Guide to Clawpack

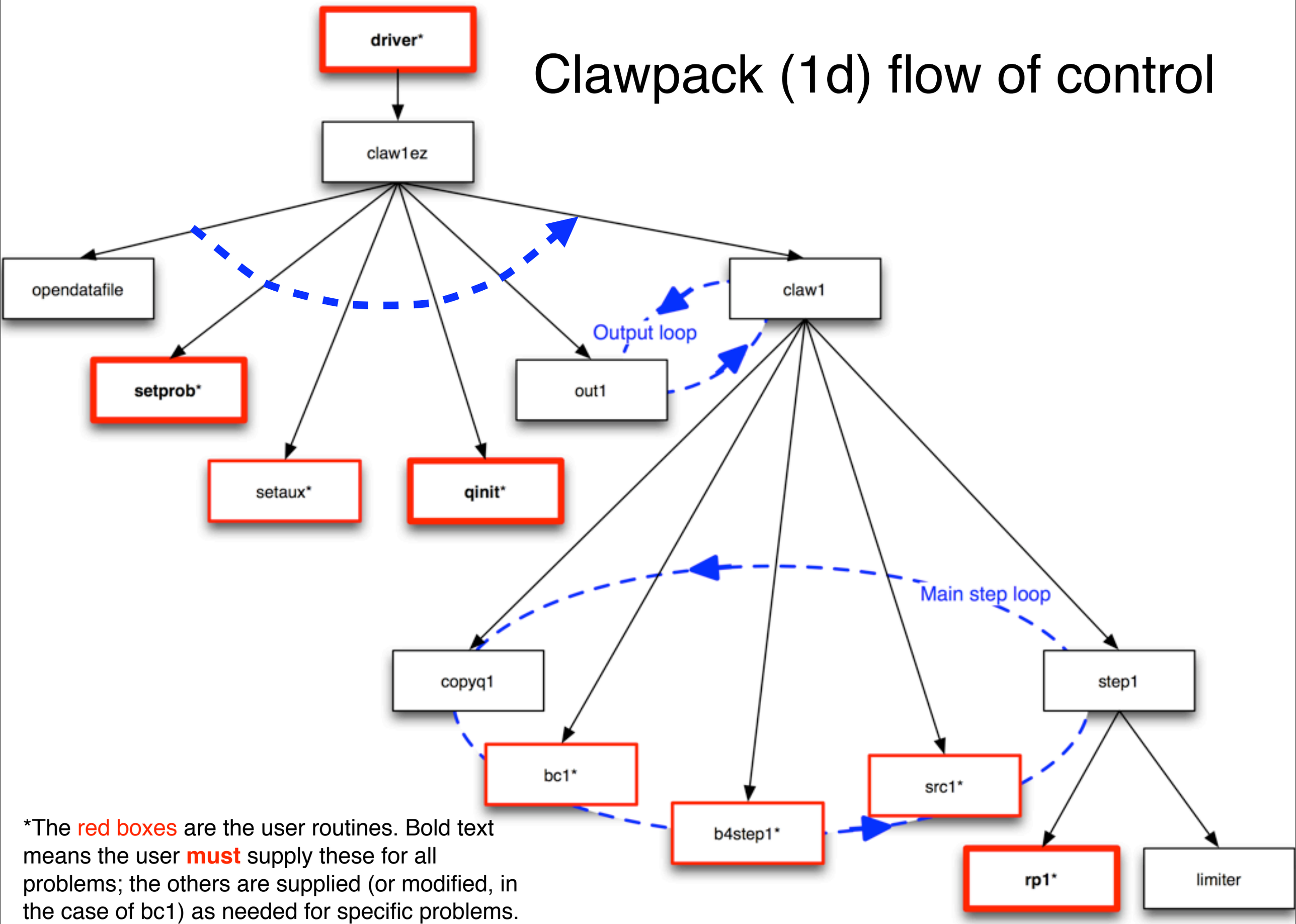
Galen Gisler, PGP
FYS-GEO 4500

Program flow -1d - Fortran version

```
driver
     call claw1ez
        call opendatafile
        call setprob*
        call setaux*
        call qinit*
        call out1
        loop on output steps
           call claw1
              loop on single steps
                 call copyq1 (if variable time step)
                 call bc1*
                 call b4step1*
                 call src1* (if using Strang splitting)
                 call step1
                   call rp1*
                   call limiter
                 call src1* (if needed)
                 call copyq1 (if needed)
              end loop on single steps
           call out1
        end loop on output steps
     end
end
```

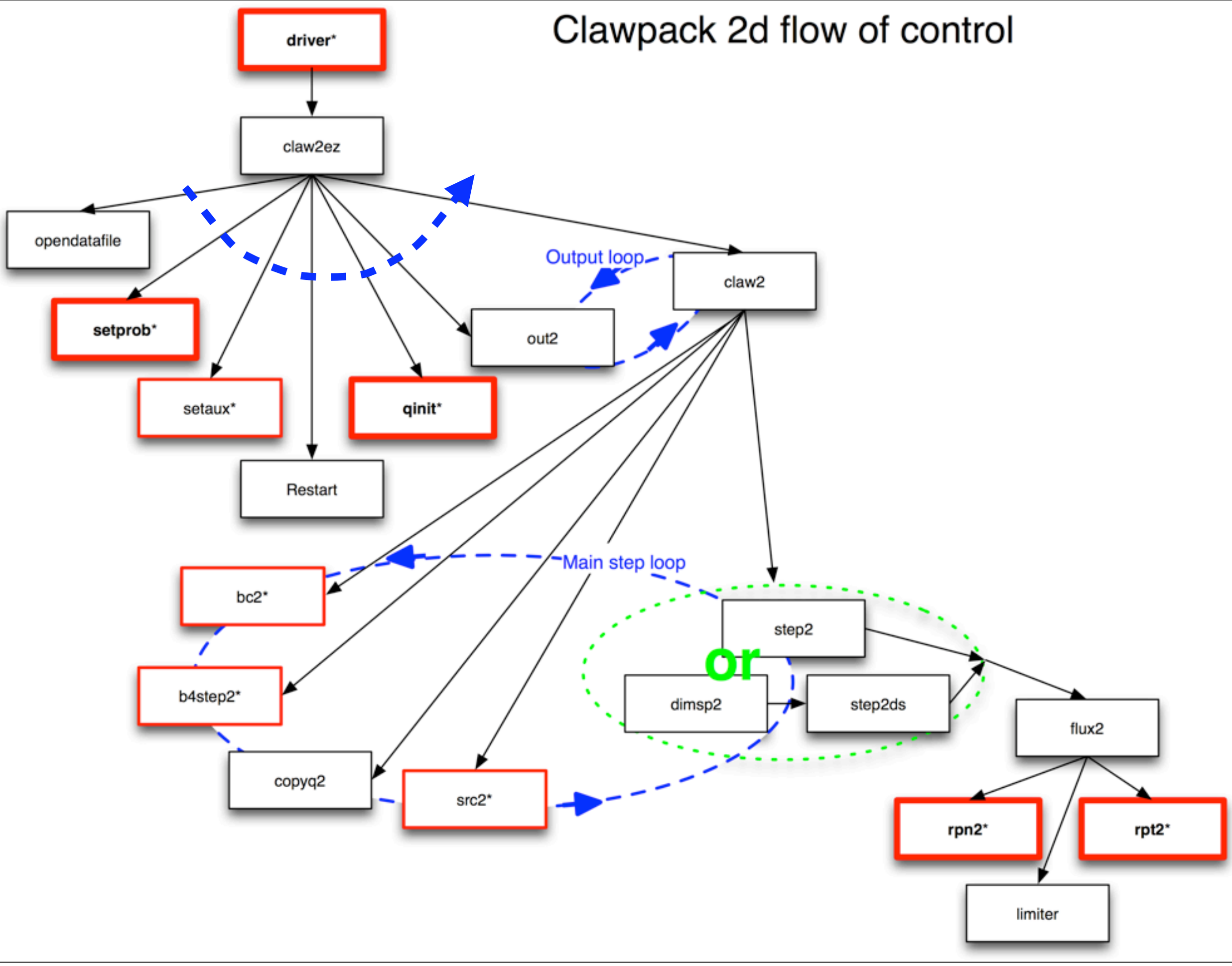
*The red names are the user routines. **Bold text** means the user must supply these for all problems, others are supplied (or modified, in the case of bc1) as needed for specific problems.

Clawpack (1d) flow of control

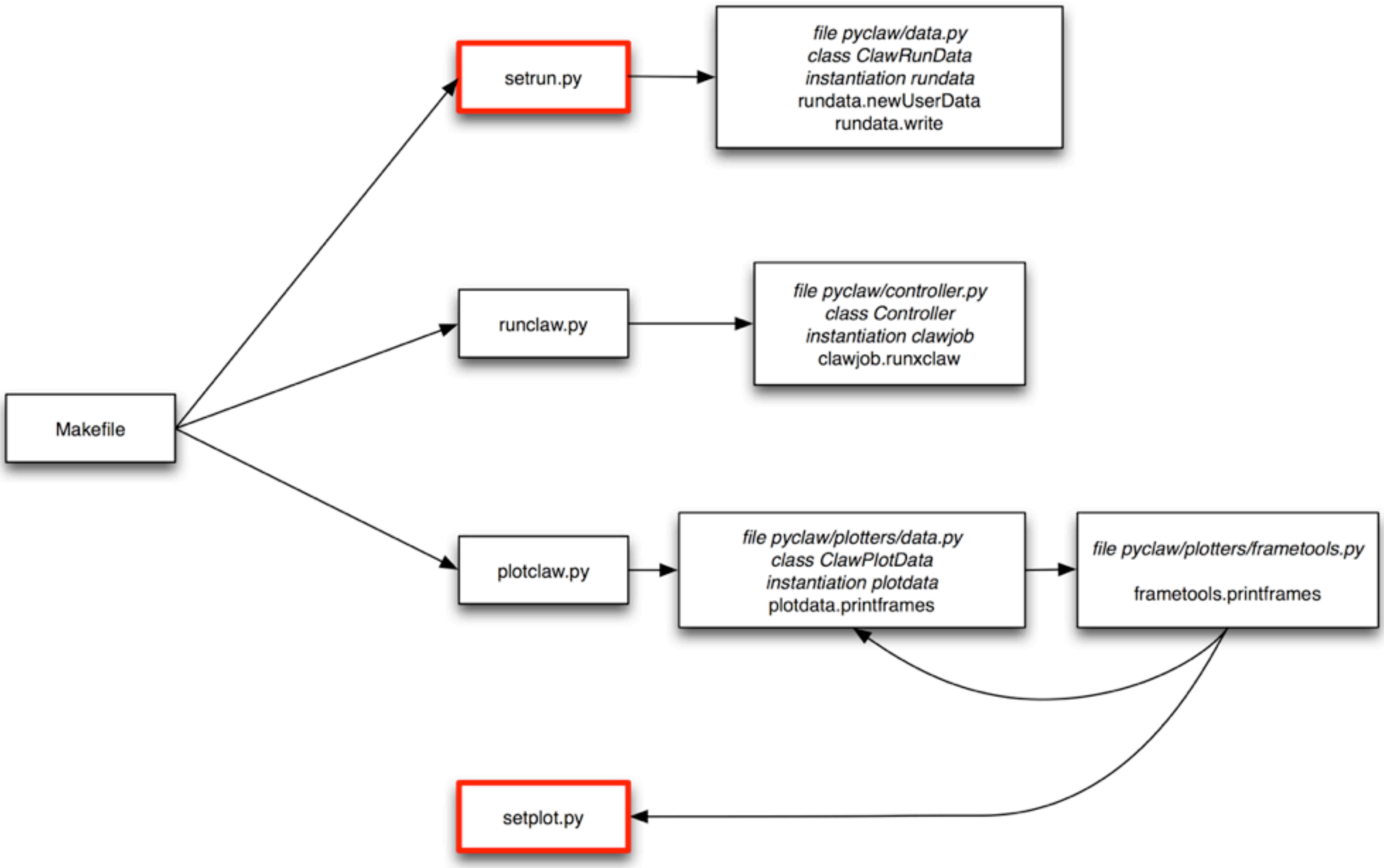


*The red boxes are the user routines. Bold text means the user **must** supply these for all problems; the others are supplied (or modified, in the case of bc1) as needed for specific problems.

Clawpack 2d flow of control



What Makefile does



Your Makefile must point to the appropriate source code files

```
#
# List of sources for this program:
#
CLAW_SOURCES = \
  driver.f \
  qinit.f \
  rpn2.f \
  rpt2.f \
  setprob.f

# Clawpack library to be used:
CLAW_LIB = $(CLAW)/clawpack/2d/lib

CLAW_LIBSOURCES = \
  $(CLAW_LIB)/claw2ez.f \
  $(CLAW_LIB)/bc2.f \
  $(CLAW_LIB)/setaux.f \
  $(CLAW_LIB)/b4step2.f \
  $(CLAW_LIB)/claw2.f \
  $(CLAW_LIB)/step2.f \
  $(CLAW_LIB)/step2ds.f \
  $(CLAW_LIB)/dimsp2.f \
  $(CLAW_LIB)/flux2.f \
  $(CLAW_LIB)/copyq2.f \
  $(CLAW_LIB)/limiter.f \
  $(CLAW_LIB)/philim.f \
  $(CLAW_LIB)/src2.f\
  $(CLAW_LIB)/out2.f\
  $(CLAW_LIB)/restart2.f\
  $(CLAW_LIB)/opendatafile.f
```

These are the files you write (or change) yourself. Include other files like bc2.f, b4step2.f, src2.f, and so on as needed. These should be in your run directory under $CLAW/myclaw (as this Makefile is).

These are the files you use from the Clawpack library referenced in the line above (2d or 1d are available now; 3d will come later).

If you want to make changes to any of these, make a copy first and move the copy to your run directory under `$CLAW/myclaw`. You may change it there, but to use it you must add its name to the list above and remove it from this list.

What Makefile does

In $CLAW/util/Makefile.common (which should be invoked by every Clawpack Makefile):

```
#       Makefile for the clawpack code:
#
#       For this help summary, type:            make .help
#
#       To make all object files, type:         make .objs
#
#       To compile a single file.f:             make file.o
#
#       To make the executable, type:           make .exe
#
#       To make data files by running
#       setrun.py:                              make .data
#
#       To make and run code putting results
#       in subdirectory named output:           make .output
#
#       To make and run code and then plot
#       results from subdirectory output
#       into subdirectory named plots:          make .plots
#
#       To create html files from the program
#       and data files using clawcode2html:     make .htmls
#
#       To clean up files created by make:      make clean
#       Deletes *.o, x*, .htmls
#
#       To clean up output and graphics files:  make clobber
#
```

setrun.py

setplot.py

What Makefile does

In $CLAW/util/Makefile.common:

```
# Executable:
.objs: $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS)
$(CLAW_EXE): $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS)
	$(LINK) $(LFLAGS) $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS) -o $(CLAW_EXE)
.exe: $(CLAW_EXE) ;
```

```
# Make data files needed by Fortran code:
.data: $(CLAW_setrun_file) ;
	python $(CLAW_setrun_file) $(CLAW_PKG)
	touch .data
```

```
# Run the code and put fort.* files into subdirectory named output:
# runclaw will execute setrun.py to create data files and determine
# what executable to run, e.g. xclaw or xamr.
.output: $(CLAW_EXE) .data;
	python $(CLAW)/python/pyclaw/runclaw.py  $(CLAW_EXE) $(CLAW_OUTDIR)
	@echo $(CLAW_OUTDIR) > .output
```

```
# Rule to make the plots into subdirectory specified by CLAW_PLOTDIR,
# using data in subdirectory specified by CLAW_OUTDIR and the plotting
# commands specified in CLAW_setplot_file.
.plots: .output $(CLAW_setplot_file) ;
	$(PLOTCMD) $(CLAW_OUTDIR) $(CLAW_PLOTDIR) $(CLAW_setplot_file)
	@echo $(CLAW_PLOTDIR) > .plots
```

```
CLAW_PKG = Classic                  # Clawpack package to use
CLAW_EXE = xclaw                    # Executable to create
CLAW_setrun_file = setrun.py        # File containing function to make data
CLAW_OUTDIR = _output               # Directory for output
CLAW_setplot_file = setplot.py      # File containing function to set plots
CLAW_PLOTDIR = _plots               # Directory for plots
PLOTCMD := python $(CLAW)/python/pyclaw/plotters/plotclaw.py
```

What Makefile does

In $CLAW/util/Makefile.common:

```
# Executable:
.objs: $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS)
$(CLAW_EXE): $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS)
     $(LINK) $(LFLAGS) $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS) -o $(CLAW_EXE)
.exe: $(CLAW_EXE) ;
```

```
# Make data files needed by Fortran code:
.data: $(CLAW_setrun_file) ;
     python $(CLAW_setrun_file) $(CLAW_PKG)
     touch .data
```

```
# Run the code and put fort.* files into subdirectory named output:
# runclaw will execute setrun.py to create data files and determine
# what executable to run, e.g. xclaw or xamr.
.output: $(CLAW_EXE) .data;
     python $(CLAW)/python/pyclaw/runclaw.py  $(CLAW_EXE) $(CLAW_OUTDIR)
     @echo $(CLAW_OUTDIR) > .output
```

Rule

```
# Rule to make the plots into subdirectory specified by CLAW_PLOTDIR,
# using data in subdirectory specified by CLAW_OUTDIR and the plotting
# commands specified in CLAW_setplot_file.
.plots: .output $(CLAW_setplot_file) ;
     $(PLOTCMD) $(CLAW_OUTDIR) $(CLAW_PLOTDIR) $(CLAW_setplot_file)
     @echo $(CLAW_PLOTDIR) > .plots
```

```
CLAW_PKG = Classic                  # Clawpack package to use
CLAW_EXE = xclaw                    # Executable to create
CLAW_setrun_file = setrun.py        # File containing function to make data
CLAW_OUTDIR = _output               # Directory for output
CLAW_setplot_file = setplot.py      # File containing function to set plots
CLAW_PLOTDIR = _plots               # Directory for plots
PLOTCMD := python $(CLAW)/python/pyclaw/plotters/plotclaw.py
```

Target Requirements

What Makefile does

In $CLAW/util/Makefile.common:

```
# Executable:
.objs: $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS)
$(CLAW_EXE): $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS)
	$(LINK) $(LFLAGS) $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS) -o $(CLAW_EXE)
.exe: $(CLAW_EXE) ;
```

```
# Make data files needed by Fortran code:
.data: $(CLAW_setrun_file) ;
	python $(CLAW_setrun_file) $(CLAW_PKG)
	touch .data
```

```
# Run the code and put fort.* files into subdirectory named output:
# runclaw will execute setrun.py to create data files and determine
# what executable to run, e.g. xclaw or xamr.
.output: $(CLAW_EXE) .data;
	python $(CLAW)/python/pyclaw/runclaw.py  $(CLAW_EXE) $(CLAW_OUTDIR)
	@echo $(CLAW_OUTDIR) > .output
```

Rule

```
# Rule to make the plots into subdirectory specified by CLAW_PLOTDIR,
# using data in subdirectory specified by CLAW_OUTDIR and the plotting
# commands specified in CLAW_setplot_file.
.plots: .output $(CLAW_setplot_file) ;
	$(PLOTCMD) $(CLAW_OUTDIR) $(CLAW_PLOTDIR) $(CLAW_setplot_file)
	@echo $(CLAW_PLOTDIR) > .plots
```

```
CLAW_PKG = Classic                  # Clawpack package to use
CLAW_EXE = xclaw                    # Executable to create
CLAW_setrun_file = setrun.py        # File containing function to make data
CLAW_OUTDIR = _output               # Directory for output
CLAW_setplot_file = setplot.py      # File containing function to set plots
CLAW_PLOTDIR = _plots               # Directory for plots
PLOTCMD := python $(CLAW)/python/pyclaw/plotters/plotclaw.py
```

What Makefile does

In $CLAW/util/Makefile.common:

```
# Executable:
.objs: $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS)
$(CLAW_EXE): $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS)
     $(LINK) $(LFLAGS) $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS) -o $(CLAW_EXE)
.exe: $(CLAW_EXE) ;

# Make data files needed by Fortran code:
.data: $(CLAW_setrun_file) ;
     python $(CLAW_setrun_file) $(CLAW_PKG)
     touch .data

# Run the code and put fort.* files into subdirectory named output:
# runclaw will execute setrun.py to create data files and determine
# what executable to run, e.g. xclaw or xamr.
.output: $(CLAW_EXE) .data;
     python $(CLAW)/python/pyclaw/runclaw.py  $(CLAW_EXE) $(CLAW_OUTDIR)
     @echo $(CLAW_OUTDIR) > .output

# Rule to make the plots into subdirectory specified by CLAW_PLOTDIR,
# using data in subdirectory specified by CLAW_OUTDIR and the plotting
# commands specified in CLAW_setplot_file.
.plots: .output $(CLAW_setplot_file) ;
     $(PLOTCMD) $(CLAW_OUTDIR) $(CLAW_PLOTDIR) $(CLAW_setplot_file)
     @echo $(CLAW_PLOTDIR) > .plots

CLAW_PKG = Classic                  # Clawpack package to use
CLAW_EXE = xclaw                    # Executable to create
CLAW_setrun_file = setrun.py        # File containing function to make data
CLAW_OUTDIR = _output               # Directory for output
CLAW_setplot_file = setplot.py      # File containing function to set plots
CLAW_PLOTDIR = _plots               # Directory for plots
PLOTCMD := python $(CLAW)/python/pyclaw/plotters/plotclaw.py
```

Command

Rule

What Makefile does

In $CLAW/util/Makefile.common:

```
# Executable:
.objs: $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS)
$(CLAW_EXE): $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS)
     $(LINK) $(LFLAGS) $(CLAW_OBJECTS) $(CLAW_LIBOBJECTS) -o $(CLAW_EXE)
.exe: $(CLAW_EXE) ;
```

```
# Make data files needed by Fortran code:
.data: $(CLAW_setrun_file) ;
     python setrun.py   Classic
     touch .data
```

```
# Run the code and put fort.* files into subdirectory named output:
# runclaw will execute setrun.py to create data files and determine
# what executable to run, e.g. xclaw or xamr.
.output: $(CLAW_EXE) .data;
     python $CLAW/python/pyclaw/runclaw.py    xclaw    _output
     @echo $(CLAW_OUTDIR) > .output
```

```
# Rule to make the plots into subdirectory specified by CLAW_PLOTDIR,
# using data in subdirectory specified by CLAW_OUTDIR and the plotting
# commands specified in CLAW_setplot_file.
.plots: .output $(CLAW_setplot_file) ;
     python $CLAW/python/pyclaw/plotters/plotclaw.py    _output    _plots    setplot.py
     @echo $(CLAW_PLOTDIR) > .plots
```

```
CLAW_PKG = Classic                    # Clawpack package to use
CLAW_EXE = xclaw                      # Executable to create
CLAW_setrun_file = setrun.py          # File containing function to make data
CLAW_OUTDIR = _output                 # Directory for output
CLAW_setplot_file = setplot.py        # File containing function to set plots
CLAW_PLOTDIR = _plots                 # Directory for plots
PLOTCMD := python $(CLAW)/python/pyclaw/plotters/plotclaw.py
```

What Makefile does

In $CLAW/util/Makefile.common:

setrun.py and **setplot.py** should be defined in the run directory and can be changed by you

runclaw.py and **plotclaw.py** are defined in $CLAW/python/pyclaw and should not be changed

```
# Make data files needed by Fortran code:
.data: $(CLAW_setrun_file) ;
	python setrun.py   Classic
	touch .data
```

```
# Run the code and put fort.* files into subdirectory named output:
# runclaw will execute setrun.py to create data files and determine
# what executable to run, e.g. xclaw or xamr.
.output: $(CLAW_EXE) .data;
	python $CLAW/python/pyclaw/runclaw.py    xclaw    _output
	@echo $(CLAW_OUTDIR) > .output
```

```
# Rule to make the plots into subdirectory specified by CLAW_PLOTDIR,
# using data in subdirectory specified by CLAW_OUTDIR and the plotting
# commands specified in CLAW_setplot_file.
.plots: .output $(CLAW_setplot_file) ;
	python $CLAW/python/pyclaw/plotters/plotclaw.py    _output    _plots    setplot.py
	@echo $(CLAW_PLOTDIR) > .plots
```

```
CLAW_PKG = Classic                  # Clawpack package to use
CLAW_EXE = xclaw                    # Executable to create
CLAW_setrun_file = setrun.py        # File containing function to make data
CLAW_OUTDIR = _output               # Directory for output
CLAW_setplot_file = setplot.py      # File containing function to set plots
CLAW_PLOTDIR = _plots               # Directory for plots
PLOTCMD := python $(CLAW)/python/pyclaw/plotters/plotclaw.py
```

What about Python?

Directory structure of $CLAW/python/pyclaw:

Makefile
README.txt

__init__.py
controller.py
data.py
runclaw.py
solution.py
util.py

} controllers, etc., for Fortran **or** Python version of clawpack

evolve
- __init__.py
- clawpack.py
- limiters.py
- solver.py
- rp
  - __init__.py
  - rp_acoustics.py
  - rp_advection.py
  - rp_burgers.py
  - rp_euler.py
  - rp_shallow.py

} Python version of clawpack

io
- __init__.py
- ascii.py
- hdf5.py
- netcdf.py

} input/output routines for Python version of clawpack

plotters
- __init__.py
- lplotclaw.py
- colormaps.py
- data.py
- frametools.py
- multiframetools.py
- plotclaw.py
- plotpages.py
- TODO

} plotting routines for Fortran **or** Python version of clawpack

More information about using Python with Clawpack, especially the plotting stuff, available from:

http://kingkong.amath.washington.edu/claw4/www/users/

http://kingkong.amath.washington.edu/claw4/www/users/plotting.html#plotting

http://kingkong.amath.washington.edu/claw4/www/users/pyclaw/index.html#pyclaw