Q
Version 5.7
Developers Manual

Mauricio Esguerra
mauricio.esguerra@gmail.com

Contents

1 Introduction

This is the developers manual for **Q**, the free energy from molecular dynamics program.
Q is entirely written in modern **FORTRAN**. Its initial development was in **FORTRAN** 90, but the incorporation of object oriented paradigms into modern **FORTRAN** have seen the code evolve to take advantage of the evolution of the language. Currently the program incorporporates elements of the **FORTRAN** 2008 standard and it's developing the use of the OpenMP 4.0 standard[i] for in-node parallelization.

1.1 Makefile

The makefile follows closely the **GNU make** manual (`http://www.gnu.org/software/make/manual/make.html`). Any modifications to the makefile should make sure that user-space is not broken, that is, **Q** should compile on any of the main operating systems and avoid as much as possible hardware specific compiler options.

[i] which is now fully implemented into gnu Fortran version 5.1

1.2 Cluster Architectures

1.3 Debugging

1.4 Profiling

1.5 Best Practices

http://www.fortran90.org/src/best-practices.html

2 ChangeLog

3 Release Schedule

4 qprep

qprep is the utility which allows the translation of pdb files into a format which the dynamics engine can understand. The main file generated by **qprep** is refered to as the topology.

```
qprep generate.inp >& generate.out &
```

5 qdum

6 qdyn

qdyn is compiled using the following explicit rules:

```
-ffree-line-length-none
-fcray-pointer
-fall-intrinsics
-std=legacy
-Wall
-Wtabs
-fstack-protector
-O3
-DG95=1 -c
```

Development flags recommended from fortran90.org:
-Wall -Wextra -Wimplicit-interface -fPIC -fmax-errors=1 -g -fcheck=all -fbacktrace

Production flags recommended from fortran90.org:
-Wall -Wextra -Wimplicit-interface -fPIC -Werror -fmax-errors=1 -O3 -march=native -ffast-math -funroll-loops

sizes: Specifies data storage for all Q programs.
Modules used= NONE
nrgy: Input/Output for energy data and energy file.
Modules used= sizes
Subroutines:
nrgy_startup = empty
put_ene =
Functions:
mpiglob: Declares global variables for MPI parallelization of qdyn Modules used = nrgy Subroutines:
Functions:
misc:
Modules used = sizes
Subroutines:
Functions:
prmfile: Data files parser. Modules used = misc Subroutines: prmfile_startup = EMPTY ? WTF
Functions:
index: Modules used = NONE

topo: Modules used = sizes, misc Subroutines: topo_startup = EMPTY ? WTF
Functions:
qatom: Modules used = sizes, nrgy, misc, prmfile, indexer, topo
mask: Modules used = topo
trj: Modules used = atom_mask, misc
-DG95=1 -cpp -c
md: This is the main molecular dynamics *module*. Modules used =
qdyn: This file contains the main **qdyn** *program* code. Modules used =

7 qfep

8 qcalc

9 Tricks

A very convenient trick for cleaning up the code is:
emacs –batch md.f90 -f mark-whole-buffer -f f90-indent-subprogram -f save-buffer

10 Links

The following is a collection of reference links useful for fotran programers.
- The description of the GNU Compiler Collection (gcc) version 5.1.0
`https://gcc.gnu.org/onlinedocs/gcc-5.1.0/gfortran.pdf`

10.1 Doxygen

To automatically document fortran code the only current available option is doxygen. To use doxygen a configuration file is necessary. This has been downloaded from:

`https://modelingguru.nasa.gov/docs/DOC-1811`
And modified to our needs.
Just doing:

```
doxygen DoxygenConfigFortran
```

Generates html code which we make available at:
doxygen.qdyn.org
A mainfile document is included in the file maintext.txt

10.2 Fortran codes available online

These projects are a good resource to check-out development and good practices on coding. Even though FORTRAN does not enforce syntax rules as Python does, the user should use them so that the code is readable. An analog of the sytax rules for Python, frequently refered to as PEP8, would be very useful for FORTRAN programers.

Jason Blevins. Benchmarking ansi C and FORTRAN.
`http://jblevins.org/git/scicomp.git/tree/`

Gyro-Kinetics at Warwick. A program for turbulence study in plasmas.
`https://bitbucket.org/gkw/gkw/wiki/Home`

OpenMC. A Monte Carlo code.
`https://github.com/mit-crpg/openmc`

A collection of generic Fortran routines.
`https://github.com/astrofrog/fortranlib`