GFDL Vortex Tracker

Users Guide Version 3.9a

April 2018

Mrinal K. Biswas, Donald Stark and Laurie Carson
National Center for Atmospheric Research and Developmental Testbed Center

Acknowledgements

Timothy Marchok
Geophysical Fluid Dynamics Laboratory

Ligia Bernardet
University of Colorado Cooperative Institute for Research in Environmental Sciences at the NOAA Earth System Research Laboratory/Global Systems Division and Developmental Testbed Center

Please send questions to: hwrf-help@ucar.edu

DTC Developmental Testbed Center

Acknowledgement

For referencing this document please use:

Biswas, M. K., D. Stark, and L. Carson, 2018: GFDL Vortex Tracker Users' Guide V3.9a, 35 pp.

Table of Contents

1

Introduction

1.1 GFDL Vortex Tracker Overview

The standalone Geophysical Fluid Dynamics Laboratory (GFDL) vortex tracker is a program that objectively analyzes forecast data to provide an estimate of the vortex center position (latitude and longitude), and track the storm for the duration of the forecast. Additionally, it reports metrics of the forecast storm, such as intensity (maximum 10-m winds and minimum mean sea level pressure (MSLP)), structure (wind radii for 34, 50 and 64 knot thresholds in each quadrant), radius of the outermost closed isobar (ROCI), and optionally integrated kinetic energy (IKE), storm surge damage potential (SDP) and cyclone thermodynamic phase at each output time. The software can track either storms that exist at model initialization time or storms that develop during the forecast (cyclogenesis).

The GFDL vortex tracker requires the forecast grids to be on a cylindrical equidistant, latitude-longitude (lat/lon) grid. Forecast files can be in GRIB (GRIB1 or GRIB2) or, starting with v3.9a, NetCDF format. When the forecast is in GRIB format, the vortex tracker also ingests a GRIB index file, which is generated by running the program *grbindex* (*grb2index.exe*) for GRIB1 (GRIB2) input files. For NetCDF files, no index files are needed. The programs *grbindex* and *grb2index* were developed by the National Centers for Environmental Prediction (NCEP) and are distributed by the DTC as part of this GFDL vortex tracker release. The output files contain the vortex position, intensity and structure information in original Automated Tropical Cyclone Forecast (ATCF) and modified formats.

The fields used by the tracker are (i) relative vorticity at 10m, 850 hPa and 700 hPa, (ii) Mean Sea Level Pressure (MSLP), (iii) geopotential height at 850 and 700 hPa, (iv) wind speed at 10 m, 850 hPa and 700 hPa, and (v) 200-500 hPa and 500-850 hPa thickness. These parameters are used in order to provide accurate position estimates for both strong and weak storms, which often have poorly defined structures/centers. The GFDL tracker code is able to function when certain input fields are missing or when certain alternate fields are provided instead of the primary fields. However, the Developmental Testbed Center (DTC) will only provide user support for running the tracker with all the fields except 200-500 hPa and 500-850 hPa thickness listed above. Advanced users may see subroutine *getdata* in the source code to understand how the tracker can be used with alternate fields.

The identification of cyclone thermodynamic phases requires that the input data also contain average temperature between 300 and 500 hPa (for the *vtt* scheme) or the geopotential height every 50 hPa from 300 to 900 hPa (for the *cps* scheme) (see Section 4.1). The *vint* and *tave* utilities are provided by the DTC to compute these quantities from the other fields used for tracking, as described in the previous paragraph.

For cyclogenesis, users can opt between providing as input all the fields usually used for tracking or just MSLP, which may be more appropriate for tracking extratropical cyclogenesis (see Section 4.2).

1.2 Support

More information about the scientific aspects of the GFDL vortex tracker can be found in the Hurricane Weather Research and Forecasting (HWRF) scientific documentation at http://www.dtcenter.org/HurrWRF/users/docs/scientific_documents/HWRFv3.9a_ScientificDoc.pdf. Please send all questions to hwrf-help@ucar.edu.

2

Installation

2.1 Downloading the GFDL Vortex Tracker Code

The standalone GFDL vortex tracker can be obtained through the HWRF Users website

http://www.dtcenter.org/HurrWRF/users/downloads/index.tracker.php

New users must first register before downloading the source code. Returning users need only provide their registration email address. A successful download results in two compressed tar files,

standalone_gfdl-vortextracker_v3.9a.tar.gz and *test_data.tar.gz*

The tar files can be unpacked by use of the GNU command, gunzip

```
        gunzip *.tar.gz
```

and the tar files may be extracted by running *tar –xvf* on each of the tar files:

```
        tar -xvf standalone_gfdl-vortextracker_v3.9a.tar
        tar -xvf test_data.tar
```

Once unpacked, there will be two directories:

standalone_gfdl-vortextracker_v3.9a and *test_data*

The directory *standalone_gfdl-vortextracker_v3.9a* contains the GFDL vortex tracker routines (*gettrk, tave, and vint*), the *grbindex* and *grb2index* programs, and the libraries needed to build the GFDL vortex tracker. The directory *test_data* contains sample input data and namelist files for users to test the installed GFDL vortex tracker.

2.2 Basic System Requirements, Libraries and Tools

The source code of the GFDL vortex tracker is in the form of programs written in FORTRAN and C. The build system relies on use of the Perl scripting language and the *make* command.

The basic requirements for building and running the vortex tracker are the following:

1. FORTRAN compiler with LAPACK or MKL
2. C compiler
3. Perl
4. Make
5. NetCDF v4.2+
6. pNetCDF
7. HDF5
8. Libraries Z, PNG and Jasper

Because these tools are typically the purview of system administrators to install and maintain, they are lumped together here as part of the basic system requirements.

The GFDL vortex tracker has been tested on a variety of computing platforms. Currently it is actively supported on Linux computing platforms using both the Intel and PGI compilers. While the build system provides legacy support for IBM AIX platforms, the unavailability of AIX test platforms means all AIX support is cursory at best.
This release of the GFDL vortex tracker has been tested on Linux PGI v14.3 and Linux Intel v15.0.3. Unforeseen build issues may occur when using older compiler versions. As always, the best results come from using the most recent version of compilers.

The *standalone_gfdl-vortextracker_v3.9a/libs* directory includes the NCEP libraries needed to build the executables:

1. BACIO library
2. W3EMC and W3NCO libraries

3. G2 and G2TMPL libraries
4. SIGIO library

2.3 Building the GFDL Vortex Tracker

2.3.1 Setting things up

A number of supplemental libraries are needed to compile the vortex tracker. Six of these are NCEP libraries (w3nco, w3emc, bacio, sigio, g2 and g2tmpl) and are included with the source code. These are automatically built as part of the tracker build. The remaining three libraries (Z, png, and Jasper) are more commonly part of the system libraries. The DTC build system assumes that these libraries are installed on your system. It further assumes that they were built with the same compilers that will be used to build the vortex tracker.

If the compilation is successful, the six NCEP libraries will be installed in the directory *standalone_gfdl-vortextracker_v3.9a/libs/*. The build automatically finds and links to these. The remaining three libraries must be made available to the build system by setting the three environment variables:

1. *LIB_Z_PATH*
2. *LIB_PNG_PATH*
3. *LIB_JASPER_PATH*

For csh/tcsh, the library paths are set with the command:

```
        setenv LIB_Z_PATH /path/to/Z_LIB/
        setenv LIB_PNG_PATH /path/to/PNG_LIB/
        setenv LIB_JASPER_PATH /path/to/JASPER_LIB/
```

In bash/ksh, the library paths to be set are:

```
        export LIB_Z_PATH=/path/to/Z_LIB/
        export LIB_PNG_PATH=/path/to/PNG_LIB/
        export LIB_JASPER_PATH=/path/to/JASPER_LIB/
```

Now that the build environment has been set up, the build may commence.

2.3.2 Building the GFDL vortex tracker

To configure the vortex tracker for compilation, from within the *standalone_gfdl-vortextracker_v3.9a* directory, type:

```
        ./configure
```

The configure script first checks the system hardware, next it confirms that the environment variables for the support libraries have been set to valid paths. If the paths are not set, the build asks for paths to the Z, PNG, and JASPER libraries. It concludes by asking the user to choose a configuration supported by current machine architecture.

For Linux, the options are:

1. *Linux x86_64, PGI compiler (serial)*
2. *Linux x86_64, Intel compiler (serial)*
3. *Linux x86_64, Intel compiler super debug (serial)*
4. *Linux x86_64, PGI compiler, SGI MPT (serial)*
5. *Linux x86_64, Intel compiler, SGI MPT (serial)*
6. *Linux x86_64, Intel compiler, IBM POE (serial)*
7. *Linux x86_64, Intel compiler, CrayPE (serial)*

For the IBM, only one choice is available:

1. *AIX (serial)*

The configure script creates a file called *configure.trk*. This file contains compilation options, rules, and paths specific to the current machine architecture. The configure file can be edited to change compilation options, if desired.

In csh/tcsh, to compile the vortex tracker and save the build output to a log file, type:

```
        ./compile |& tee tracker.log
```

In bash/ksh, type:

```
        ./compile 2>&1 | tee tracker.log
```

To remove all object files, type:

```
        ./clean
```

To conduct a complete clean which removes all built files, including the executables, libraries, and the *configure.trk*, type:

```
        ./clean -a
```

A complete clean is strongly recommended if the compilation failed, or if the configuration file is changed.

If the compilation was successful, five executables, *gettrk.exe, grbindex.exe, grb2index.exe*, *tave.exe* and *vint.exe,* will be created in the *standalone_gfdl-vortextracker_v3.9a/trk_exec* directory.

3

Executables

3.1 gribindex.exe

This executable is used to generate a GRIB1 index file for input to the GFDL vortex tracker.
INPUT:
Model forecast in GRIB1 format.

USAGE:

```
            grbindex.exe GRIB1_FILE INDEX_FILE
```

OUTPUT:
GRIB1 index file.

3.2 grb2index.exe

This executable is used to generate a GRIB2 index file for input to the GFDL vortex tracker.
INPUT:
Model forecast in GRIB2 format.

USAGE:

```
            grb2index.exe GRIB2_FILE INDEX_FILE
```

OUTPUT:

GRIB2 index file.

3.3 gettrk.exe

This is the main vortex tracking program.
INPUT:
Input GRIB or NetCDF model forecast and index files (for GRIB files).

When all lead times of the post-processed model forecast are in a single GRIB file (that is, when *inp%file_se*q in the namelist is set to *onebig*), the input GRIB model forecast and index files include:

1. ***fort.11***: a single GRIB file containing the post-processed model forecast
2. ***fort.31***: a GRIB index file generated by using the program *grbindex.exe* (*grb2index.exe*) on the GRIB1(GRIB2) file fort.11

Conversely, when there are multiple input GRIB files, each with a forecast lead time (that is, when *inp%file_seq* in the namelist is set to *multi*), the input GRIB model forecast and index files include:

gmodname.rundescr.atcfdescr.CCYYMMDDHH.fmin
gmodname.rundescr.atcfdescr.CCYYMMDDHH.fmin.ix

Where *gmodname, rundescr* and *atcfdescr* are the name of the model, description of the model run and the description of the storm, respectively. These three options are specified in the namelist by *fnameinfo%gmodname*, *fnameinfo%rundescr* and *fnameinfo%atcfdescr*, respectively.

CCYYMMDDHH specifies the first two digits in the 4-digit year description, year, month, day and hour of the starting time of the forecast. The first two digits of the year, the second two digits of the year, the month, the day and the hour are specified in the namelist by *inp%bcc*, *inp%byy*, *inp%bmm*, *inp%bdd* and *inp%bhh*, respectively.

Variable *min* specifies the forecast lead time (in minutes). The forecast lead time is specified in the format of 5-digit integer.

As is in the example of one big GRIB1 file, for each GRIB1 file, there is a GRIB index file with the suffix *.ix*. The index file is generated by using the program *grbindex.exe* on the GRIB1 files.

For example, the following are the input file names where the model is *hwrf* (\$*gmodname*), the data are *25x25* degrees (\$*rundescr*), and the storm is *AL142016* (\$*atcfdescr*), the forecast starting time is 2016100400, and the forecast lead time interval is 360 minutes:

```
hwrf.25x25.AL142016.2016100400.f00000
hwrf.25x25.AL142016.2016100400.f00000.ix
hwrf.25x25.AL142016.2016100400.f00360
hwrf.25x25.AL142016.2016100400.f00360.ix
……
hwrf.25x25.AL142016.2016100400.f07560
hwrf.25x25.AL142016.2016100400.f07560.ix
```

Note when *inp%file_seq* in the namelist is set to *multi*, fort.11 and fort.31 are not needed.

3. ***tcvit_rsmc_storms.txt:*** TCVitals file containing the first guess location of the forecast vortex.

For example, the following TCVitals file (this should be a 1-line file without line break) provides a first guess location for Hurricane Irene of 20.6 N and 70.6 W

```
NHC  09L IRENE     20110823 1200 206N 0706W 295 051 0978 1008 0556 44 028 0334 0222 0222 0241
D 0167 0111 0111 0130 72 280N  780W 0083 0056 0037 0065
```

4. ***fort.14***: TCVitals file used for tropical cyclogenesis tracking. The file can be blank, and must exist in the directory where the tracker is run even when the cyclogenesis capability is not used; otherwise the tracker will stop.

5. ***fort.15:*** Forecast lead times (in minutes) the tracker will process.

For example, the following file specifies that the tracker will process the GRIB files for lead times 0, 180, 360 and 540 minutes.

```
   1    0
   2  180
   3  360
   4  540
```

Note the format of the records in this file is a 4-digit integer showing the number of the forecast lead time, followed by 1 blank space, followed by a 4-digit integer showing the forecast lead time in minutes.

6. **input.nml**: Namelist file with fields described in Appendix A. You can change the name of this file as it goes as input to *gettrk.exe*.

USAGE:
When the input files and the namelist exist in the work directory, copy or link the executable *gettrk.exe* to this work directory and type

```
        gettrk.exe < input.nml
```

OUTPUT:
When tracking existing storms, three files are output, all in a modified ATCF format: *fort.64, fort.68, and fort.69*. When the tracker runs in cyclogenesis mode, it produces another ATCF format file: *fort.66*. And if the *ikeflag* is set to *y* in the namelist, still another output file will be created: *fort.74.* Most of these files are described in Appendix B, and description of the contents of file *fort.68* can be found in subroutine *output_atcf_sink* in file *standalone_gfdl-vortextracker_v3.9a/trk_src/gettrk/gettrk_main.f.*

3.4 vint.exe

This is a program to interpolate from various pressure levels onto a regularly spaced grid, with 50-hpa vertical level intervals. Each run only processes one lead time. Therefore, it is necessary to use this executable separately for each lead time.

INPUT:
fort.11: GRIB file containing the post-processed model output containing at least two levels of geopotential height/temperature data: 300 and 500 hpa.
fort.16: text file containing the number of input pressure levels in i4,1x, i4 format similar to fort.15 (shown above).
fort.31: index file of *fort.11*
namelist: text file containing the following (Edit *input.nml* and follow the format)

&timein	
ifcsthour	Forecast lead time
iparm	Variable to be processed; Input GRIB parm to search*
gribver	For GRIB data, 1=GRIB1, 2=GRIB2
g2_jpdtn	For GRIB2 data, 0=analysis or forecast, 1=ensemble forecast

For phase space diagnostics, geopotential height (when *phasescheme=cps*, *iparm*=7) or temperature (when *phasescheme=vtt*, *iparm*=11) or both (when *phasescheme=both*) need to be processed.

USAGE:

```
        vint.exe< namelist
```

OUTPUT:
fort.51: GRIB file containing the temperature or geopotential height data on vertical levels 300, 350, 400, 450 and 500 hpa.

3.5 tave.exe

This is a program to vertically average temperature in the 500-300 hpa layer.

INPUT:

fort.11: GRIB file containing the temperature at least at levels 300, 350, 400, 450 and 500 hpa. This file can be generated by *vint.exe*
fort.16: text file containing the number of input pressure levels in i4,1x, i4 format similar to fort.15 (shown above).
fort.31: index file of *fort.11.*
namelist: text file containing the following (Edit *input.nml* and follow the format)

&timein	
ifcsthour	Forecast lead time
iparm	=11
gribver	For GRIB data, 1=GRIB1, 2=GRIB2
g2_jpdtn	For GRIB2 data, 0=analysis or forecast, 1=ensemble forecast

USAGE:

```
        tave.exe < namelist
```

OUTPUT:
fort.51: GRIB file containing the mean temperature in the 300-500 hpa layer.

4

Additional Capabilities

4.1 How to Generate Phase Space Diagnostics

To use this function, the user should run the following procedures.

1. In the GFDL vortex tracker namelist (see NAMELIST above for detailed info) set the following items:

   phaseflag=y,
   phasescheme=both or cp’ or vtt
   wcore_depth=1.0

2. If *phasescheme* is set to *cps*, run *vint.exe* (see 3.4) to vertically interpolate the geopotential from 300 to 500 hPa at 50 hPa interval. Then append these geopotential variables to the tracker’s GRIB format input file.
3. If *phasescheme* is set to *vtt* , run *vint.exe* (see 3.4) to vertically interpolate the temperature from 300 to 500 hPa at a 50 hPa interval. Then run *tave.exe* (see 3.4) to obtain the average temperature between 300 and 500 hPa. This average temperature field is appended to the tracker’s GRIB format input file.
4. If *phasescheme* is set to *both*, then both steps 2) and 3) are needed.
5. When the phase space diagnostics is performed, the output will be generated in *fort.64* as fields 37-41 (see Appendix B).

4.2 How to Run the Tracker in Cyclogenesis Mode

To use this function, the user should run the following procedures.

1. In the GFDL vortex tracker namelist set the following items:

trkrinfo%westbd
trkrinfo%eastbd
trkrinfo%southbd
trkrinfo%northbd

They are the boundaries for searching for new storms in cyclogenesis mode. They do not need to match the boundaries of the forecast grid.

2. In the GFDL vortex tracker namelist, set the item *trkrinfo%type=tcgen* or *trkinfo%type=midlat* (for the difference between *tcgen* and *midlat*, see 3.1).
3. In addition to *fort.64* and *fort.69*, another ATCF format output file, *fort.66* (see Appendix B), will be produced by the tracker when it runs in cyclogenesis mode.

5

Running Test Case

5.1 Running tracker with multiple GRIB1 files

Sample input data and namelists are provided. Refer to Section 2.1 for downloading and uncompressing the test case data file. The sample input data are the 126 hr HWRF forecast of Hurricane Matthew (14L) in 2016, starting from 2016 October 14th 00 UTC. We recommend that the users test the installed GFDL vortex tracker program with the included sample input data and namelist before they work on their own model output.

The directory of the sample input data, *test_case,* contains two sub-directories: *onebig* and *multi*. *Onebig* contains the sample input data that is provided as one single GRIB1 file (when *inp%file_seq* in the namelist is set to *onebig*), while *multi* contains the same input data but provided as multiple GRIB1 files each of which contains the forecast for one forecast lead time (when *inp%file_seq* in the namelist is set to *multi*).

To test the installed GFDL vortex tracker using the multiple GRIB1 input data files provided, enter the directory *test_case/multi*, and check the following files:

hwrf.25x25.AL142016.2016100400.f[00000-7560]: HWRF files from 00 h to 126 hr
fort.15: link to *fcst_minutes*.
tcvit_rsmc_storms.txt : link to TCVitals file containing the first guess location of the forecast vortex.
input.nml: namelist for tracker.exe.

Creating index files

Now link or copy the executable *grbindex.exe* to *test_case/multi* directory and type

```
        grbindex.exe hwrf.25x25.AL142016.2016100400.f00000 \
              hwrf.25x25.AL142016.2016100400.f00000.ix
```

Repeat this for all the forecast lead times.

Now link or copy the executable *gettrk.exe* to *test_case/multi* directory and type

```
        gettrk.exe < input.nml
```

If the program runs successfully, the output files *fort.64* and *fort.69* should be generated.

5.2 Running tracker with a single GRIB1 file

To test the installed GFDL vortex tracker using the multiple GRIB1 input data files provided, enter the directory *test_case/onebig*, and check the following files:

hwrf_onebig.25x25.AL142016.2016100400: HWRF file containing forecasts from 00 h to 126 hr
fort.15: link to *fcst_minutes*.
tcvit_rsmc_storms.txt : link to TCVitals file containing the first guess location of the forecast vortex.
input.nml: namelist for tracker.exe.

Creating index files

Now link or copy the executable *grbindex.exe* to *test_case/multi* directory and type

```
        grbindex.exe hwrf_onebig.25x25.AL142016.2016100400 \
              hwrf_onebig.25x25.AL142016.2016100400.ix
```

Now link *hwrf_onebig.25x25.AL142016.2016100400* to fort. 11 and *hwrf_onebig.25x25.AL142016.2016100400.ix* to fort.31.

Now link or copy the executable *gettrk.exe* to *test_case/onebig* directory and type

```
          gettrk.exe < input.nml
```

If the program runs successfully, the output files *fort.64* and *fort.69* should be generated.

Note that the files *fort.64* and *fort.69* generated in *onebig* and *multi* directories should be identical.

6

Plotting Tracker Output

6.1 Plotting using GrADS

The standalone GFDL vortex tracker release includes *atcf_plot*, which is a set of GrADS scripts that can be used to plot hurricane track files in ATCF format. The *atcf_plot* package can be found in directory *standalone_gfdl-vortextracker_v3.9a/trk_plot/plottrak.*

To use *atcf_plot* to plot the storm's track:

- Enter the directory *standalone_gfdl-vortextracker_v3.9a/trk_plot/ plottrak.*
- In order to plot the output of the vortex tracker, it is necessary to add the output to the operational A-deck file (e.g. aal142016.dat provided with the release in the *gfdl-vortextracker_v3.9a/trk_plot/plottrak* directory) so it can be visualized with the *atcf_plot* tool.
- Copy the track output *fort.64* to the current directory.
- The default *atcfname* assigned to the provided HWRF output in *test_data/multi* is HREL. The user should check the ATCF name (in the *input.nml &atcfinfo%atcfname*), if it matches with any model in the A-deck file.
- If necessary, change the *atcfname* in *fort.64* file.
- Append the newly created track data to the operational A-deck file.
- Verify that the updated A-deck file contains the *atcfname* forecast.

Run Tracker Visualization

ATCF_PLOT is a GUI based visualization tool that renders storm tracks using GrADS. *atcf_plot* allows the user to overlay multiple storm tracks along with observed storm locations on a single plot, as well as time series of storm intensity.

- There are some predefined ATCF names in the *atcfplot.gs* GrADS script. Check whether the user defined ATCF name is already present in the script. If not, replace a currently existing ATCF name with the user defined name. The user needs to edit at 3 instances in the script (around lines 1626, 1828, and 2938). You can also search the name you are switching and replace it with your ATCF name.
- Start up *atcf_plot*
  - Run the script *./atcfplot.sh 2016 al*
    The script takes as input the ***storm year*** (2016, in this exercise) and ***basin*** (al, in this exercise). A GUI will be spawned, with buttons on the left, top, and bottom. Rearrange the windows so you can see both the GUI and terminal windows. The GUI writes status messages to that terminal window.
- From the top drop down **Storms** menu, select **al142016 Matthew**.
- From the top drop down **Dates** menu, select the date **20161004** and then hour **00**.
- Select models from the left column of buttons, e.g. the **HREL** track we just generated and the operational **HWRF** track.
- Press the **Plot** button to generate the plot.
- Save the plot to an image file by selecting the top drop down **Main** menu, then any of the **Print** sub-menus.
- From the top menu **Opts_1** dropdown list, select **Intensity forecast** to plot a time series. The available options are:
  MSLP
  Winds
- Press the **Plot** button to generate the time series intensity plot.
- Explore the other menus, pressing **Plot** whenever you are ready to create a new image.
- Press **Quit** to shut down the GUI.
- Once you are done creating plots with ATCF_PLOT, the image files you created will be stored in *standalone_gfdl-vortextracker_v3.9a/ trk_plot/plottrak/*t*racks* directory and may be viewed with the **display** command (for example ***display mslp1.png***). **Note: display** is a utility in the **ImageMagick** program

Note: From Grads webpage: N.B. The Cairo graphics display interface in version 2.1+ does not support widgets).

A

Namelist

Appendix A

Namelist for GRIB files

&datein	
inp%bcc	First 2 digits of the year for the initial time of the forecast (e.g., the *20* in *2011*)
inp%byy	Last 2 digits of the year for the initial time of the forecast (e.g., the *11* in *2011*)
inp%bmm	2-digit month (01, 02, etc) for the initial time of the forecast
inp%bdd	2-digit day for the initial time of the forecast
inp%bhh	2-digit hour for the initial time of the forecast
inp%model	Model ID number as defined by the user. This is used in subroutine getdata to define what the GRIB IDs are for surface wind levels. Create a unique number for your model and make sure you have the corresponding IDs set up for it in subroutine getdata. For HWRF use 17. The Model ID numbers for other models are listed below: (1) GFS, (2) MRF, (3) UKMET, (4) ECMWF, (5) NGM, (6) NAM, (7) NAVGEM, (8) GDAS, (10) NCEP Ensemble, (11) ECMWF Ensemble, (13) SREF Ensemble, (14) NCEP Ensemble (from ensstat mean fields), (15) CMC, (16) CMC Ensemble, (17) HWRF, (18) HWRF Ensemble, (19) HWRF-DAS (HDAS), (20) Ensemble RELOCATION, (21) UKMET hi-res (NHC), (23) FNMOC Ensemble
inp%lt_units	*hours* or *minutes*, this defines the lead time units used by the PDS in your GRIB header

inp%file_seq	*onebig* or *multi*, this specifies if the tracker will process one big input file *or* multiple files for each individual lead times.
inp%modtyp	Type of the model. Either *global* or *regional*.
inp%nesttyp	Type of the nest grid. Either *moveable* or *fixed*.
&atcfinfo	
atcfnum	Obsolete; can be set to any integer
atcfname	Character model ID that will appear in the ATCF output (e.g., GFSO, HWRF, NAHW, HCOM etc)
atcfymdh	10-digit yyyymmddhh date that will be used in output text track files
atcffreq	Frequency (hours*100) of output for atcfunix. Default value is 600 (six hourly).
&trackerinfo	
trkrinfo%westbd	For genesis runs, the western boundary for searching for new storms. Does not need to match the boundaries of your grid, it can be smaller than your grid.
trkrinfo%eastbd	For genesis runs, the eastern boundary for searching for new storms. Does not need to match the boundaries of your grid, it can be smaller than your grid.
trkrinfo%northbd	For genesis runs, the northern boundary for searching for new storms. Does not need to match the boundaries of your grid, it can be smaller than your grid.
trkrinfo%southbd	For genesis runs, the southern boundary for searching for new storms. Does not need to match the boundaries of your grid, it can be smaller than your grid.
fnameinfo%gmodname	Defines the model name in the input files, e.g., *hwrf*. Only when *inp%file_seq=multi*
fnameinfo%rundescr	Describe the model runs in the input files, e.g., *coupled*. Only when *inp%file_seq=multi*
fnameinfo%atcfdescr	Describe the storm information in the input files, e.g., *irene09l*. Only when *inp%file_seq=multi*
trkrinfo%type	trkrinfo%type defines the type of tracking to do. A *tracker* run functions as the standard TC tracker and tracks only existing storms from the TCVitals. Options *tcgen* and *midlat* define a genesis run in which the tracker will look for new storms in addition to tracking existing ones. Option *tcgen* means all parameters at the various vertical levels will be used, while

	midlat means only MSLP will be used and no checks will be performed to differentiate tropical from non-tropical cyclones.
trkrinfo%mslpthresh	Threshold for the minimum MSLP gradient (units hPa km^{-1}) that must be met in order to continue tracking. Default: 0.0015
trkrinfo%use_backup_mslp_grad_check	Flag to check whether MSLP was read in. Best to leave this check as “y”.
trkrinfo%v850thresh	Threshold for the minimum azimuthally-average 850 hPa cyclonic tangential wind speed (ms^{-1}) that must be exceeded in order to keep tracking. Default: 1.5000
trkrinfo%use_backup_850_vt_check	Flag to check whether the 850 mb u- and v- wind component were read in. Best to leave this check as “y”.
trkrinfo%gridtype	*global* or *regional*, this defines the type of domain grid. For limited area models, choose *regional*.
trkrinfo%enable_timing	Flag to do timing calls and print out results. 0 will turn off timing calls, any other number will print it.
trkrinfo%contint	This specifies the interval (in Pa) used by subroutine *check_closed_contour* to check for a closed contour in the MSLP field when running in genesis mode. Note that *check_closed_contour* is also called from the routine that checks for a warm core, but the contour interval is hard-wired in the executable as 1.0 degree K for that usage.
trkrinfo%want_oci	Diagnose ROCI and include it in ATCF output
trkrinfo%out_vit	This is only set to *y* if the tracker is running in genesis mode, and it tells the tracker to write out a TCVitals record for any storms that it finds at the initial time in a forecast. Print out “gen_vitals” for 00h storms.
trkrinfo%use_land_mask	For genesis runs, checks land-sea mask.
trkrinfo%inp_data_type	Data type: ‘grib’ or ‘netcdf’
trkrinfo%gribver	For GRIB data, 1=GRIB1, 2=GRIB2
trkrinfo%g2_jpdtn	For GRIB2 data, 0=analysis or forecast, 1=ensemble forecast
trkrinfo%g2_mslp_parm_id	GRIB2 MSLP ID: 192 = membrane, 1= standard Recommended to use: 192.

trkrinfo%g1_mslp_parm_id	GRIB1 MSLP ID: 130 = membrane, 102 = standard Recommended to use: 130.
trkrinfo%g1_sfcwind_level_typ	GRIB1 near-sfc winds: Most models = 105
trkrinfo%g1_sfcwind_val	GRIB1 near-sfc winds: Most models = 10
&phaseinfo	
phaseflag	*y* or *n*, tells the program whether or not to determine the cyclone thermodynamic phase.
phasescheme	*cps*, *vtt*, *both*, tells the program which scheme to use for checking the cyclone phase. Option *cps* is Hart's cyclone phase space, *vtt* is a simple 300-500 hPa warm core check based on Vitart, and *both* tells the program to use both schemes. Not used if phaseflag=*n*.
wcore_depth	The contour interval (in deg K) used in determining if a closed contour exists in the 300-500 hPa temperature data, for use with the *vtt* scheme.
&structinfo	
structflag	*y* or *n*, tells the program whether or not to compute additional diagnostics of the cyclone wind structure.
ikeflag	*y* or *n*, tells the program whether or not to calculate the Integrated. Kinetic Energy (IKE) and Storm Surge Damage Potential (SDP).
&waitinfo	
use_waitfor	*y* or *n*, for waiting for input files. Use *n* for inp%file_seq = "onebig.
wait_min_age	For realtime HWRF runs
wait_min_size	For realtime HWRF runs
wait_max_wait	For realtime HWRF runs
wait_sleeptime	For realtime HWRF runs
use_per_fcst_command	For realtime HWRF runs
per_fcst_command	For realtime HWRF runs
&parmpreflist	
user_wants_to_track_zeta850	850 hPa Vorticity; Flag that dictate variables used for tracking. ‘y’ or ‘n’

user_wants_to_track_zeta700	700 hPa Vorticity; Flag that dictate variables used for tracking. 'y' or 'n'
user_wants_to_track_wcirc850	850 hPa Wind circulation; Flag that dictate variables used for tracking. 'y' or 'n'
user_wants_to_track_wcirc700	700 hPa Wind circulation; Flag that dictate variables used for tracking. 'y' or 'n'
user_wants_to_track_gph850	850 hPa Height; Flag that dictate variables used for tracking. 'y' or 'n'
user_wants_to_track_gph700	700 hPa Height; Flag that dictate variables used for tracking. 'y' or 'n'
user_wants_to_track_mslp	MSLP; Flag that dictate variables used for tracking. 'y' or 'n'
user_wants_to_track_wcircsfc	Surface Wind circulation; Flag that dictate variables used for tracking. 'y' or 'n'
user_wants_to_track_zetasfc	Surface Height; Flag that dictate variables used for tracking. 'y' or 'n'
user_wants_to_track_thick500850	500-850 hPa thickness; Flag that dictate variables used for tracking. 'y' or 'n'
user_wants_to_track_thick200500	200-500 hPa thickness; Flag that dictate variables used for tracking. 'y' or 'n'
user_wants_to_track_thick200850	200-850 hPa thickness; Flag that dictate variables used for tracking. 'y' or 'n'
&verbose	
verb	Level of detail printed to terminal. Choose from 0 (no output), 1 (error messages only), 2 (more messages), 3 (all messages).
verb_g2	=0 suppresses GRIB2 I/O messages.

Namelist for NetCDF files

If the input data is in NetCDF format, additional namelist options are needed to define the variable names in the input data. Compared to GRIB files, the namelist for NetCDF files requires more comprehensive listing of variables at different levels. Write 'X' if the variable is not included in the input dataset. Remember that the NetCDF data should be in a regular lat-lon to be read by the tracker.

&netcdflist

netcdfinfo%num_netcdf_vars	999 Don’t change this.
netcdfinfo%netcdf_filename	Path to NetCDF file.
netcdfinfo%lat_name	Name of variable containing latitude information.
netcdfinfo%lmaskname	Name of variable containing land mask information.
netcdfinfo%lon_name	Name of variable containing longitude information.
netcdfinfo%mslpname	Name of variable containing MSLP information.
netcdfinfo%rv850name	Name of variable containing radial velocity at 850 hPa information.
netcdfinfo%rv700name	Name of variable containing radial velocity at 700 hPa. information.
netcdfinfo%time_name	Name of variable containing time information.
netcdfinfo%time_units	Time units: hours, days.
netcdfinfo%tmean_300_500_name	Name of variable containing mean temperature in the 300-500 hPa layer information. This allows to use the “vtt” phase checking (simple check for a warm core in the 300-500 hPa layer.
netcdfinfo%u500name	Name of variable containing 500 hPa u wind information.
netcdfinfo%u700name	Name of variable containing 700 hPa u wind information.
netcdfinfo%u850name	Name of variable containing 850 hPa u wind information.
netcdfinfo%usfcname	Name of variable containing surface u wind information.
netcdfinfo%v500name	Name of variable containing 500 hPa v wind information.
netcdfinfo%v700name	Name of variable containing 700 hPa v wind information.
netcdfinfo%v850name	Name of variable containing 850 hPa v wind information.
netcdfinfo%vsfcname	Name of variable containing surface v wind information.
netcdfinfo%z200name	Name of variable containing 200 hPa height information.
netcdfinfo%z300name	Name of variable containing 300 hPa height information.
netcdfinfo%z350name	Name of variable containing 350 hPa height information.
netcdfinfo%z400name	Name of variable containing 400 hPa height information.
netcdfinfo%z450name	Name of variable containing 450 hPa height information.
netcdfinfo%z500name	Name of variable containing 500 hPa height information.
netcdfinfo%z550name	Name of variable containing 550 hPa height information.
netcdfinfo%z650name	Name of variable containing 650 hPa height information.

netcdfinfo%z700name	Name of variable containing 700 hPa height information.
netcdfinfo%z750name	Name of variable containing 750 hPa height information.
netcdfinfo%z800name	Name of variable containing 800 hPa height information.
netcdfinfo%z850name	Name of variable containing 850 hPa height information.
netcdfinfo%z900name	Name of variable containing 900 hPa height information.

B

Output

Appendix B

Output formats

fort.69

A sample of the vortex tracker output *fort.69* is below:

```
AL, 09, 2011082312, 03, HCOM, 00000, 204N,  706W,  87,  978, XX,  34, NEQ, 0103, 0077, 0058, 0095,    0,    0,  24
AL, 09, 2011082312, 03, HCOM, 00000, 204N,  706W,  87,  978, XX,  50, NEQ, 0058, 0040, 0031, 0055,    0,    0,  24
AL, 09, 2011082312, 03, HCOM, 00000, 204N,  706W,  87,  978, XX,  64, NEQ, 0043, 0025, 0016, 0042,    0,    0,  24
AL, 09, 2011082312, 03, HCOM, 00600, 208N,  714W,  95,  964, XX,  34, NEQ, 0155, 0100, 0058, 0145,    0,    0,  21
AL, 09, 2011082312, 03, HCOM, 00600, 208N,  714W,  95,  964, XX,  50, NEQ, 0066, 0057, 0037, 0060,    0,    0,  21
AL, 09, 2011082312, 03, HCOM, 00600, 208N,  714W,  95,  964, XX,  64, NEQ, 0046, 0033, 0028, 0042,    0,    0,  21
AL, 09, 2011082312, 03, HCOM, 01200, 208N,  722W,  94,  963, XX,  34, NEQ, 0123, 0096, 0060, 0109,    0,    0,  23
AL, 09, 2011082312, 03, HCOM, 01200, 208N,  722W,  94,  963, XX,  50, NEQ, 0069, 0048, 0049, 0062,    0,    0,  23
```

Column 1: basin name. *AL* represents Atlantic and *EP* northeast Pacific.

Column 2: ATCF storm ID number. Irene was the 9th storm in the Atlantic Basin in 2011.

Column 3: model starting time.

Column 4: constant and 03 simply indicates that this record contains model forecast data.

Column 5: model ATCF name.

Column 6: forecast lead time in hours multiplied by 100 (e.g, 00900 represents 9 .00 h).

Column 7-8: vortex center position (latitude and longitude multiplied by 10).

Column 9: vortex maximum 10-m wind (in kt).

Column 10: vortex minimum MSLP (in hpa).

Column 11: placeholder for character strings that indicate whether the storm is a depression, tropical storm, hurricane, subtropical storm etc. Currently, that storm type character string is only used for the observed storm data in the NHC Best Track data set.

Column 12: thresholds wind speed in knots, an identifier that indicates whether this record contains radii for the 34-, 50- or 64-knot wind thresholds.

Column 13: *NEQ* indicates that the four radii values that follow will begin in the northeast quadrant and progress clockwise.

Column 14-17: wind radii (in nm) for the threshold winds in each quadrant.

Column 18-19: not used.

Column 20: radius of maximum winds, in nautical miles.

fort.64

A sample of the vortex tracker output *fort.64* is listed below:

```
AL, 09, 2011082312, 03, HCOM, 000, 204N,  706W,  87,  978, XX,  34, NEQ, 0103, 0077, 0058, 0095,    0,
0,  24,   0,   0,    ,   0,    ,   0,   0,          , ,  ,    ,  0,   0,   0,   0,      THERMO PARAMS,   -9999,   -9999, -
9999, U, 10, DT, -999
AL, 09, 2011082312, 03, HCOM, 000, 204N,  706W,  87,  978, XX,  50, NEQ, 0058, 0040, 0031, 0055,    0,
0,  24,   0,   0,    ,   0,    ,   0,   0,          , ,  ,    ,  0,   0,   0,   0,      THERMO PARAMS,   -9999,   -9999, -
9999, U, 10, DT, -999
AL, 09, 2011082312, 03, HCOM, 000, 204N,  706W,  87,  978, XX,  64, NEQ, 0043, 0025, 0016, 0042,    0,
0,  24,   0,   0,    ,   0,    ,   0,   0,          , ,  ,    ,  0,   0,   0,   0,      THERMO PARAMS,   -9999,   -9999, -
9999, U, 10, DT, -999
AL, 09, 2011082312, 03, HCOM, 006, 208N,  714W,  95,  964, XX,  34, NEQ, 0155, 0100, 0058, 0145,    0,
0,  21,   0 ,   0,    ,   0,    ,   0,  0,          , ,  ,    ,  0,   0,   0,   0,      THERMO PARAMS,   -9999,   -9999, -
9999, U, 10, DT, -999
AL, 09, 2011082312, 03, HCOM, 006, 208N,  714W,  95,  964, XX,  50, NEQ, 0066, 0057, 0037, 0060,    0,
0,  21,   0 ,   0,    ,   0,    ,   0,  0,          , ,  ,    ,  0,   0,   0,   0,      THERMO PARAMS,   -9999,   -9999, -
9999, U, 10, DT, -999
AL, 09, 2011082312, 03, HCOM, 006, 208N,  714W,  95,  964, XX,  64, NEQ, 0046, 0033, 0028, 0042,    0,
0,  21,   0 ,   0,    ,   0,    ,   0,   0,          , ,  ,    ,  0,   0,   0,   0,      THERMO PARAMS,   -9999,   -9999, -
9999, U, 10, DT, -999
AL, 09, 2011082312, 03, HCOM, 012, 208N,  722W,  94,  963, XX,  34, NEQ, 0123, 0096, 0060, 0109,    0,
0,  23,   0 ,   0,    ,   0,    ,   0,   0,          , ,  ,    ,  0,   0,   0,   0,      THERMO PARAMS,   -9999,   -9999, -
9999, U, 10, DT, -999
AL, 09, 2011082312, 03, HCOM, 012, 208N,  722W,  94,  963, XX,  50, NEQ, 0069, 0048, 0049, 0062,    0,
0,  23,   0 ,   0,    ,   0,    ,   0,   0,          , ,  ,    ,  0,   0,   0,   0,      THERMO PARAMS,   -9999,   -9999, -
9999, U, 10, DT, -999
AL, 09, 2011082312, 03, HCOM, 012, 208N,  722W,  94,  963, XX,  64, NEQ, 0045, 0032, 0036, 0046,    0,
0,  23,   0 ,   0,    ,   0,    ,   0,   0,          , ,  ,    ,  0,   0,   0,   0,      THERMO PARAMS,   -9999,   -9999, -
9999, U, 10, DT, -999
```

Column 1-20: same as *fort.69* except that column 6, the forecast lead time, instead of being a 5-digit integer as in *fort.69*, is a 3-digit integer.

Column 21-35: space fillers

Column 36: *THERMO PARAMS*, indicating that thermodynamics parameters will follow.

Column 37-39: The three cyclone phase space parameters, and all values shown have been multiplied by a factor of 10. The values are listed in the following order: (1) Parameter B (left-right thickness asymmetry);
(2) Thermal wind (warm/cold core) value for lower troposphere (900-600 hPa); and (3) Thermal wind value for upper troposphere (600-300 hPa).

Column 40: Presence of a warm core. In this sample it is *U*, which stands for *undetermined*, meaning the warm core check was not performed. When the warm core check is performed, this field will be either *Y* or *N*, indicating whether the warm core is identified or not.

Column 41: warm core strength x 10 (in degrees). It indicates the value of the contour interval that was used in performing the check for the warm core in the 300-500 hPa layer.

Column 42-43: constant strings.

fort.66

A sample of the vortex tracker output *fort.66* is listed below:

```
A   B    C                                 D           E  F
ML  0005 2015022700_F000_409N_0604W_FOF    2015022700  03 GFSO

G   H     I    J  K   L  M  N    R1   R2   R3   R4   S    T
012 450N  488W 42 998 XX 34 NEQ  0182 0276 0259 0000 1010 237

U   V     W     X     Y Z  AA  BB  CC  DD  EE
155 1185  -5339 -3337 N 71 272 237 301 172 205
```

A - From a standard ATCF file for tropical cyclones, this is where the basin ID would be (AL, EP, WP, etc). Here it is identified as either "TG" for a "tcgen" run, where all available parameters up through 700 mb is used to search for a center, or "ML" for a "mid-latititude" genesis run, where only MSLP is used to search.

B - The sequential disturbance number for this run of the tracker. This number is really only useful for diagnosing issues. Don't use this to try to compare storms between runs of different models or to compare against a run of the same model at the next cycle time.

C - Unique, 30-character storm ID containing the initial time of the run in which the system was found, the lead time at which the storm was first found, the lat and lon at which it was first found. "FOF" stands for "Found On the Fly", as opposed to "TCV" meaning that the storm was identified on tcvitals.

D - Initial time of the current forecast

E - Leave this as "03"
F - Model ID.
G - Lead time for this record
H - Latitude * 10
I - Longitude * 10
J - Max near-surface (usually 10 m) wind speed, in knots.
K - Minimum MSLP in mb
L - Level of development (e.g., "TS", "TD", "HU", "TY", "LO", "DB", etc.). Generally only used in the best track data. Forecast models don't usually report this, and so are left with what is essentially a missing value of "XX".

M,N - "34, NEQ" means that positions R1 - R4 contain the radii (nm) of 34-knot winds in each quadrant, with the "NEQ" indicating that the first one listed (R1) contains the value for the northeastern quadrant and then going clockwise from there. If winds exceeding 50 and 64 knots exist at this lead time in the model, then there will be additional records with "50, NEQ" and "64, NEQ" where the information contained in the records is identical to that contained in the "34, NEQ" record with the exception of the radii data.

S - Pressure of last closed isobar

T - Radius (n mi) of last closed isobar

U - Radius (n mi) of max wind

V - Cyclone Phase Space "Parameter B" for thermal asymmetry. (Values are *10)

W - Cyclone Phase Space lower level (600-900 mb) thermal wind parameter, for diagnosing low-level warm core. (Values are *10)

X - Cyclone Phase Space upper level (300-600 mb) thermal wind parameter, for diagnosing upper-level warm core. (Values are *10)

Y - "Y" or "N" for warm core diagnostic in 300-500 mb layer. Value of "Y" is returned if a closed contour in the temperature field exists surrounding the 300-500 mb warm core, using a 1-deg K contour Interval.

Z - Direction of storm movement, 0-360 degrees

AA - Speed of storm translation, in m/s * 10

BB - Gaussian-smoothed value of 850 mb relative vorticity (* 1e6) computed near storm center using Barnes analysis

CC - Max gridpoint value of 850 mb relative vorticity (* 1e6) found near the storm center

DD - Gaussian-smoothed value of 700 mb relative vorticity (* 1e6) computed near storm center using Barnes analysis

EE - Max gridpoint value of 700 mb relative vorticity (* 1e6) found near the storm center

fort.67

The "gen_vitals" file. If you specify that flag in your tracker input namelist of output_vitals=y, then if the tracker is being run in genesis mode (type=tcgen), and if the tracker finds a storm at tau=00h that is not already specified in a TCvitals message, then the tracker creates a "genesis tc vitals" for these storms and outputs that file. So then, if you are cycling the tracker, the next run of the tracker can read in this genesis vitals record and the tracker will now start off looking for a storm where the previous cycle's tracker run had already found one, and it will keep the unique name for that storm that was originally found at tau=00h from the previous cycle's run. This helps with continuity and "bookkeeping" if cycling these genesis runs.

fort.72

Radial wind structure. The values printed here represent near-surface V, Vt and Vr values extending radially outward in each quadrant. Values are printed both for earth-relative and storm motion-relative quadrants.

fort.73

Fractional wind coverage. The values printed here represent fractional areal coverage (within a specified radius) of near-surface winds for thresholds of 34, 50 and 64 knots.

fort.74

A sample of the vortex tracker output *fort.74* is listed below:

```
AL, 09, 2011082312, 03, HCOM, 000, 204N,  706W,  87,  978, XX,  91, IKE,   0,  23,  34,  16,   5,  0,   0,   0, 2039N,  7062W
AL, 09, 2011082312, 03, HCOM, 006, 208N,  714W,  94,  965, XX,  91, IKE,   0,  28,  42,  25,   8,  0,   0,   0, 2081N,  7142W
AL, 09, 2011082312, 03, HCOM, 012, 209N,  722W,  93,  964, XX,  91, IKE,   0,  28,  44,  25,   8,  0,   0,   0, 2088N,  7220W
AL, 09, 2011082312, 03, HCOM, 018, 213N,  728W,  99,  962, XX,  91, IKE,   0,  25,  46,  19,   9,  0,   0,   0, 2131N,  7276W
AL, 09, 2011082312, 03, HCOM, 024, 218N,  733W,  92,  962, XX,  91, IKE,   0,  27,  50,  23,   8,  0,   0,   0, 2179N,  7333W
AL, 09, 2011082312, 03, HCOM, 030, 225N,  741W,  97,  959, XX,  91, IKE,   0,  28,  51,  26,   9,  0,   0,   0, 2245N,  7415W
AL, 09, 2011082312, 03, HCOM, 036, 231N,  749W,  95,  961, XX,  91, IKE,   0,  29,  51,  27,  11,   0,   0,   0, 2314N,  7488W
AL, 09, 2011082312, 03, HCOM, 042, 239N,  756W, 100,  956, XX,  91, IKE,   0,  29,  54,  28,  11,   0,   0,   0, 2387N,  7562W
AL, 09, 2011082312, 03, HCOM, 048, 248N,  762W, 107,  953, XX,  91, IKE,   0,  30,  58,  30,  14,   0,   0,   0, 2479N,  7621W
AL, 09, 2011082312, 03, HCOM, 054, 258N,  767W, 111,  949, XX,  91, IKE,   0,  32,  62,  34,  16,   0,   0,   0, 2575N,  7668W
```

Column 1-11: Same as *fort.64*
Column 12-13: fixed fields.
Column 14: wind damage potential (not computed in this version, therefore is always zero).
Column 15: storm surge damage potential (SDP) (multiplied by 10).
Column 16-18: IKE, in terajoule, for 10 ms^{-1}, 18 ms^{-1} and 33 ms^{-1} winds, respectively.
Column 19-21: IKE for 25-40 ms^{-1}, 41-54 ms^{-1} and 55 ms^{-1} winds, currently not computed, therefore are always zero.
Column 22-23: vortex center position (latitude and longitude multiplied by 100).

fort.76

PDF: Values printed here represent PDF values of near-surface wind magnitude in each quadrant.