

## iMolpro 1.0.1

### ウェブページ

<https://github.com/molpro/iMolpro>

### バージョン

1.0.1

### ビルド環境

-

### ビルドに必要なファイル

- Miniforge3-Linux-x86\_64.sh (miniforge installer)
- diff-projectwindow.py: csh/tcsh でも動かすための修正

```
--- ProjectWindow.py.org 2024-03-08 14:34:07.000000000 +0900
+++ ProjectWindow.py 2024-03-08 14:33:22.000000000 +0900
@@ -122,8 +122,12 @@
     settings['project_directory'] = os.path.dirname(self.project.filename(run=-1))

     if 'PATH' in os.environ and 'SHELL' in os.environ:
+         if re.search( 'csh', os.environ['SHELL'] ):
+             myopts = " -c 'echo $PATH'"
+         else:
+             myopts = " -l -c 'echo $PATH'"
         try:
-             os.environ['PATH'] = os.popen(os.environ['SHELL'] + " -l -c 'echo $PATH'").read() + ':' + os.environ[
+             os.environ['PATH'] = os.popen(os.environ['SHELL'] + myopts).read() + ':' + os.environ[
                'PATH'] # make PATH just as if running from shell
        except Exception as e:
            msg = QMessageBox()
```

### ビルド手順

```
#!/bin/sh

VERSION=1.0.1
#INSTALL_DIR=/apl/imolpro/${VERSION}

BASEDIR=/home/users/${USER}/Software/iMolpro/${VERSION}
MINIFORGE=${BASEDIR}/Miniforge3-Linux-x86_64.sh
IMOLPRO_GIT=https://github.com/molpro/iMolpro.git
IMOLPRO_TAG=refs/tags/${VERSION}

WORKDIR=/gwork/users/${USER}
WORKDIR_IMOLPRO=${WORKDIR}/imolpro
WORKDIR_BUILD=${WORKDIR_IMOLPRO}/build

PATCH=${BASEDIR}/diff-projectwindow.py

# -----
umask 0022
export LANG=C
export LC_ALL=C

module -s purge

if [ -d ${WORKDIR_IMOLPRO} ]; then
    mv ${WORKDIR_IMOLPRO} ${WORKDIR_IMOLPRO}_erase
    rm -rf ${WORKDIR_IMOLPRO}_erase &
```

```

fi
mkdir ${WORKDIR_IMOLPRO} && cd ${WORKDIR_IMOLPRO}

sh ${MINIFORGE} -b -f -p ${WORKDIR_IMOLPRO}/miniforge3

eval "$(${WORKDIR_IMOLPRO}/miniforge3/bin/conda shell.bash hook)"

cd ${WORKDIR}
if [ -d iMolpro ]; then
  mv iMolpro iMolpro-erase
  rm -rf iMolpro-erase &
fi
git clone ${IMOLPRO_GIT}
cd iMolpro
git checkout ${IMOLPRO_TAG}

# /tmp is not only for you!
export TMPDIR=${WORKDIR_BUILD}
mkdir -p ${TMPDIR}
sed -i -e "s/\tmp//g" iMolpro.py database.py
# for csh... is it ok?
patch -p0 < $PATCH
# ad hoc workaround necessary only in RCCS (regarding token file protection)
conda install -c conda-forge -y --file=requirements.txt || exit 1
sed -i -e "s/[\vbin\sh\]/[" ${WORKDIR_IMOLPRO}/miniforge3/lib/python3.*/site-packages/pymolpro/project.py

sh ./build.sh
# install dist/iMolpro-1.0.1-dirty.Linux.x86_64.tar.bz2 into $INSTALL_DIR manually

```

## メモ

- RCCS で利用するには X の転送が必要です。
  - Windows なら MobaXterm など、mac ならば Xquartz, Linux ならば ssh 接続時に X の forwarding を有効にするなどの準備が必要です。
  - 内蔵 JMol で分子の絵を表示することも可能なようです。
- 独自の修正(csh での動作と /tmp の利用回避)を行っているため、1.0.1-dirty というバージョン名が表示されます。
- molpro コマンドが PATH 中に見つかれば guided mode が有効化されます。
  - guided mode では toggle box など設定を指定可能です。
- 現時点では iMolpro から jsub でジョブを投入する方法は準備していません。
  - (backend の設定を行い、ジョブ投入用のユーティリティを用意すれば可能だとは思われます。公式サイト doc/example.md に例があります。)
  - インプットの作成とログインサーバでの簡単なテスト計算は現時点でも実行できます。
- RCCS では imolpro のモジュールを読み込めば、imolpro-molpro2023.2.0 のようなコマンドで molpro のバージョンを指定した形で iMolpro を起動できるように設定しています。
  - imolpro-molpro2023.2.0 (molpro 2023.2.0)
  - imolpro-molpro2022.3.0 (molpro 2022.3.0)
  - imolpro-molpro2021.3.1 (molpro 2021.3.1)
  - imolpro-molpro2024.1.0 (molpro 2024.1.0; 近日中に利用可能になる予定)