

## DIRAC 23.0

### ウェブページ

<http://www.diracprogram.org/doku.php>

### バージョン

23.0

### ビルド環境

- Intel oneAPI Compiler Classic 2022.2.1 (oneAPI 2022.3.1)
- Intel MKL 2022.2.1 (oneAPI 2022.3.1)

### ビルドに必要なファイル

- DIRAC-23.0-Source.tgz
- openmpi-4.1.5.tar.gz
  - (デフォルトを 8 バイト整数にするために既存のものとは別にビルド)
- diff\_memcon (see /apl/dirac/23.0/patches/diff\_memcon)
  - 利用可能なメモリ量を増やすためのパッチです

```
--- src/gp/memory_control.F90.orig 2019-08-08 13:42:54.000000000 +0900
+++ src/gp/memory_control.F90 2019-08-08 13:44:14.000000000 +0900
@@ -110,10 +110,10 @@

    subroutine allocator_setmax_i8(size)
        integer(kind=8), intent(in) :: size
-       integer(kind=4)           :: max_mem_i4
+       integer(kind=8)           :: max_mem_i8

        if (size.le.0) then ! there are no limits so we use a large integer
-       max_mem = real(huge(max_mem_i4))*real(kreal)
+       max_mem = real(huge(max_mem_i8))*real(kreal)
        else
            max_mem = real(size)*real(kreal)
        endif
```

### ビルド手順

```
#!/bin/sh

VERSION=23.0
INSTALL_PREFIX=/apl/dirac/23.0

# openmpi (8-byte integer)
OMPI_VERSION=4.1.5
OMPI_INSTALL_PREFIX=${INSTALL_PREFIX}/openmpi415_i8
OMPI_TARBALL=/home/users/${USER}/Software/OpenMPI/${OMPI_VERSION}/openmpi-${OMPI_VERSION}.tar.gz
PBSROOT=/apl/pbs/22.05.11/

# dirac
BASEDIR=/home/users/${USER}/Software/DIRAC/${VERSION}
TARBALL=${BASEDIR}/DIRAC-${VERSION}-Source.tgz

PATCH_README=${BASEDIR}/README.patch
PATCH_MEMCONTROL=${BASEDIR}/diff_memcon

WORKDIR=/gwork/users/${USER}

PARALLEL=8
#-----
umask 0022
```

```

export LC_ALL=C
export LANG=C
export OMP_NUM_THREADS=1

ulimit -s unlimited

module -s purge
. ~/intel/oneapi/compiler/latest/env/vars.sh
module -s load mkl/2022.2.1

# openmpi (8-byte integer default)
cd ${WORKDIR}
if [ -d openmpi-${OMPI_VERSION} ]; then
  mv openmpi-${OMPI_VERSION} openmpi_erase
  rm -rf openmpi_erase &
fi

tar zxf ${OMPI_TARBALL}
cd openmpi-${OMPI_VERSION}
mkdir rccs-i8 && cd rccs-i8
CC=icc CXX=icpc FC=ifort FCFLAGS=-i8 CFLAGS=-m64 CXXFLAGS=-m64 \
  ./configure --prefix=${OMPI_INSTALL_PREFIX} \
    --with-tm=${PBSROOT} \
    --enable-mpi-cxx \
    --enable-mpi1-compatibility \
    --enable-mpi-fortran=usempi \
    --with-ucx
make -j ${PARALLEL} && make install && make check

# dirac
cd ${WORKDIR}
if [ -d DIRAC-${VERSION}-Source ]; then
  mv DIRAC-${VERSION}-Source DIRAC_erase
  rm -rf DIRAC_erase &
fi

export PATH="${OMPI_INSTALL_PREFIX}/bin:$PATH"
export LIBRARY_PATH="${OMPI_INSTALL_PREFIX}/lib:$LIBRARY_PATH"
export LD_LIBRARY_PATH="${OMPI_INSTALL_PREFIX}/lib:$LD_LIBRARY_PATH"

export DIRAC_TMPDIR=${WORKDIR}

tar zxf ${TARBALL}
cd DIRAC-${VERSION}-Source
patch -p0 < ${PATCH_MEMCONTROL}

python3 ./setup \
  --mpi \
  --fc=mpif90 \
  --cc=mpicc \
  --cxx=mpicxx \
  --mkl=parallel \
  --int64 \
  --python=python2 \
  --extra-fc-flags="-march=core-avx2 -I${OMPI_INSTALL_PREFIX}/lib" \
  --extra-cc-flags="-march=core-avx2" \
  --extra-cxx-flags="-march=core-avx2" \
  --prefix=${INSTALL_PREFIX} \
  build.rccs
cd build.rccs
make -j ${PARALLEL} && make install

# copy license and patch files
cp -f ../LICENSE ${INSTALL_PREFIX}

```

```
cp -f ${PATCH_README} ${INSTALL_PREFIX}
mkdir ${INSTALL_PREFIX}/patches
cp -f ${PATCH_MEMCONTROL} ${INSTALL_PREFIX}/patches

# prepare test results directories
mkdir ${INSTALL_PREFIX}/test_results
mkdir ${INSTALL_PREFIX}/test_results/serial
mkdir ${INSTALL_PREFIX}/test_results/parallel

# serial test
export DIRAC_MPI_COMMAND="mpirun -np 1"
make test
cp Testing/Temporary/* ${INSTALL_PREFIX}/test_results/serial
rm -f Testing/Temporary/*

# parallel test
export DIRAC_MPI_COMMAND="mpirun -np ${PARALLEL}"
make test
cp Testing/Temporary/* ${INSTALL_PREFIX}/test_results/parallel
rm -f Testing/Temporary/*

exit 0
```

## テスト

### 失敗したテスト(serial)

```
The following tests FAILED:
 31 - fde_response_mag (Failed)
 42 - fde_response_shield (Failed)
 46 - cc_restart (Failed)
149 - laplace (Failed)
172 - x_amfi (Failed)
173 - pe_energy (Failed)
174 - pe_response (Failed)
175 - pe_exc (Failed)
176 - pe_cpp (Failed)
```

### 失敗したテスト(parallel)

```
The following tests FAILED:
 21 - fsc restart (Failed)
 31 - fde_response_mag (Failed)
 38 - dft_overlap_diagnostic (Failed)
 42 - fde_response_shield (Failed)
 46 - cc_restart (Failed)
139 - lucita_short (Failed)
149 - laplace (Failed)
172 - x_amfi (Failed)
173 - pe_energy (Failed)
174 - pe_response (Failed)
175 - pe_exc (Failed)
176 - pe_cpp (Failed)
```

## メモ

- テスト結果のコピーが /apl/dirac/23.0/test\_results 以下にあります。
- gcc 10, 11 はビルドに dirac のビルドに失敗する
- gcc 8, 9 と intel 2022.2.1 については甲乙つけがたいが、intel 2022.2.1 の方がわずかに速度が出ているように見える(主に上記テストの結果より)
- intel compiler 2023.1.0 ではエラーが少し増えるため、今回は回避
- openmpi 3.1.6 よりも openmpi 4.1.5 の方がエラーが少しだけ少ない
  - 4ノードでの並列(512コア) HF 計算が正常に動くことは確認済
- aocc については dirac が未対応で、少なくとも大幅な修正が必要だと思われるため回避
- --python=python3 ではエラー(ascii, unicode 関連)が増えるため回避

