NWChem 6.8

Webpage

https://nwchemgit.github.io/

Version

6.8

Build Environment

- GCC 8.5.0
- Intel MKL 2022.2.1
- HPC-X 2.11 (Open MPI 4.1.4)

Files Required

- nwchem-6.8-release.revision-v6.8-47-gdf6c956-src.2017-12-14.tar.bz2

Build Procedure

```
#!/bin/sh

VERSION=6.8
INSTALL_PREFIX=/apl/reactionplus/1.0/nwchem-6.8-mpipr

WORKDIR=/gwork/users/${USER}

BASEDIR=/home/users/${USER}/Software/reactionplus/1.0
NWCHEM_DIRNAME=nwchem-6.8
NWCHEM_TARBALL=${BASEDIR}/nwchem-6.8-release.revision-v6.8-47-gdf6c956-src.2017-
12-14.tar.bz2

# -------
umask 0022
export LANG=C
ulimit -s unlimited

module purge
module load mkl/2022.2.1
module load openmpi/4.1.4-hpcx/gcc8

cd ${WORKDIR}
if [ -d ${NWCHEM_DIRNAME} ]; then
  mv ${NWCHEM_DIRNAME} nwchem-erase
  rm -rf nwchem-erase &
fi

tar jxf ${NWCHEM_TARBALL}
cd ${NWCHEM_DIRNAME}

export NWCHEM_TOP=${WORKDIR}/${NWCHEM_DIRNAME}
export NWCHEM_MODULES="all python"
export NWCHEM_TARGET=LINUX64
export ARMCI_NETWORK=MPI-PR

export USE_OPENMP=y
export USE_MPI=y
export USE_MPIF=y
export USE_MPIF4=y

export USE_NOFSCHECK=TRUE
export USE_NOIO=TRUE
```

```
export PYTHONHOME=/usr
export PYTHONVERSION=2.7
export PYTHONLIBTYPE=so
export USE_PYTHON64=y

export MRCC_METHODS=TRUE
export CCSDTQ=TRUE
export LIB_DEFINES=-DDFLT_TOT_MEM=268435456 # 2GiB/process

export BLAS_SIZE=8
export BLASOPT="-L${MKLROOT}/lib/intel64 -Wl,--no-as-needed -lmkl_gf_ilp64 -lmkl_intel_thread -lmkl_core -liomp5 -lpthread -lm -ldl"
export LAPACK_SIZE=8
export LAPACK_LIB="${BLASOPT}"
export USE_SCALAPACK=y
export SCALAPACK_SIZE=8
export SCALAPACK="-L${MKLROOT}/lib/intel64 -lmkl_scalapack_ilp64 -Wl,--no-as-needed -lmkl_cdft_core -lmkl_gf_ilp64 -lmkl_intel_thread -lmkl_core -
lmkl_blacs_openmpi_ilp64 -liomp5 -lpthread -lm -ldl"

export CC=gcc
export FC=gfortran

cd ${NWCHEM_TOP}/src
make nwchem_config

sed -i -e /types.h/a"typedef size_t loff_t;" util/util_fadvise.c
sed -i -e "s/python-config/python2-config/g" config/makefile.h

make

# involve also version info
cd ${NWCHEM_TOP}/src/util
make version
make
cd ${NWCHEM_TOP}/src
make link
cd ${NWCHEM_TOP}

mkdir -p ${INSTALL_PREFIX}
# installation; assume INSTALL_PREFIX directory already exists
cp -fr LICENSE.TXT README.md release.notes.* examples ${INSTALL_PREFIX}

mkdir ${INSTALL_PREFIX}/bin
mkdir ${INSTALL_PREFIX}/data

cp -f ${NWCHEM_TOP}/bin/${NWCHEM_TARGET}/nwchem ${INSTALL_PREFIX}/bin
chmod 755 ${INSTALL_PREFIX}/bin/nwchem

cp -fr ${NWCHEM_TOP}/src/data            ${INSTALL_PREFIX}
cp -fr ${NWCHEM_TOP}/src/basis/libraries ${INSTALL_PREFIX}/data
cp -fr ${NWCHEM_TOP}/src/nwpw/libraryps  ${INSTALL_PREFIX}/data

# create default.nwchemrc
cat << EOS > ${INSTALL_PREFIX}/default.nwchemrc
nwchem_basis_library ${INSTALL_PREFIX}/data/libraries/
nwchem_nwpw_library ${INSTALL_PREFIX}/data/libraryps/
ffield amber
amber_1 ${INSTALL_PREFIX}/data/amber_s/
amber_2 ${INSTALL_PREFIX}/data/amber_q/
amber_3 ${INSTALL_PREFIX}/data/amber_x/
amber_4 ${INSTALL_PREFIX}/data/amber_u/
spce    ${INSTALL_PREFIX}/data/solvents/spce.rst
charmm_s ${INSTALL_PREFIX}/data/charmm_s/
charmm_x ${INSTALL_PREFIX}/data/charmm_x/
EOS
```

```
# run test
export NWCHEM_EXECUTABLE=${INSTALL_PREFIX}/bin/nwchem
export OMP_NUM_THREADS=1

# some tests need this...
cp -f ${INSTALL_PREFIX}/default.nwchemrc ~/.nwchemrc

cd ${NWCHEM_TOP}/QA
./doqmtests.mpi 2 >& doqmtests.mpi.log

mkdir ${INSTALL_PREFIX}/testlog
cp -fr doqmtests.mpi.log testoutputs/ ${INSTALL_PREFIX}/testlog
```

Notes

- In case export ARMCI_NETWORK=OPENIB was used, calculations failed when large number of MPI processes were employed. MPI-PR version is free from this issue.