

XCONFIGURE

XCONFIGURE is a collection of configure wrapper scripts for various HPC applications. The purpose of the scripts is to configure the application in question to make use of Intel's software development tools (Intel Compiler, Intel MPI, Intel MKL). This may sound cumbersome, but it actually helps to rely on a "build recipe", which is known to expose the highest performance or to reliably complete the build process.

Each application (or library) is hosted in a separate subdirectory. In order to configure (and ultimately build) an application, one may clone or download the entire collection.

```
git clone https://github.com/hfp/xconfigure.git
```

Alternatively, one can rely on a single script which then downloads a specific wrapper into the current working directory (of the desired application).

```
wget https://github.com/hfp/xconfigure/raw/master/configure-get.sh
chmod +x configure-get.sh
./configure-get.sh qe hsw
```

To configure an application, please follow into one of the aforementioned subfolders and read the build recipe of this application e.g., [qe](#) in case of Quantum Espresso.

CP2K

Build and Run Instructions

The build and run instructions for CP2K using Intel Tools are exercised at <https://github.com/hfp/libxsmm/tree/master/documentation/cp2k.md> (pdf).

Please note, in terms of functionality it is beneficial to rely on [LIBINT](#) and [LIBXC](#), whereas [ELPA](#) is not only beneficial for performance but might be also more reliable depending on the ScaLAPACK implementation.

Sanity Check

There is nothing that can replace the full regression test suite to be clear. However, in order to quickly check whether a build is sane or not: run `tests/QS/benchmark/H2O-64.inp`, and check if the SCF iteration prints like the following:

Step	Update	method	Time	Convergence	Total energy	Change	
1	OT	DIIS	0.15E+00	0.5	0.01337191	-1059.6804814927	-1.06E+03
2	OT	DIIS	0.15E+00	0.3	0.00866338	-1073.3635678409	-1.37E+01
3	OT	DIIS	0.15E+00	0.3	0.00615351	-1082.2282197787	-8.86E+00
4	OT	DIIS	0.15E+00	0.3	0.00431587	-1088.6720379505	-6.44E+00
5	OT	DIIS	0.15E+00	0.3	0.00329037	-1092.3459788564	-3.67E+00
6	OT	DIIS	0.15E+00	0.3	0.00250764	-1095.1407783214	-2.79E+00
7	OT	DIIS	0.15E+00	0.3	0.00187043	-1097.2047924571	-2.06E+00
8	OT	DIIS	0.15E+00	0.3	0.00144439	-1098.4309205383	-1.23E+00
9	OT	DIIS	0.15E+00	0.3	0.00112474	-1099.2105625375	-7.80E-01
10	OT	DIIS	0.15E+00	0.3	0.00101434	-1099.5709299131	-3.60E-01
[...]							

The column called "Convergence" has to monotonically converge towards zero.

Eigenvalue SoLvers for Petaflop-Applications (ELPA)

Build Instructions

Download and unpack ELPA, and make the configure wrapper scripts available in ELPA's root folder.

```
wget http://elpa.mpcdf.mpg.de/html/Releases/2016.05.004/elpa-2016.05.004.tar.gz
tar xvf elpa-2016.05.004.tar.gz
cd elpa-2016.05.004
wget https://github.com/hfp/xconfigure/raw/master/configure-get.sh
chmod +x configure-get.sh
./configure-get.sh elpa
```

Please make the Intel Compiler available on the command line. This actually depends on the environment. For instance, many HPC centers rely on `module load`.

```
source /opt/intel/compilers_and_libraries_2017.0.098/linux/bin/compilervars.sh intel64
```

For example, to configure and make for an Intel Xeon E5v4 processor (formerly codenamed “Broadwell”):

```
./configure-elpa-hsw-omp.sh  
make -j ; make install
```

For different targets (instruction set extensions) or different versions of the Intel Compiler, the configure scripts support an additional argument (“default” is the default tagname):

```
./configure-elpa-hsw-omp.sh tagname
```

As shown above, an arbitrary “tagname” can be given (without editing the script). This might be used to build multiple variants of the ELPA library.

References

<https://software.intel.com/en-us/articles/quantum-espresso-for-the-intel-xeon-phi-processor>

LIBINT

Build Instructions

TBD

References

TBD

LIBXC

Build Instructions

TBD

<http://www.tddft.org/programs/octopus/wiki/index.php/Libxc:download>
<http://www.tddft.org/programs/octopus/down.php?file=libxc/libxc-3.0.0.tar.gz>

References

TBD

LIBXSMM

LIBXSMM is a library targeting Intel Architecture (x86) for small, dense or sparse matrix multiplications, and small convolutions. The build instructions can be found at <https://github.com/hfp/libxsmm> (pdf).

Quantum Espresso (QE)

Build Instructions

Download and unpack Quantum Espresso, and make the configure wrapper scripts available in QE’s root folder. Please note that the configure wrapper scripts support QE 5.3 and QE 5.4 as well as QE 6.0. Before building QE, one needs to complete the [ELPA build recipe](#)!

```
wget http://www.qe-forge.org/gf/download/frsrelease/224/1044/qe-6.0.tar.gz  
tar xvf qe-6.0.tar.gz  
cd qe-6.0  
wget https://github.com/hfp/xconfigure/raw/master/configure-get.sh  
chmod +x configure-get.sh  
./configure-get.sh qe
```

Please make the Intel Compiler available on the command line. This actually depends on the environment. For instance, many HPC centers rely on `module load`.

```
source /opt/intel/compilers_and_libraries_2017.0.098/linux/bin/compilervars.sh intel64
```

For example, configure for an Intel Xeon E5v4 processor (formerly codenamed “Broadwell”), and build the desired application(s) e.g., “pw”, “cp”, or “all”.

```
./configure-qe-hsw-omp.sh  
make pw -j
```

Building “all” may requires to repeat `make all` until no compilation error occurs. This is because of some incorrect build dependencies (build order issue), which might have been introduced by the configure wrapper scripts. In case of starting over, one can run `make distclean`, reconfigure the application, and build it again. For different targets (instruction set extensions) or different versions of the Intel Compiler, the configure scripts support an additional argument (“default” is the default tagname):

```
./configure-qe-hsw-omp.sh tagname
```

As shown above, an arbitrary “tagname” can be given (without editing the script). This might be used to build multiple variants of QE. Please note: this tagname also selects the corresponding ELPA library (or should match the tagname used to build ELPA). Make sure to save your current QE build before building an additional variant!

Run Instructions

TBD

References

<https://software.intel.com/en-us/articles/quantum-espresso-for-the-intel-xeon-phi-processor>