

Open MPI 4.1.6

ウェブページ

<https://www.open-mpi.org/>

バージョン

- 4.1.6

ビルド環境

- gcc 8.5.0, 11.2.1, 12.1.1
- Intel oneAPI Compiler Classic 2022.2.1, 2023.2.0
- Intel oneAPI Compilers 2023.2.0
- AOCC 4.1
- NVIDIA HPC SDK 23.5

- OpenPBS 22.05.11

必要なファイル

- openmpi-4.1.6.tar.bz2

ビルド手順

gcc 8, 11, 12

```
#!/bin/sh

VERSION=4.1.6
WORKDIR=/gwork/users/${USER}
TARBALL=/home/users/${USER}/Software/OpenMPI/${VERSION}/openmpi-${VERSION}.tar.bz2
PBSROOT=/apl/pbs/22.05.11

PARALLEL=12

# -----
umask 0022

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

tar jxf ${TARBALL}
cd openmpi-${VERSION}

# GCC 8
module purge
INSTALLDIR=/apl/openmpi/${VERSION}/gcc8
mkdir rccs-gcc8 && cd rccs-gcc8
../configure --prefix=${INSTALLDIR} \
  --with-tm=${PBSROOT} \
  --enable-mpi-cxx \
  --with-ucx \
  --enable-mpi1-compatibility
make -j ${PARALLEL} && make install && make check
cd ../

# GCC 11
module purge
module load gcc-toolset/11
```

```
INSTALLDIR=/apl/openmpi/${VERSION}/gcc11
mkdir rccs-gcc11 && cd rccs-gcc11
../configure --prefix=${INSTALLDIR} \
    --with-tm=${PBSROOT} \
    --enable-mpi-cxx \
    --with-ucx \
    --enable-mpi1-compatibility
make -j ${PARALLEL} && make install && make check
cd ../
```

```
# GCC 12
```

```
module purge
```

```
module load gcc-toolset/12
```

```
INSTALLDIR=/apl/openmpi/${VERSION}/gcc12
```

```
mkdir rccs-gcc12 && cd rccs-gcc12
```

```
../configure --prefix=${INSTALLDIR} \
```

```
    --with-tm=${PBSROOT} \
```

```
    --enable-mpi-cxx \
```

```
    --with-ucx \
```

```
    --enable-mpi1-compatibility
```

```
make -j ${PARALLEL} && make install && make check
```

```
cd ../
```

intel 2022 (classic)

```
#!/bin/sh
```

```
VERSION=4.1.6
```

```
WORKDIR=/gwork/users/${USER}
```

```
TARBALL=/home/users/${USER}/Software/OpenMPI/${VERSION}/openmpi-${VERSION}.tar.bz2
```

```
PBSROOT=/apl/pbs/22.05.11
```

```
PARALLEL=12
```

```
#-----
```

```
umask 0022
```

```
cd ${WORKDIR}
```

```
if [ -d openmpi-${VERSION} ]; then
```

```
    mv openmpi-${VERSION} openmpi-erase
```

```
    rm -rf openmpi-erase &
```

```
fi
```

```
tar jxf ${TARBALL}
```

```
cd openmpi-${VERSION}
```

```
# intel compiler classic 2022.2.1
```

```
module purge
```

```
. ~/intel/oneapi/compiler/2022.2.1/env/vars.sh
```

```
export CC=icc
```

```
export CXX=icpc
```

```
export FC=ifort
```

```
INSTALLDIR=/apl/openmpi/${VERSION}/intel2022
```

```
mkdir rccs-intel && cd rccs-intel
```

```
../configure --prefix=${INSTALLDIR} \
```

```
    --with-tm=${PBSROOT} \
```

```
    --enable-mpi-cxx \
```

```
    --with-ucx \
```

```
    --enable-mpi1-compatibility
```

```
make -j ${PARALLEL} && make install && make check
```

```
cd ../
```

intel 2023 (classic)

```

#!/bin/sh

VERSION=4.1.6
WORKDIR=/gwork/users/${USER}
TARBALL=/home/users/${USER}/Software/OpenMPI/${VERSION}/openmpi-${VERSION}.tar.bz2
PBSROOT=/apl/pbs/22.05.11

PARALLEL=12

#-----
umask 0022

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

tar jxf ${TARBALL}
cd openmpi-${VERSION}

# intel compiler classic 2023.2.0
module purge
. ~/intel/oneapi/compiler/2023.2.0/env/vars.sh

export CC=icc
export CXX=icpc
export FC=ifort

INSTALLDIR=/apl/openmpi/${VERSION}/intel2023
mkdir rccs-intel && cd rccs-intel
./configure --prefix=${INSTALLDIR} \
  --with-tm=${PBSROOT} \
  --enable-mpi-cxx \
  --with-ucx \
  --enable-mpi1-compatibility
make -j ${PARALLEL} && make install && make check
cd ../

```

intel 2023

```

#!/bin/sh

VERSION=4.1.6
WORKDIR=/gwork/users/${USER}
TARBALL=/home/users/${USER}/Software/OpenMPI/${VERSION}/openmpi-${VERSION}.tar.bz2
PBSROOT=/apl/pbs/22.05.11

PARALLEL=12

#-----
umask 0022

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

tar jxf ${TARBALL}
cd openmpi-${VERSION}

# intel compiler icx/icpx 2023.2.0
module purge
. ~/intel/oneapi/compiler/2023.2.0/env/vars.sh

```

```

export CC=icx
export CXX=icpx
export FC=ifx

INSTALLDIR=/apl/openmpi/${VERSION}/intel2023-icx
mkdir rccs-intel && cd rccs-intel
./configure --prefix=${INSTALLDIR} \
    --with-tm=${PBSROOT} \
    --enable-mpi-cxx \
    --with-ucx \
    --enable-mpi1-compatibility
make -j ${PARALLEL} && make install && make check
cd ../

```

aocc 4.1

```

#!/bin/sh

VERSION=4.1.6
WORKDIR=/gwork/users/${USER}
TARBALL=/home/users/${USER}/Software/OpenMPI/${VERSION}/openmpi-${VERSION}.tar.bz2
PBSROOT=/apl/pbs/22.05.11

PARALLEL=12

#-----
umask 0022

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

tar jxf ${TARBALL}
cd openmpi-${VERSION}

# AOCC 4.1.0
module -s purge
module -s load aocc/4.1.0

export CC=clang
export CXX=clang++
export FC=flang

INSTALLDIR=/apl/openmpi/${VERSION}/aocc4.1
mkdir rccs-aocc41 && cd rccs-aocc41
./configure --prefix=${INSTALLDIR} \
    --with-tm=${PBSROOT} \
    --enable-mpi-cxx \
    --with-ucx \
    --without-libfabric \
    --enable-mpi1-compatibility
make -j ${PARALLEL} && make install && make check
cd ../

```

nvidia hpc sdk 23.5

```

#!/bin/sh

VERSION=4.1.6
WORKDIR=/gwork/users/${USER}
TARBALL=/home/users/${USER}/Software/OpenMPI/${VERSION}/openmpi-${VERSION}.tar.bz2
PBSROOT=/apl/pbs/22.05.11

```

PARALLEL=12

```
#-----  
umask 0022  
  
cd ${WORKDIR}  
if [ -d openmpi-${VERSION} ]; then  
  mv openmpi-${VERSION} openmpi-erase  
  rm -rf openmpi-erase &  
fi  
  
tar jxf ${TARBALL}  
cd openmpi-${VERSION}  
  
# nvidia hpc sdk 22.11  
module purge  
module load nvhpc/23.5-nompi  
  
export CC=nvc  
export CXX=nvc++  
export FC=nvfortran  
  
export CFLAGS="-fPIC"  
export CXXFLAGS="-fPIC"  
export FCFLAGS="-fPIC"  
export LDFLAGS="-fPIC"  
  
export CUDA_HOME=${NVHPC_ROOT}/cuda  
  
INSTALLDIR=/apl/openmpi/${VERSION}/nv23  
mkdir rccs-nv23 && cd rccs-nv23  
./configure --prefix=${INSTALLDIR} \  
  --with-tm=${PBSROOT} \  
  --enable-mpi-cxx \  
  --with-ucx \  
  --with-cuda=${CUDA_HOME} \  
  --enable-mpi1-compatibility  
make -j ${PARALLEL} && make install && make check  
cd ../
```