GAMESS 2021R1 (2021Jun30)

(this version will be installed on Mar 6, 2023)

Webpage

https://www.msg.chem.iastate.edu/gamess/index.html

Version

2021R1 (2021Jun30)

Build Environment

- Intel oneAPI Compiler Classic 2022.2.1
- Open MPI 3.1.6

Files Required

- gamess-current.tar.gz (2021R1)
- gmsnbo.i8.a (NBO7.0)
- rungms_rccs_openmpi3 (please check installed rungms)
- exam43.patch

```
--- tests/standard/exam43.inp.orig      2018-03-13 11:58:15.322187865 +0900
+++ tests/standard/exam43.inp   2018-03-13 11:58:32.049289234 +0900
@@ -48,7 +48,7 @@
 ! geometry in $DATA, although this is not necessary.
 !
  $contrl scftyp=rhf runtyp=g3mp2 $end
- $system timlim=2 mwords=2 memddi=5 $end
+ $system timlim=2 mwords=10 memddi=5 $end
  $scf    dirscf=.true. $end
  $data
 Methane...G3(MP2,CCSD(T))
```

- pbs_remsh

```
#!/bin/sh
host="$1"
shift
/usr/bin/ssh -n "$host" env PBS_JOBID="$PBS_JOBID" pbs_attach $*
```

Build Procedure

```
#!/bin/sh

VERSION=2021Jun30
DIRNAME=gamess${VERSION}
INSTDIR=/apl/gamess/2021R1

# files and patches
MYROOT="/home/users/${USER}/Software/GAMESS/gamess${VERSION}"
GAMESS_TARBALL="${MYROOT}/gamess-current.tar.gz"
GAMESS_NBOI8A="${MYROOT}/gmsnbo.i8.a"
PATCH_EXAM43="${MYROOT}/exam43.patch"
RUNGMS_RCCS="${MYROOT}/rungms_rccs_openmpi3"
PBS_REMSH="${MYROOT}/pbs_remsh"

PARALLEL=12

#------------------------------------------------------------------------------
umask 0022
```

```
export LANG=C
export LC_ALL=C

module -s purge
. ~/intel/oneapi/compiler/2022.2.1/env/vars.sh
module -s load mkl/2022.2.1
module -s load openmpi/3.1.6/intel2022.2.1

IFORTVER=2021
OMPIDIR=/apl/openmpi/3.1.6/intel2022.2.1

cd ${INSTDIR}
if [ -d gamess ]; then
  mv gamess gamess-erase
  rm -rf gamess-erase &
fi

tar zxf ${GAMESS_TARBALL}
mv ${INSTDIR}/gamess/* .
rm -rf ${INSTDIR}/gamess # maybe already empty

for f in comp compall config lked gms-files.csh runall ddi/compddi \
         tools/remd.csh tools/localgms tools/libxc/download-libxc.csh; do
  sed -i -e "1s/.*/#\!\/bin\/csh -f/" $f
done

patch -p0 < ${PATCH_EXAM43}
cp ${PBS_REMSH} .

expect << EXPECT
spawn csh -f ./config
expect "After the new window is open"
send "\r"
expect "please enter your target machine name:"
send "linux64\r"
expect "GAMESS directory?"
send "${INSTDIR}\r"
expect "GAMESS build directory?"
send "${INSTDIR}\r"
expect "Version?"
send "\r"
expect "Please enter your choice of FORTRAN:"
send "ifort\r"
expect "Version?"
send "${IFORTVER}\r"
expect "hit <ENTER> to continue to the math library setup."
send "\r"
expect "Enter your choice of 'mkl' or 'atlas' or 'acml' or 'libflame' or 'openblas' or 'pgiblas' or 'armpl' or 'none':"
send "mkl\r"
expect "MKL pathname?"
send "${MKLROOT}\r"
expect "MKL version (or 'proceed')?"
send "proceed\r"
expect "please hit <ENTER> to set up your network for Linux clusters."
send "\r"
expect "communication library ('serial','sockets' or 'mpi' or 'mixed')?"
send "mpi\r"
expect "Enter MPI library"
send "openmpi\r"
expect "Please enter your openmpi"
send "${OMPIDIR}\r"
expect "Optional: Build LibXC interface?  (yes/no): "
send "yes\r"
expect "Please hit <ENTER>"
send "\r"
```

```
expect "Optional: Build Michigan State University CCT3 & CCSD3A methods?"
send "yes\r"
expect "Do you want to try LIBCCHEM"
send "no\r"
expect "Build GAMESS with OpenMP thread support?"
send "yes\r"
expect eof
EXPECT

make ddi
# do libxc first (according to the installation guide)
csh -f ./tools/libxc/download-libxc.csh
make -j ${PARALLEL} libxc

make modules
make -j ${PARALLEL}

GAMESS_NBOI8A_ESC=`echo ${GAMESS_NBOI8A} | sed -e 's/\//\\\\\//g'`
sed -i -e "s/NBO=false/NBO=true/" lked
sed -i -e "s/NBOLIB=.*/NBOLIB=${GAMESS_NBOI8A_ESC}/" lked
#sed -i -e "s/mpiifort/mpiifort -lgfortran/" lked

# retry linking; first try above would fail
csh -f ./lked

mv rungms rungms.orig
cp ${RUNGMS_RCCS} ./rungms

chmod -R o-rwx source object libcchem
chmod -R o-rwx ddi/src ddi/server ddi/olddi ddi/kickoff
find . -name "src" | xargs chmod -R o-rwx

sed -i -e "s/tools\/localgms/rungms/" \
       -e "/RUNGMS/s/VERSION)/VERSION) \$(NCPUS)/" \
        Makefile.in

export OMP_NUM_THREADS=8
make checktest
make clean_exams

export OMP_NUM_THREADS=1

TEST_LIST="eda qmefpea efp-ci standard"
# Excluded tests
# cim: does not run with parallel, and ksh not avail
# efmo: not enough memory? problem of input?
# excitatinos: no avail test for parallel?
# neb: too long
for tp in $TEST_LIST; do
  make checktest NCPUS=8 EXAM_PATH=$tp
done

ipcrm -a
```

Notes

- (This version is not carefully checked. Please check notes of2022R2 version.)