

How to compile PeakFQ on Linux

First, [install the Intel Fortran Compiler](#).

You can download the incomplete source code of PeakFQ from [its website](#). Here, I'll use [PeakFQ_7.5.1.src.zip](#).

[compile_peakfq.sh](#)

```
#!/bin/sh

# download source code
wget
https://water.usgs.gov/software/PeakFQ/code/7.5.1/PeakFQ_7.5.1.src.zip

# unzip it
mkdir -p ~/usr/local/src
unzip PeakFQ_7.5.1.src.zip -d ~/usr/local/src
cd ~/usr/local/src
mv PeakFQ_7.5.1.src/src/FORTRAN peakfq
rm -rf PeakFQ_7.5.1.src
cd peakfq

# rename all filenames to lowercase
for i in *.*; do
  j=$(echo $i | tr A-Z a-z)
  [ $i = $j ] || mv $i $j
done

# rename include and module names in source files to lowercase
for i in *.*; do
  grep -q ".*.INC" $i || continue
  for j in $(grep ".*.INC" $i | sort -u | fromdos | sed
"s/^[^']*\\'|$/g"); do
    k=$(echo $j | tr A-Z a-z)
    sed -i "s/$j/$k/" $i
  done
done

# move DENYNONE from ACTION to SHARE
sed -Ei "s/(, )(DENYNONE)/'\1SHARE='\2/" wdoppc90.for

# comment out non-existent modules and data type
sed -Ei 's/^(.*(KERNEL32|T_OVERLAPPED))/!\1/' scenmod.f90

# comment out Windows functions
sed -Ei 's/^(.*(PeekNamedPipe|ReadFile|WriteFile))/!\1/' scenmod.f90

# fix disclaimer line
```

```
sed -i "/^ *WRITE(DISCLM(1)/a \           DISCLM(2) = '\r" j407xe.for

# reenable default gen skew computation
sed -Ei 's/^C(. *GENSKU *=.*)/\1/; s/^Cprh( {6}AUX)/\1/; s/^(
{6}AUX\(\1\) = -999)/C\1/' j407xe.for

# download missing files
for i in \
  adwdm/cfbuff.inc \
  adwdm/cdrloc.inc \
  adwdm/fmsgwd.inc \
  adwdm/utwdmd.for \
  adwdm/utwdmf.for \
  adwdm/utwdt1.for \
  adwdm/wdatm1.for \
  adwdm/wdmchk.for \
  adwdm/wdmess.for \
  wdm/ctsbuf.inc \
  wdm/cwdmid.inc \
  wdm/cwtsds.inc \
  wdm/tsbufr.for \
  wdm/wdatm2.for \
  wdm/wdatrb.for \
  wdm/wdatru.for \
  wdm/wdbtch.for \
  wdm/wdmid.for \
  wdm/wdtms1.for \
  wdm/wdtms2.for \
; do
  wget
  https://svn.oss.deltares.nl/repos/openda/trunk/model_hspf/fortran/liban
  ne4.0/src/$i
done

# dos to unix
for i in * */*; do
  sed -i 's/\r//' $i
done

# create main.f90
cat << 'EOT' > main.f90
character(len=256) :: specfile

if(command_argument_count().eq.0) then
  write(*,*) "Usage: peakfq specfile"
else
  call get_command_argument(1, specfile)
  call peakfq(specfile)
endif
end
```

```
EOT

# create Makefile
cat << 'EOT' > Makefile
FC=ifx
LDFLAGS=-nofor-main

all: peakfq

clean:
    $(RM) *.o EMAUtil/*.o *.mod peakfq

peakfq: \
    main.o \
    EMAUtil/dcdflib1.o \
    EMAUtil/imslfake.o \
    EMAUtil/probfun.o \
    compspecs.o \
    datsys90.o \
    emafit.o \
    emathresh.o \
    j407wc.o \
    j407xe.o \
    ktutil.o \
    peakfq.o \
    pkfqsta.o \
    pkwdm.o \
    qfdprs.o \
    scenmod.o \
    stationdata.o \
    stgaus.o \
    stutil.o \
    tsbufr.o \
    utchar.o \
    utcpgn.o \
    utdate.o \
    utgnrl.o \
    utj407.o \
    utnumb.o \
    utstat.o \
    utwdmd.o \
    utwdmf.o \
    utwdt1.o \
    wdatm1.o \
    wdatm2.o \
    wdatrb.o \
    wdatru.o \
    wdbtch.o \
    wdmchk.o \
    wdmess.o \
    wdmid.o \
```

```
wdoppc90.o \  
wdpeak.o \  
wdtble.o \  
wdtms1.o \  
wdtms2.o  
$(FC) $(LDFLAGS) -o $@ $^  
  
peakfq.o: emathresh.o compspecs.o  
  
pkfqsta.o: scenmod.o  
  
j407xe.o: emathresh.o stationdata.o  
  
j407wc.o: emathresh.o  
  
%.o: %.f90  
    $(FC) $(FFLAGS) -c -o $@ $<  
  
%.o: %.for  
    $(FC) $(FFLAGS) -c -o $@ $<  
EOT  
  
# build  
make  
  
# install PeakFQ for Windows; find and copy pkfqms.wdm in lowercase  
# download it from this website  
wget https://clawiki.isnew.info/\_media/howtos/pkfqms.wdm
```

You don't need to run this command, but I used it to find undefined references:

```
# see what symbols are undefined  
for i in $(make &> /dev/stdout | grep "undefined reference" | sed 's/.*`//';  
s/_.*`//' | sort -u); do  
    if ! grep -qiE "(function|subroutine) *$i" *.* */*.*; then  
        echo "$i: NOT FOUND"  
    fi  
done
```

From:
<https://clawiki.isnew.info/> - **CLAWRIM Wiki**

Permanent link:
https://clawiki.isnew.info/howtos/how_to_compile_peakfq_on_linux?rev=1727155487

Last update: **2024-09-23 11:24 pm**

