tr>
<td>High level of optimization</td>
<td>-fast -no-ipo</td>
<td>default</td>
<td>-fast</td>
<td>-Ofast</td>
</tr>
<tr>
<td>Activate OpenMP directives and pragmas</td>
<td>-openmp</td>
<td>-homp (on by default)</td>
<td>-mp=nonuma</td>
<td>-fopenmp</td>
</tr>
<tr>
<td>Read and write Fortran unformatted data files as big-endian</td>
<td>-convert bigendian</td>
<td>-h byteswapio</td>
<td>-byteswapio</td>
<td>-fconvert=big-endian</td>
</tr>
<tr>
<td>Process Fortran source code as fixed form</td>
<td>-fixed</td>
<td>-f fixed</td>
<td>-Mfixed</td>
<td>-ffixed-form</td>
</tr>
<tr>
<td>Process Fortran source code as free form</td>
<td>-free</td>
<td>-f free</td>
<td>-Mfree</td>
<td>-ffree-form</td>
</tr>
<tr>
<td>Display compiler version number</td>
<td>--version</td>
<td>-V</td>
<td>--version</td>
<td>--version</td>
</tr>
</tbody>
</table>
National Energy Research Scientific Computing Center