

# METplus Wrappers Version 2.2

## Automation for the Model Evaluation Tools

Developmental Testbed Center  
Boulder, Colorado

*Daniel Adriaansen<sup>1</sup>, Minna Win-Gildenmeister<sup>1,4</sup>, James Frimel<sup>2,4</sup>,  
Julie Prestopnik<sup>1,4</sup>, Mallory Row<sup>3</sup>, John Halley Gotway<sup>1,4</sup>,  
George McCabe<sup>1,4</sup>, Tara Jensen<sup>1,4</sup>, Jonathan Vigh<sup>1,4</sup>,  
Christina Kalb<sup>1</sup>, and Hank Fisher<sup>1</sup>*

*<sup>1</sup>National Center for Atmospheric Research,  
Research Applications Laboratory*

*<sup>2</sup>Cooperative Institute for Research in the Atmosphere at  
National Oceanic and Atmospheric Administration (NOAA)  
Earth System Research Laboratory*

*<sup>3</sup>I.M. Systems Group at  
NOAA Environmental Modeling Center*

*<sup>4</sup>Developmental Testbed Center*

July 2019

# Contents

<b>1 Overview</b>	<b>14</b>
1.1 Purpose and organization of the User's Guide	14
1.2 The Developmental Testbed Center (DTC)	14
1.3 METplus Wrappers goals and design philosophy	15
1.4 METplus Wrappers Components	15
1.5 Future development plans	15
1.6 Code support	16
<b>2 Software Installation/Getting Started</b>	<b>17</b>
2.1 Introduction	17
2.2 Supported architectures	17
2.3 Programming/scripting languages	17
2.4 Pre-requisites	17
2.5 Getting the METplus Wrappers source code	18
2.5.1 Get the source code via your Web Browser	18
2.5.2 Get the source code via Command line	21
2.6 METplus Wrappers directory structure	21
2.7 Set up your environment	22
2.8 Set up METplus Wrappers Configuration files	23
2.9 Running METplus Wrappers	24

<i>CONTENTS</i>	2
<b>3 Python Wrappers</b>	<b>27</b>
3.1 CyclonePlotter . . . . .	27
3.1.1 Description . . . . .	27
3.1.2 Configuration . . . . .	27
3.2 EnsembleStat . . . . .	28
3.2.1 Description . . . . .	28
3.2.2 Configuration . . . . .	28
3.3 ExtractTiles . . . . .	29
3.3.1 Configuration . . . . .	29
3.4 GempakToCF . . . . .	30
3.4.1 Description . . . . .	30
3.4.2 Configuration . . . . .	30
3.5 GridStat . . . . .	30
3.5.1 Description . . . . .	30
3.5.2 Configuration . . . . .	30
3.6 MakePlots . . . . .	31
3.6.1 Description . . . . .	31
3.6.2 Configuration . . . . .	31
3.7 Mode . . . . .	32
3.7.1 Description . . . . .	32
3.7.2 Configuration . . . . .	33
3.8 MTD . . . . .	34
3.8.1 Description . . . . .	34
3.8.2 Configuration . . . . .	34
3.9 PB2NC . . . . .	34

<i>CONTENTS</i>	3
3.9.1 Description . . . . .	34
3.9.2 Configuration . . . . .	35
3.10 PcpCombine . . . . .	36
3.10.1 Description . . . . .	36
3.10.2 Configuration . . . . .	36
3.11 PointStat . . . . .	37
3.11.1 Description . . . . .	37
3.11.2 Configuration . . . . .	37
3.12 RegridDataPlane . . . . .	39
3.12.1 Description . . . . .	39
3.12.2 Configuration . . . . .	39
3.13 SeriesByInit . . . . .	39
3.13.1 Description . . . . .	39
3.13.2 Configuration . . . . .	40
3.14 SeriesByLead . . . . .	40
3.14.1 Description . . . . .	40
3.14.2 Configuration . . . . .	41
3.15 StatAnalysis . . . . .	41
3.15.1 Description . . . . .	41
3.15.2 Configuration . . . . .	42
3.16 TcPairs . . . . .	44
3.16.1 Description . . . . .	44
3.16.2 Configuration . . . . .	44
3.17 TcStat . . . . .	45
3.17.1 Description . . . . .	45

<i>CONTENTS</i>	4
3.17.2 Configuration	45
3.18 TCMPRPlotter	47
3.18.1 Description	47
3.18.2 Configuration	47
<b>4 System Configuration</b>	<b>49</b>
4.1 Config Best Practices	49
4.2 Config File Structure	50
4.3 Common Config Variables	50
4.3.1 Timing Control	50
4.3.1.1 LOOP_BY	50
4.3.1.2 Looping by Valid Time	51
4.3.1.3 Looping by Initialization Time	52
4.3.1.4 Looping over Forecast Leads	53
4.3.1.5 Realtime Looping	54
4.3.2 Field Info	56
4.3.3 Directory and Filename Template Info	59
4.4 Config Quick Start Example	62
4.5 User Defined Config	65
4.6 A-Z Config Glossary	65
4.6.1 A	66
4.6.2 B	67
4.6.3 C	68
4.6.4 D	71
4.6.5 E	72

4.6.6 F	77
4.6.7 G	96
4.6.8 H	99
4.6.9 I	99
4.6.10 J	101
4.6.11 K	102
4.6.12 L	102
4.6.13 M	106
4.6.14 N	116
4.6.15 O	117
4.6.16 P	135
4.6.17 Q	142
4.6.18 R	142
4.6.19 S	144
4.6.20 T	150
4.6.21 U	165
4.6.22 V	165
4.6.23 W	168
4.6.24 X	168
4.6.25 Y	169
4.6.26 Z	169

# Foreword: A note to METplus Wrappers users

This User's Guide is provided as an aid to users of the Model Evaluation Tools (MET) and its companion package METplus Wrappers. MET is a set of verification tools developed and supported to community via the Developmental Testbed Center (DTC) for use by the numerical weather prediction community. METplus Wrappers is intended to be a suite of Python wrappers and ancillary scripts to enhance the user's ability to quickly set-up and run MET. Over the next few years, METplus Wrappers will become the authoritative repository for verification of the Unified Forecast System.

It is important to note here that METplus Wrappers is an evolving software package. Previous releases of METplus Wrappers have occurred since 2017. This documentation describes the 2.1 release in May 2019. Intermediate releases may include bug fixes. METplus Wrappers is also be able to accept new modules contributed by the community. If you have code you would like to contribute, we will gladly consider your contribution. While we are setting up our community contribution protocol, please send email to: [met\\_help@ucar.edu](mailto:met_help@ucar.edu) and inform us of your desired contribution. We will then determine the maturity of any new verification method and coordinate the inclusion of the new module in a future version.

This User's Guide was prepared by the developers of the METplus Wrappers, including Dan Adriaansen, Minna Win-Gildenmeister, Julie Prestopnik, Jim Frimel, Mallory Row, John Halley Gotway, George McCabe, Paul Prestopnik, Christana Kalb, Hank Fisher, Jonathan Vigh, Lisa Goodrich, Tara Jensen, Tatiana Burek, and Bonny Strong.

## New for METplus Wrappers v2.2

### METplus Wrappers v2.2 Release Notes:

Configuration:

- Configuration variables that previously had to be specified in seconds (such as `OBS_WINDOW_END`) can now be specified in minutes or hours by adding M or H respectively to the number, i.e. `3600 = 60M = 1H`

- Non-MET executables (such as `rm`) no longer need to be specified with a full path in the configuration files. Default values are now the name of the executable. If the executable is found in the user environment path, the scripts will automatically find it.

#### Wrapper Specific:

- TcPairs wrapper has been updated to be more flexible and handle more use cases.

#### METplus Wrappers v2.1.1 Release Notes:

##### General:

- Fixed bug in filename templates where a file with the same number of characters in between tags are considered a valid file even if the characters differ, i.e. `template = {valid?fmt=%Y}_dog_{valid?fmt%m}` would treat both `2017_dog_02` and `2017_cat_02` as valid files when it should only consider `2017_dog_02` to be a valid file. This only applies when using file windows to determine which files are valid (See “Using Windows to Find Valid Files” in section [4.3.3](#)).
- Removed requirement for non-MET executables to exist on the system for wrappers that don’t need to use them. For example, if a use case doesn’t use `wgrib2`, setting the path to the executable correctly is not required.

#### METplus Wrappers v2.1 Release Notes:

##### Configuration:

- Added `gather_by_date.conf` for grid-to-grid and grid-to-obs use cases
- Created specific directory for plotting use case config files
- Added error checking so it is more clear what config items need to be added or changed to run
- Renamed many config variables for clarity
  - Variables specific to a wrapper will contain the app name in the variable name
  - Error message at startup listing all deprecated config items and the suggested new item to use
- MET configuration files updated to contain new default values that were recently added in the latest MET release
- Current time environment variables set to be used in MET or METplus Wrappers config files if needed (See section describing Timing Control [4.3.1](#)).
- Added `INIT_SEQ` for looping by valid time to calculate forecast lead list (instead of specifying with `LEAD_SEQ`)

- Added window begin and end variables for individual wrappers instead of having the same window for every wrapper (See OBS\_WINDOW\_BEGIN and OBS\_WINDOW\_END in section describing A-Z Config Glossary 4.6)
- Added option to skip processing of a file in some wrappers (i.e. pb2nc or regrid\_data\_plane) if the output file already exists (See PB2NC\_SKIP\_IF\_OUTPUT\_EXISTS in section describing A-Z Config Glossary 4.6)

Wrapper specific:

- MakePlots Wrapper
  - Reworked following the changes to the work done on StatAnalysis wrapper
- StatAnalysis Wrapper
  - Reworked to give users greater use of the MET stat\_analysis tool
- PcpCombine Wrapper
  - Added support for -derive mode
  - Fixed bugs in -subtract mode and -sum mode
- PB2NC Wrapper
  - Refactored to allow looping by valid time and utilize filename templates instead of regex
- PointStat Wrapper
  - Refactored to allow looping by valid time and utilize filename templates instead of regex
  - Added verification mask and neighborhood width/shape configurations
- EnsembleStat Wrapper
  - Refactored to allow looping by valid time
- GridStat Wrapper
  - Added verification mask and neighborhood width/shape configurations
- SeriesByLead Wrapper
  - Changed configuration names for forecast hour to more closely match other wrappers (See LEAD\_SEQ\_<n>, LEAD\_SEQ\_<n>\_LABEL, SERIES\_BY\_LEAD\_GROUP\_FCSTS, and LEAD\_SEQ in section describing A-Z Config Glossary 4.6)

General:

- Improved string template substitution and extraction functionality to be more flexible and to allow time shifting/truncating (See section describing Directory and Filename Template Info 4.3.3)

- Improve timed handling so filename templates can be more flexible
- Added 'now' time item to allow start and end time of run to be defined relative to the current time at execution (See section describing Timing Control [4.3.1](#))
- Logfiles now contains METplus Wrappers version number and start/end logs to easily discern multiple runs within a single log file
- Various bug fixes

# TERMS OF USE

## IMPORTANT!

USE OF THIS SOFTWARE IS SUBJECT TO THE FOLLOWING TERMS AND CONDITIONS:

1. **License.** Subject to these terms and conditions, University Corporation for Atmospheric Research (UCAR) grants you a non-exclusive, royalty-free license to use, create derivative works, publish, distribute, disseminate, transfer, modify, revise and copy the Model Evaluation Tools (MET) software, in both object and source code (the "Software").

You shall not sell, license or transfer for a fee the Software, or any work that in any manner contains the Software.

2. **Disclaimer of Warranty on Software.** Use of the Software is at your sole risk. The Software is provided "AS IS" and without warranty of any kind and UCAR EXPRESSLY DISCLAIMS ALL WARRANTIES AND/OR CONDITIONS OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT OF A THIRD PARTY'S INTELLECTUAL PROPERTY, MERCHANTABILITY OR SATISFACTORY QUALITY AND FITNESS FOR A PARTICULAR PURPOSE. THE PARTIES EXPRESSLY DISCLAIM THAT THE UNIFORM COMPUTER INFORMATION TRANSACTIONS ACT (UCITA) APPLIES TO OR GOVERNS THIS AGREEMENT. No oral or written information or advice given by UCAR or a UCAR authorized representative shall create a warranty or in any way increase the scope of this warranty. Should the Software prove defective, you (and neither UCAR nor any UCAR representative) assume the cost of all necessary correction.

3. **Limitation of Liability.** UNDER NO CIRCUMSTANCES, INCLUDING NEGLIGENCE, SHALL UCAR BE LIABLE FOR ANY DIRECT, INCIDENTAL, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES INCLUDING LOST REVENUE, PROFIT OR DATA, WHETHER IN AN ACTION IN CONTRACT OR TORT ARISING OUT OF OR RELATING TO THE USE OF OR INABILITY TO USE THE SOFTWARE, EVEN IF UCAR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

- 4. Compliance with Law.** All Software and any technical data delivered under this Agreement are subject to U.S. export control laws and may be subject to export or import regulations in other countries. You agree to comply strictly with all applicable laws and regulations in connection with use and distribution of the Software, including export control laws, and you acknowledge that you have responsibility to obtain any required license to export, re-export, or import as may be required.
- 5. No Endorsement/No Support.** The names UCAR/NCAR, National Center for Atmospheric Research and the University Corporation for Atmospheric Research may not be used in any advertising or publicity to endorse or promote any products or commercial entity unless specific written permission is obtained from UCAR. The Software is provided without any support or maintenance, and without any obligation to provide you with modifications, improvements, enhancements, or updates of the Software.
- 6. Controlling Law and Severability.** This Agreement shall be governed by the laws of the United States and the State of Colorado. If for any reason a court of competent jurisdiction finds any provision, or portion thereof, to be unenforceable, the remainder of this Agreement shall continue in full force and effect. This Agreement shall not be governed by the United Nations Convention on Contracts for the International Sale of Goods, the application of which is hereby expressly excluded.
- 7. Termination.** Your rights under this Agreement will terminate automatically without notice from UCAR if you fail to comply with any term(s) of this Agreement. You may terminate this Agreement at any time by destroying the Software and any related documentation and any complete or partial copies thereof. Upon termination, all rights granted under this Agreement shall terminate. The following provisions shall survive termination: Sections 2, 3, 6 and 9.
- 8. Complete Agreement.** This Agreement constitutes the entire agreement between the parties with respect to the use of the Software and supersedes all prior or contemporaneous understandings regarding such subject matter. No amendment to or modification of this Agreement will be binding unless in writing and signed by UCAR.
- 9. Notices and Additional Terms.** Copyright in Software is held by UCAR. You must include, with each copy of the Software and associated documentation, a copy of this Agreement and the following notice:

"The source of this material is the Research Applications Laboratory at the National Center for Atmospheric Research, a program of the University Corporation for Atmospheric Research (UCAR) pursuant to a Cooperative Agreement with the National Science Foundation; ©2007-2017 University Corporation for Atmospheric Research. All Rights Reserved."

**The following notice shall be displayed on any scholarly works associated with, related to or derived from the Software:**

*"Model Evaluation Tools (MET) and METplus were developed at the National Center for Atmospheric Research (NCAR) through grants from the National Science Foundation (NSF), the National Oceanic and Atmospheric Administration (NOAA), and the United States Air Force (USAF). NCAR is sponsored by the United States National Science Foundation."*

**By using or downloading the Software, you agree to be bound by the terms and conditions of this Agreement.**

The citation for this User's Guide should be:

Adriaansen, D., M. Win-Gildenmeister, J. Frimel, J. Prestopnik, J. Halley Gotway, T. Jensen, J. Vigh, C. Kalb, G. McCabe, and H. Fisher, 2018:  
The METplus Wrappers Version 2.1 User's Guide. Developmental Testbed Center.  
Available at: <https://github.com/NCAR/METplus/releases>. 85 pp.

# Acknowledgments

We thank the the National Science Foundation (NSF) along with three organizations within the National Oceanic and Atmospheric Administration (NOAA): 1) Office of Atmospheric Research (OAR); 2) Next Generation Global Prediction System project (NGGPS); and 3) United State Weather Research Program (USWRP) for their support of this work. Thanks also go to the staff at the Developmental Testbed Center for their help, advice, and many types of support. We released METplus Alpha in February 2017 and would not have made a decade of cutting-edge verification support without those who participated in DTC planning workshops and the NGGPS United Forecast System Strategic Implementation Plan Working Groups (NGGPS UFS SIP WGs).

The DTC is sponsored by the National Oceanic and Atmospheric Administration (NOAA), the United States Air Force, and the National Science Foundation (NSF). NCAR is sponsored by the National Science Foundation (NSF).

# Chapter 1

## Overview

### 1.1 Purpose and organization of the User's Guide

The goal of this User's Guide is to equip users with the information needed to use the Model Evaluation Tools (MET) and its companion package METplus Wrappers. MET is a set of verification tools developed and supported to community via the Developmental Testbed Center (DTC) for use by the numerical weather prediction community. METplus Wrappers is a suite of Python wrappers and ancillary scripts to enhance the user's ability to quickly set-up and run MET. Over the next few years, METplus Wrappers will become the authoritative repository for verification of the Unified Forecast System.

The METplus Wrappers User's Guide is organized as follows. Chapter 1 provides an overview of METplus Wrappers. Chapter 2 contains basic information about how to get started with METplus Wrappers - including system requirements, required software, and how to download METplus Wrappers. Chapter 4 provides information about configuring your environment and METplus Wrappers installation.

### 1.2 The Developmental Testbed Center (DTC)

METplus Wrappers has been developed, and will be maintained and enhanced, by the Developmental Testbed Center (DTC; <http://www.dtcenter.org/>). The main goal of the DTC is to serve as a bridge between operations and research, to facilitate the activities of these two important components of the numerical weather prediction (NWP) community. The DTC provides an environment that is functionally equivalent to the operational environment in which the research community can test model enhancements; the operational community benefits from DTC testing and evaluation of models before new models are implemented operationally. METplus Wrappers serves both the research and operational communities in this way - offering capabilities for researchers to test their own enhancements to models and providing a capability for the DTC to evaluate the strengths and weaknesses of advances in NWP prior to operational implementation.

METplus Wrappers will also be available to DTC visitors and to the WRF modeling community for testing and evaluation of new model capabilities, applications in new environments, and so on. The METplus Wrappers release schedule is coincident with the MET release schedule and the METplus Wrappers major release number is six less than the MET major release number (e.g. MET 8.X is released with METplus Wrappers 2.X).

### 1.3 METplus Wrappers goals and design philosophy

METplus Wrappers is a Python scripting infrastructure for the MET tools. The primary goal of METplus Wrappers development is to provide MET users with a highly configurable and simple means to perform model verification using the MET tools. Prior to the availability of METplus Wrappers, users who had more complex verifications that required the use of more than one MET tool were faced with setting up multiple MET config files and creating some automation scripts to perform the verification. METplus Wrappers provides the user with the infrastructure to modularly create the necessary steps to perform such verifications.

METplus Wrappers has been designed to be modular and adaptable. This is accomplished through wrapping the MET tools with Python and the use of hierarchical configuration files to enable users to readily customize their verification environments. Wrappers can be run individually, or as a group of wrappers that represent a sequence of MET processes. New wrappers can readily be added to the METplus Wrappers package due to this modular design. Currently, METplus Wrappers can easily be applied by any user on their own computer platform that supports Python 2.7.

The METplus Wrappers code and documentation is maintained by the DTC in Boulder, Colorado. METplus Wrappers is freely available to the modeling, verification, and operational communities, including universities, governments, the private sector, and operational modeling and prediction centers through a publicly accessible GitHub repository. Refer section 2.5 for simple examples of obtaining METplus Wrappers.

### 1.4 METplus Wrappers Components

The major components of the METplus Wrappers package are METplus Python wrappers to the MET tools, MET configuration files and a hierarchy of METplus Wrappers configuration files. Some Python wrappers do not correspond to a particular MET tool, but wrap utilities to extend METplus functionality.

### 1.5 Future development plans

METplus Wrappers is an evolving application. New capabilities are planned in controlled, successive version releases that are synchronized with MET releases. Bug fixes and user-identified problems will be addressed as they are found and posted to the known issues section of the METplus Wrappers Users web page (<https://>

[dtcenter.org/community-code/model-evaluation-tools-met](https://dtcenter.org/community-code/model-evaluation-tools-met)). Future METplus Wrappers development plans are based on several contributing factors, including the needs of both the operational and research community. Issues that are in the development queue detailed in the “Issues” section of the GitHub repository. Please send questions to [met\\_help@ucar.edu](mailto:met_help@ucar.edu).

## 1.6 Code support

Support for METplus Wrappers is provided through a MET-help e-mail address: [met\\_help@ucar.edu](mailto:met_help@ucar.edu). We will endeavor to respond to requests for help in a timely fashion. In addition, information about METplus Wrappers and tools that can be used with MET are provided on the MET Users web page (<https://dtcenter.org/community-code/model-evaluation-tools-met>).

We welcome comments and suggestions for improvements to METplus Wrappers, especially information regarding errors. Comments may be submitted using the MET Feedback form available on the MET website. In addition, comments on this document would be greatly appreciated. While we cannot promise to incorporate all suggested changes, we will certainly take all suggestions into consideration.

METplus Wrappers is a "living" set of wrappers and configuration files. Our goal is to continually enhance it and add to its capabilities. Because our time, resources, and talents are limited, we welcome contributed code for future versions of METplus. These contributions may represent new use cases or new plotting functions. For more information on contributing code to METplus Wrappers, please contact [met\\_help@ucar.edu](mailto:met_help@ucar.edu).

## Chapter 2

# Software Installation/Getting Started

### 2.1 Introduction

This chapter describes how to download and set up METplus Wrappers. METplus Wrappers has been developed and tested on the Debian Linux operating system.

### 2.2 Supported architectures

METplus Wrappers was developed on Debian Linux and is supported on this platform.

### 2.3 Programming/scripting languages

METplus Wrappers is written in Python 2.7. It is intended to be a tool for the modeling community to use and adapt. As users make upgrades and improvements to the tools, they are encouraged to offer those upgrades to the broader community by offering feedback to the developers or coordinating for a GitHub pull. For more information on contributing code to METplus Wrappers, please contact [met\\_help@ucar.edu](mailto:met_help@ucar.edu).

### 2.4 Pre-requisites

The following software is required to run METplus Wrappers:

- Python 2.7

- R version 3.2.5 <sup>1</sup>
- nco (netCDF operators)
- MET version 8.1 or above
- Basic familiarity with MET
- GitHub account (if you plan on contributing code to METplus Wrappers)

## 2.5 Getting the METplus Wrappers source code

The METplus Wrappers source code is available for download from a public GitHub repository. You can retrieve the source code through your web browser or the command line.

### 2.5.1 Get the source code via your Web Browser

If you are a new METplus Wrappers user and would like to experiment with the use cases, you will want to follow these instructions to retrieve the source code, additional documentation and sample data that accompanies the use cases:

- On your local host (or wherever you wish to install the METplus Wrappers code) create a directory where you want the code to reside
- Open the browser of your choice and navigate to <https://github.com/NCAR/METplus>. You will see something like the following:

---

<sup>1</sup>R version 3.2.5 is required when the TCMRPlotter wraps the `plot_tcmpr.R` script. Please refer to Chapter 21 Plotting and Graphics Support for more information about `plot_tcmpr.R`.

Search or jump to...

[Pull requests](#)
[Issues](#)
[Marketplace](#)
[Explore](#)

---

NCAR / **METplus**

[Unwatch](#) 17
 [Star](#) 17
 [Fork](#) 5

<> Code
Issues 55
Pull requests 0
Projects 0
Wiki
Insights
Settings

Python scripting infrastructure for MET tools. [Edit](#)

[Manage topics](#)

1,037 commits
14 branches
13 releases
7 contributors

Branch: **master** [New pull request](#)

[Create new file](#)
[Upload files](#)
[Find file](#)
[Clone or download](#)

CPKalb Update grid2grid\_hwt\_env.conf
Latest commit a724bbc 2 hours ago

<b>doc</b>	Merge branch 'develop' of https://github.com/NCAR/METplus into develop	8 months ago
<b>internal_tests</b>	Updated to reflect changes in the conf file.	5 months ago
<b>parm</b>	Update grid2grid_hwt_env.conf	2 hours ago
<b>sorc</b>	Change name from Alpha-produtil to Beta-METplus.	11 months ago
<b>ush</b>	Accidentally committed GitHub Issue #92 changes to master branch. Rev...	2 months ago
<b>.gitignore</b>	Initial commit	2 years ago
<b>.travis.yml</b>	try ./ since we don't know how directory is structured in the travis ...	6 months ago
<b>README.md</b>	Updated top-level README .	a year ago

README.md

## METplus Repository README File {#METplus\_README}

---

Welcome to the documentation for the Model Evaluation Tools Plus (METplus).

This is the METplus repository Top level README.md

### Basic DOCUMENTATION - getting started

---

ALL Documentation specific to this repository can be found in the doc/ directory.

The ORIGINAL setup text documentation in a markdown file is found here.

- doc/README\_install.md — installation, configuration, running
- doc/README\_terms\_of\_use.md — legal Terms Of Use

---

METplus is a Python scripting infrastructure around the MET verification tools (and eventually METViewer, a tool used for plotting MET output verification statistics).

This infrastructure utilizes the NCEP produtil package. A Platform-independent weather and ocean forecasting utility package. Developed at the National Oceanic and Atmospheric Administration (NOAA).

### Website Documentation

---

Users can generate an entire METplus documentation website, if they have Doxygen version 1.8.9.1 or later installed.

```
cd METplus/sorc
make doc
In your browser, open the page METplus/doc/html/index.html
```

---

### Terms of Use

---

[@ref METplus\\_TermsOfUse](#)

### Install/Configure/Run Guide

---

[@ref METplus\\_install\\_guide](#)

### Produtil Guide

---

[@ref METplus\\_produtil](#)

- Click on the 'releases' link, highlighted by a red circle in the diagram below:

The screenshot shows the GitHub repository page for NCAR/METplus. The repository description is "Python scripting infrastructure for MET tools." The repository statistics show 1,493 commits, 14 branches, 14 releases (circled in red), and 7 contributors. The current branch is 'develop'. The commit history shows a commit by georgemccabe titled "Changed time range and updated old variable name" from 3 hours ago. Below the commit history, there are links for 'doc', 'internal\_tests', 'parm', 'sorc', 'ush', '.gitignore', and 'README.md'.

- You will be redirected to another screen. The latest available release appears at the top of the screen:

The screenshot shows the GitHub repository page for NCAR/METplus, specifically the 'Releases' tab. The latest release is 'METplus-1.0', released on May 8 by JohnHalleyGotway. The release includes assets: 'sample\_data.tar.gz' (479 MB), 'Source code (zip)', and 'Source code (tar.gz)'. The commit message for this release is: "Merge remote-tracking branch 'origin/develop' Merging all changes from the develop branch to master branch in preparation for the METplus Gamma Release which will be cut from the master branch. This commit has been run through and passed all working unit and integration".

- Click on the 'Source code' link (either the *zip* or *tar.gz*) and when prompted, save it to the directory you created.

- Uncompress the source code (on Linux/Unix: *gunzip* for zip file or *tar xvfz* for the tar.gz file)
- Create a directory for the sample data directory for the use case you are interested in running
- Click on the sample data link for the use case you will run and when prompted, save the file to the directory you created above. Sample data are available for the following use cases:
  - Cyclone Track and Feature: *sample\_data-cyclone\_track\_feature.tgz*
  - Ensemble: *sample\_data-ensemble.tgz*
  - Grid-to-Grid: *sample\_data-grid\_to\_grid.tgz*
  - Grid-to-Obs: *sample\_data-grid\_to\_obs.tgz*
  - Mode: *sample\_data-mode.tgz*
  - QPF: *sample\_data-qpf.tgz*

### 2.5.2 Get the source code via Command line

- On your local host (or wherever you wish to install the METplus code) create a directory where you want the code to reside
- `cd` to the directory you just created.
- On the command line, enter the following:
  - *git clone https://github.com/NCAR/METplus*
  - The source code should appear under the METplus Wrappers directory
- To update your copy, `cd` to your METplus Wrappers install directory: `</path/to/>METplus` and enter *git pull* at the command line

## 2.6 METplus Wrappers directory structure

Once you have cloned the METplus Wrappers from the GitHub repository at <https://github.com/NCAR/METplus> to a location on your host, change directories to the METplus Wrappers directory. You should have the following directory structure:

```
METplus
  doc
  internal_tests
  parm
  src
  ush
  README.md
```

The top-level METplus Wrappers directory consists of a README.md file and several subdirectories.

The doc/ directory contains documentation for users (PDF) and Doxygen files that are used to create the developer documentaton. The Doxygen documentation can be created and viewed via web browser if the developer has Doxygen installed on the host.

The internal\_tests/ directory contains unit test scripts that are only relevant to METplus Wrappers developers and contributors.

The parm/ directory contains all the configuration files for MET and METplus Wrappers.

The src/ directory contains Doxygen executables to generate documentation for developers.

The src/ directory contains the source code for each of the wrappers in METplus Wrappers.

The ush/ directory contains the Python wrappers to the MET tools.

## 2.7 Set up your environment

Environment variables need to be set to allow the METplus Wrappers application to be run from any directory and for locating the necessary Python modules. There is an option to set the JLOGFILE environment variable, which indicates where JLOGS will be saved. JLOGS provide information pertinent to the configuration-file framework. If this environment is unset, then output from the configuration framework will be directed to stdout (your display).

Add the following information to your .cshrc (C shell) or .bashrc (Bash shell):

**.cshrc:**

- Open your .cshrc file and do the following:
- To your PATH, add: `<full-path-to-METplus>/ush`
- Optional: add JLOGFILE variable and set to `<full-path-to-save-jlog-files>`
- Close your .cshrc file and run `source ~/.cshrc`

```
e.g.
set path = (other_path_entries ~/METplus/ush
# optional
setenv JLOGFILE ~/jlog_out
```

**.bashrc:**

- Open your .bashrc file and do the following:
- To your PATH, add : `<full-path-to-METplus>/ush`
- Optional: add a JLOGFILE environment variable and set it to the directory where you want the logs to reside
- Close your .bashrc file and run `source ~/.bashrc`

*e.g.*

```
export PATH=~ /METplus/ush:$PATH
#optional
export JLOGFILE=~ /
```

## 2.8 Set up METplus Wrappers Configuration files

There are four METplus Wrappers configuration files that must be defined prior to running METplus Wrappers. These configuration files reside in the `METplus_INSTALL_DIRECTORY/METplus/parm/metplus_config`

The following configuration files are automatically loaded during a METplus Wrappers run and do not need to be invoked on the command line.

- `metplus_data.conf`
  - data-relevant settings:
    - \* filename templates
    - \* regular expressions for input or output filenames
    - \* directories where input data are located
- `metplus_logging.conf`
  - set logging levels for METplus and MET output
  - turn on/off logging to stdout (screen) or log files
- `metplus_runtime.conf`
  - runtime-related settings:
    - \* location of METplus master `_metplus.conf` file (the 'master' conf file that is a collection of all the final METplus configuration files)
- `metplus_system.conf`
  - system-related settings:
    - \* location of METplus source code

- \* location of MET source and build
- \* location of other non-MET executables/binaries
- \* location of METplus parm directory

They must be fully defined by replacing all variables preset to *</path/to>* with valid path names, or have those variables defined in a down-stream config file. If configuring METplus Wrappers in a common location for multiple users, it is recommended that the these four configuration files are fully defined. Individual users have the option to make customizations by over-riding any of these values in their own configuration files.

## 2.9 Running METplus Wrappers

Running METplus Wrappers involves invoking the Python script `master_metplus.py` from any directory followed by a list of configuration files (file path relative to the *<path\_to\_METplus\_install\_dir>/METplus/parm* directory).

Example: Using a “default” configuration

Copy and paste the following into an empty text file:

```
//This is a comment, comments are defined with a // at the beginning of the line.
// Setting the PROCESS_LIST to Usage indicates that we want usage information
[config]
PROCESS_LIST = Usage
// Set the MET_INSTALL_DIR to the location of the MET install
[dir]
MET_INSTALL_DIR = /usr/local/met-8.1
// Set these to any valid directory to override the </path/to> placeholder set in the metplus_system.conf
file
[dir]
TMP_DIR = /tmp
INPUT_BASE = /tmp
OUTPUT_BASE = /tmp
```

```
>master_metplus.py -c ./<my_user_config.conf>
```

or

```
>master_metplus.py -c /<username>/<my_user_config.conf>
```

if you saved your default config in a directory other than where you are running master\_metplus.py

A usage message appears, indicating that other config files are required to perform useful tasks and a list of currently supported wrappers:

```
USAGE: This is a default process, please indicate more specific processes in the PROCESS_LIST variable
in one or more of the following configuration files:
```

```
-parm/metplus_config/metplus_runtime.conf
```

```
-parm/metplus_use_cases/<usecase_name>/<usecase_name>.conf
```

```
-parm/metplus_use_cases/<usecase_name>/examples/<example_name>.conf  Currently  available
```

processes are:

- CyclonePlotter
- EnsembleStat
- ExtractTiles
- GempakToCF
- GridStat
- Mode
- MTD
- PB2NC
- PcpCombine
- PointStat
- RegridDataPlane
- SeriesByLead
- SeriesByInit
- TcPairs
- TcStat
- TCMPRPlotter

Example: Using a use-case configuration

```
>master_metplus.py -c use_cases/feature_relative/feature_relative.conf
```

Runs METplus using the defaults set in the three config files found in `parm/metplus_config`. Any variables defined in these three config files can be over-ridden in the `parm/use_cases/feature_relative/feature_relative.conf` file. METplus will run using the values specified in the `feature_relative.conf` file.

Example: Using example configuration to perform specific evaluation (e.g. Model 1 vs. Obs1, Model 1 vs Obs 2, Model 2 vs. Obs 1, etc.)

```
>master_metplus.py -c use_cases/feature_relative/feature_relative.conf \  
-c use_cases/feature_relative/example/series_by_lead_all_fhrs.conf
```

This runs METplus Wrappers using the defaults set in the three config files found in `parm/metplus_config`, where variables can be over-ridden by `parm/use_cases/feature_relative/feature_relative.conf` or in `parm/use_cases/feature_relative/example/series_by_lead_all_fhrs.conf`. The order in which conf files are called is important. Variables that are defined in intermediate conf files will be over-ridden by the same variables set in the conf file following it, or the last conf file.

# Chapter 3

## Python Wrappers

This chapter provides a description of each supported Python wrapper in METplus Wrappers. A wrapper is generally a Python script that encapsulates the behavior of a corresponding MET tool. Each of these sections can be added to the `PROCESS_LIST` configuration list variable. The Configuration section of each wrapper section below lists the METplus Wrappers configuration variables that are specific to that wrapper organized by config file section. You can find more information about each item in the A-Z Config Glossary (4.6).

### 3.1 CyclonePlotter

#### 3.1.1 Description

This wrapper does not have a corresponding MET tool but instead wraps the logic necessary to create plots of cyclone tracks. Currently only the output from the MET `tc-pairs` tool can be plotted.

#### 3.1.2 Configuration

[dir]

`CYCLONE_INPUT_DIR` 4.6.3

`CYCLONE_OUTPUT_DIR` 4.6.3

[config]

`CYCLONE_INIT_DATE` 4.6.3

`CYCLONE_INIT_HOUR` 4.6.3

`CYCLONE_MODEL` 4.6.3

CYCLONE\_PLOT\_TITLE [4.6.3](#)  
CIRCLE\_MARKER\_SIZE [4.6.3](#)  
CROSS\_MARKER\_SIZE [4.6.3](#)  
GENERATE\_TRACK\_ASCII [4.6.7](#)

**Deprecated:**

CYCLONE\_OUT\_DIR [4.6.3](#)

## 3.2 EnsembleStat

### 3.2.1 Description

Used to configure the MET tool ensemble\_stat.

### 3.2.2 Configuration

[dir]

OBS\_ENSEMBLE\_STAT\_POINT\_INPUT\_DIR [4.6.15](#)  
OBS\_ENSEMBLE\_STAT\_GRID\_INPUT\_DIR [4.6.15](#)  
FCST\_ENSEMBLE\_STAT\_INPUT\_DIR [4.6.6](#)  
ENSEMBLE\_STAT\_OUTPUT\_DIR [4.6.5](#)

[filename\_templates]

OBS\_ENSEMBLE\_STAT\_POINT\_INPUT\_TEMPLATE [4.6.15](#)  
OBS\_ENSEMBLE\_STAT\_GRID\_INPUT\_TEMPLATE [4.6.15](#)  
FCST\_ENSEMBLE\_STAT\_INPUT\_TEMPLATE [4.6.6](#)

[config]

ENSEMBLE\_STAT\_ONCE\_PER\_FIELD [4.6.5](#)  
FCST\_ENSEMBLE\_STAT\_INPUT\_DATATYPE [4.6.6](#)  
OBS\_ENSEMBLE\_STAT\_INPUT\_POINT\_DATATYPE [4.6.15](#)  
OBS\_ENSEMBLE\_STAT\_INPUT\_GRID\_DATATYPE [4.6.15](#)  
ENSEMBLE\_STAT\_GRID\_VX [4.6.5](#)  
ENSEMBLE\_STAT\_CONFIG\_FILE [4.6.5](#)  
ENSEMBLE\_STAT\_MET\_OBS\_ERROR\_TABLE [4.6.5](#)  
ENSEMBLE\_STAT\_N\_MEMBERS [4.6.5](#)  
OBS\_ENSEMBLE\_STAT\_WINDOW\_BEGIN [4.6.15](#)  
OBS\_ENSEMBLE\_STAT\_WINDOW\_END [4.6.15](#)

ENSEMBLE\_STAT\_ENS\_THRESH 4.6.5  
 ENS\_VAR<n>\_NAME (optional) 4.6.5  
 ENS\_VAR<n>\_LEVELS (optional) 4.6.5  
 ENS\_VAR<n>\_THRESH (optional) 4.6.5  
 ENS\_VAR<n>\_OPTIONS (optional) 4.6.5

**Deprecated:**

ENSEMBLE\_STAT\_OUT\_DIR 4.6.5  
 ENSEMBLE\_STAT\_CONFIG 4.6.5

## 3.3 ExtractTiles

### 3.5.1 Description

The ExtractTiles wrapper is used to regrid and extract subregions from paired tropical cyclone tracks that are created by the tc\_pairs\_wrapper. Unlike the other wrappers, the extract\_tiles\_wrapper does not correspond to a specific MET tool. It invokes the tc\_stat\_wrapper, which in turn calls the MET tc\_stat tool to determine the lat/lon positions of the paired track data. This information is then used to create tiles of subregions. The ExtractTiles wrapper creates a 2n degree x 2m degree grid/tile with each storm located at the center.

#### 3.3.1 Configuration

The following should be set in the METplus configuration file to define the dimensions and density of the tiles comprising the subregion:

[dir]

EXTRACT\_TILES\_OUTPUT\_DIR 4.6.5

[config]

LON\_ADJ 4.6.12

LAT\_ADJ 4.6.12

NLAT 4.6.14

NLON 4.6.14

DLON 4.6.4

DLAT 4.6.4

EXTRACT\_TILES\_FILTER\_OPTS 4.6.5

EXTRACT\_TILES\_VAR\_LIST 4.6.5

**Deprecated:**

EXTRACT\_OUT\_DIR 4.6.5

## 3.4 GempakToCF

### 3.4.1 Description

Used to configure the utility GempakToCF.

### 3.4.2 Configuration

[exe]

GEMPAKTOCF\_CLASSPATH [4.6.7](#)

[dir]

GEMPAKTOCF\_INPUT\_DIR [4.6.7](#)

GEMPAKTOCF\_OUTPUT\_DIR [4.6.7](#)

[filename\_templates]

GEMPAKTOCF\_INPUT\_TEMPLATE [4.6.7](#)

GEMPAKTOCF\_OUTPUT\_TEMPLATE [4.6.7](#)

[config]

GEMPAKTOCF\_SKIP\_IF\_OUTPUT\_EXISTS [4.6.7](#)

## 3.5 GridStat

### 3.5.1 Description

Used to configure the MET tool grid\_stat.

### 3.5.2 Configuration

[dir]

FCST\_GRID\_STAT\_INPUT\_DIR [4.6.6](#)

OBS\_GRID\_STAT\_INPUT\_DIR [4.6.15](#)

GRID\_STAT\_OUTPUT\_DIR [4.6.7](#)

[filename\_templates]

FCST\_GRID\_STAT\_INPUT\_TEMPLATE 4.6.6  
 OBS\_GRID\_STAT\_INPUT\_TEMPLATE 4.6.15  
 GRID\_STAT\_VERIFICATION\_MASK\_TEMPLATE (optional) 4.6.7

[config]

GRID\_STAT\_CONFIG\_FILE 4.6.7  
 FCST\_GRID\_STAT\_INPUT\_DATATYPE 4.6.6  
 OBS\_GRID\_STAT\_INPUT\_DATATYPE 4.6.15  
 GRID\_STAT\_ONCE\_PER\_FIELD 4.6.7  
 FCST\_GRID\_STAT\_PROB\_THRESH (optional) 4.6.6  
 OBS\_GRID\_STAT\_PROB\_THRESH (optional) 4.6.15  
 GRID\_STAT\_NEIGHBORHOOD\_WIDTH (optional) 4.6.7  
 GRID\_STAT\_NEIGHBORHOOD\_SHAPE (optional) 4.6.7  
 FCST\_GRID\_STAT\_WINDOW\_BEGIN (optional) 4.6.6  
 FCST\_GRID\_STAT\_WINDOW\_END (optional) 4.6.6  
 OBS\_GRID\_STAT\_WINDOW\_BEGIN (optional) 4.6.15  
 OBS\_GRID\_STAT\_WINDOW\_END (optional) 4.6.15

**Deprecated:**

GRID\_STAT\_OUT\_DIR 4.6.7  
 GRID\_STAT\_CONFIG 4.6.7

## 3.6 MakePlots

### 3.6.1 Description

The MakePlots wrapper creates various statistical plots using python scripts for the various METplus Wrappers use cases. This can only be run following StatAnalysis wrapper when LOOP\_ORDER = processes. To run MakePlots wrapper, include MakePlots in PROCESS\_LIST.

### 3.6.2 Configuration

The following values **must** be defined in the METplus Wrappers configuration file:

[dir]

PLOTTING\_SCRIPTS\_DIR 4.6.16  
 STAT\_FILES\_INPUT\_DIR 4.6.19  
 PLOTTING\_OUTPUT\_DIR 4.6.16

[config]

VERIF\_CASE 4.6.22  
VERIF\_TYPE 4.6.22  
PLOT\_TIME 4.6.16  
VALID\_BEG 4.6.22  
VALID\_END 4.6.22  
INIT\_BEG 4.6.9  
INIT\_END 4.6.9  
VALID\_HOUR\_METHOD 4.6.22  
VALID\_HOUR\_BEG 4.6.22  
VALID\_HOUR\_END 4.6.22  
VALID\_HOUR\_INCREMENT 4.6.22  
INIT\_HOUR\_METHOD 4.6.9  
INIT\_HOUR\_BEG 4.6.9  
INIT\_HOUR\_END 4.6.9  
INIT\_HOUR\_INCREMENT 4.6.9  
MODEL<n>\_NAME 4.6.13  
MODEL<n>\_OBS\_NAME 4.6.13  
MODEL<n>\_NAME\_ON\_PLOT 4.6.13  
FCST\_VAR<n>\_NAME 4.6.6  
FCST\_VAR<n>\_LEVELS 4.6.6  
REGION\_LIST 4.6.18  
LEAD\_LIST 4.6.12  
INTERP 4.6.9  
PLOT\_STATS\_LIST 4.6.16  
CI\_METHOD 4.6.3  
VERIF\_GRID 4.6.22  
EVENT\_EQUALIZATION 4.6.5

The following values are **optional** in the METplus Wrappers configuration file:

FCST\_VAR<n>\_THRESH 4.6.6  
FCST\_VAR<n>\_OPTIONS 4.6.6  
VAR<n>\_FOURIER\_DECOMP 4.6.22  
VAR<n>\_WAVE\_NUM\_LIST 4.6.22

## 3.7 Mode

### 3.7.1 Description

Used to configure the MET tool mode.

### 3.7.2 Configuration

[dir]

FCST\_MODE\_INPUT\_DIR [4.6.6](#)

OBS\_MODE\_INPUT\_DIR [4.6.15](#)

MODE\_OUTPUT\_DIR [4.6.13](#)

[filename\_templates]

FCST\_MODE\_INPUT\_TEMPLATE [4.6.6](#)

OBS\_MODE\_INPUT\_TEMPLATE [4.6.15](#)

[config]

MODE\_CONFIG\_FILE [4.6.13](#)

FCST\_MODE\_INPUT\_DATATYPE [4.6.6](#)

OBS\_MODE\_INPUT\_DATATYPE [4.6.15](#)

MODE\_QUILT [4.6.13](#)

MODE\_CONV\_RADIUS [4.6.13](#)

FCST\_MODE\_CONV\_RADIUS [4.6.6](#)

OBS\_MODE\_CONV\_RADIUS [4.6.15](#)

MODE\_CONV\_THRESH [4.6.13](#)

FCST\_MODE\_CONV\_THRESH [4.6.6](#)

OBS\_MODE\_CONV\_THRESH [4.6.15](#)

MODE\_MERGE\_THRESH [4.6.13](#)

FCST\_MODE\_MERGE\_THRESH [4.6.6](#)

OBS\_MODE\_MERGE\_THRESH [4.6.15](#)

MODE\_MERGE\_FLAG [4.6.13](#)

FCST\_MODE\_MERGE\_FLAG [4.6.6](#)

OBS\_MODE\_MERGE\_FLAG [4.6.15](#)

MODE\_MERGE\_CONFIG\_FILE [4.6.13](#)

FCST\_MODE\_WINDOW\_BEGIN [4.6.6](#)

FCST\_MODE\_WINDOW\_END [4.6.6](#)

OBS\_MODE\_WINDOW\_BEGIN [4.6.15](#)

OBS\_MODE\_WINDOW\_END [4.6.15](#)

#### Deprecated:

MODE\_OUT\_DIR [4.6.13](#)

MODE\_CONFIG [4.6.13](#)

## 3.8 MTD

### 3.8.1 Description

Used to configure the MET tool mtd (mode time domain).

### 3.8.2 Configuration

[dir]

FCST\_MTD\_INPUT\_DIR [4.6.6](#)

OBS\_MTD\_INPUT\_DIR [4.6.15](#)

MTD\_OUTPUT\_DIR [4.6.13](#)

[filename\_templates]

FCST\_MTD\_INPUT\_TEMPLATE [4.6.6](#)

OBS\_MTD\_INPUT\_TEMPLATE [4.6.15](#)

[config]

MTD\_CONFIG\_FILE [4.6.13](#)

MTD\_MIN\_VOLUME [4.6.13](#)

MTD\_SINGLE\_RUN [4.6.13](#)

MTD\_SINGLE\_DATA\_SRC [4.6.13](#)

FCST\_MTD\_INPUT\_DATATYPE [4.6.6](#)

OBS\_MTD\_INPUT\_DATATYPE [4.6.15](#)

FCST\_MTD\_CONV\_RADIUS [4.6.6](#)

FCST\_MTD\_CONV\_THRESH [4.6.6](#)

OBS\_MTD\_CONV\_RADIUS [4.6.15](#)

OBS\_MTD\_CONV\_THRESH [4.6.15](#)

#### Deprecated:

MTD\_OUT\_DIR [4.6.13](#)

MTD\_CONFIG [4.6.13](#)

## 3.9 PB2NC

### 3.9.1 Description

The PB2NC wrapper is a Python script that encapsulates the behavior of the MET pb2nc tool to convert prepBUFR files into netCDF.

### 3.9.2 Configuration

[dir]

PB2NC\_INPUT\_DIR [4.6.16](#)

PB2NC\_OUTPUT\_DIR [4.6.16](#)

[filename\_templates]

PB2NC\_INPUT\_TEMPLATE [4.6.16](#)

PB2NC\_OUTPUT\_TEMPLATE [4.6.16](#)

[config]

PB2NC\_SKIP\_IF\_OUTPUT\_EXISTS [4.6.16](#)

PB2NC\_OFFSETS [4.6.16](#)

PB2NC\_INPUT\_DATATYPE [4.6.16](#)

PB2NC\_CONFIG\_FILE [4.6.16](#)

PB2NC\_MESSAGE\_TYPE (optional) [4.6.16](#)

PB2NC\_STATION\_ID (optional) [4.6.16](#)

PB2NC\_GRID (optional) [4.6.16](#)

PB2NC\_POLY [4.6.16](#)

PB2NC\_OBS\_BUFR\_VAR\_LIST (optional) [4.6.16](#)

PB2NC\_TIME\_SUMMARY\_FLAG [4.6.16](#)

PB2NC\_TIME\_SUMMARY\_BEG [4.6.16](#)

PB2NC\_TIME\_SUMMARY\_END [4.6.16](#)

PB2NC\_TIME\_SUMMARY\_VAR\_NAMES [4.6.16](#)

PB2NC\_TIME\_SUMMARY\_TYPES [4.6.16](#)

PB2NC\_WINDOW\_BEGIN [4.6.16](#)

PB2NC\_WINDOW\_END [4.6.16](#)

#### Deprecated:

PREPBUFR\_DATA\_DIR [4.6.16](#)

PREPBUFR\_MODEL\_DIR\_NAME [4.6.16](#)

PREPBUFR\_DIR\_REGEX [4.6.16](#)

PREPBUFR\_FILE\_REGEX [4.6.16](#)

NC\_FILE\_TMPL [4.6.14](#)

PB2NC\_VERTICAL\_LEVEL [4.6.16](#)

OBS\_BUFR\_VAR\_LIST [4.6.15](#)

TIME\_SUMMARY\_FLAG [4.6.20](#)

TIME\_SUMMARY\_BEG [4.6.20](#)

TIME\_SUMMARY\_END [4.6.20](#)

TIME\_SUMMARY\_VAR\_NAMES [4.6.20](#)

TIME\_SUMMARY\_TYPE [4.6.20](#)

OVERWRITE\_NC\_OUTPUT [4.6.15](#)

VERTICAL\_LOCATION [4.6.22](#)

## 3.10 PcpCombine

### 3.10.1 Description

The PcpCombine wrapper is a Python script that encapsulates the MET pcp\_combine tool. It provides the infrastructure to combine or extract from files to build desired accumulations.

### 3.10.2 Configuration

[dir]

FCST\_PCP\_COMBINE\_INPUT\_DIR [4.6.6](#)

FCST\_PCP\_COMBINE\_OUTPUT\_DIR [4.6.6](#)

OBS\_PCP\_COMBINE\_INPUT\_DIR [4.6.15](#)

OBS\_PCP\_COMBINE\_OUTPUT\_DIR [4.6.15](#)

[filename\_templates]

FCST\_PCP\_COMBINE\_INPUT\_TEMPLATE [4.6.6](#)

FCST\_PCP\_COMBINE\_OUTPUT\_TEMPLATE [4.6.6](#)

OBS\_PCP\_COMBINE\_INPUT\_TEMPLATE [4.6.15](#)

OBS\_PCP\_COMBINE\_OUTPUT\_TEMPLATE [4.6.15](#)

[config]

FCST\_IS\_PROB [4.6.6](#)

OBS\_IS\_PROB [4.6.15](#)

FCST\_PCP\_COMBINE\_<n>\_FIELD\_NAME [4.6.6](#)

OBS\_PCP\_COMBINE\_<n>\_FIELD\_NAME [4.6.15](#)

FCST\_PCP\_COMBINE\_DATA\_INTERVAL [4.6.6](#)

OBS\_PCP\_COMBINE\_DATA\_INTERVAL [4.6.15](#)

FCST\_PCP\_COMBINE\_TIMES\_PER\_FILE [4.6.6](#)

OBS\_PCP\_COMBINE\_TIMES\_PER\_FILE [4.6.15](#)

FCST\_PCP\_COMBINE\_IS\_DAILY\_FILE [4.6.6](#)

OBS\_PCP\_COMBINE\_IS\_DAILY\_FILE [4.6.15](#)

FCST\_PCP\_COMBINE\_INPUT\_DATATYPE [4.6.6](#)

OBS\_PCP\_COMBINE\_INPUT\_DATATYPE [4.6.15](#)

FCST\_PCP\_COMBINE\_INPUT\_LEVEL [4.6.6](#)

OBS\_PCP\_COMBINE\_INPUT\_LEVEL 4.6.15  
FCST\_PCP\_COMBINE\_RUN 4.6.6  
OBS\_PCP\_COMBINE\_RUN 4.6.15  
FCST\_PCP\_COMBINE\_METHOD 4.6.6  
OBS\_PCP\_COMBINE\_METHOD 4.6.15  
FCST\_PCP\_COMBINE\_MIN\_FORECAST 4.6.6  
OBS\_PCP\_COMBINE\_MIN\_FORECAST 4.6.15  
FCST\_PCP\_COMBINE\_MAX\_FORECAST 4.6.6  
OBS\_PCP\_COMBINE\_MAX\_FORECAST 4.6.15  
FCST\_PCP\_COMBINE\_STAT\_LIST 4.6.6  
OBS\_PCP\_COMBINE\_STAT\_LIST 4.6.15  
FCST\_PCP\_COMBINE\_DERIVE\_LOOKBACK 4.6.6  
OBS\_PCP\_COMBINE\_DERIVE\_LOOKBACK 4.6.15  
PCP\_COMBINE\_SKIP\_IF\_OUTPUT\_EXISTS 4.6.16

**Deprecated:**

PCP\_COMBINE\_METHOD 4.6.16  
FCST\_MIN\_FORECAST 4.6.6  
FCST\_MAX\_FORECAST 4.6.6  
OBS\_MIN\_FORECAST 4.6.15  
OBS\_MAX\_FORECAST 4.6.15  
FCST\_DATA\_INTERVAL 4.6.6  
OBS\_DATA\_INTERVAL 4.6.15  
FCST\_IS\_DAILY\_FILE 4.6.6  
OBS\_IS\_DAILY\_FILE 4.6.15  
FCST\_TIMES\_PER\_FILE 4.6.6  
OBS\_TIMES\_PER\_FILE 4.6.15  
FCST\_LEVEL 4.6.6  
OBS\_LEVEL 4.6.15

## 3.11 PointStat

### 3.11.1 Description

The PointStat wrapper is a Python script that encapsulates the MET point\_stat tool. It provides the infrastructure to read in gridded model data and netCDF point observation data to perform grid-to-point (grid-to-obs) verification.

### 3.11.2 Configuration

[dir]

FCST\_POINT\_STAT\_INPUT\_DIR 4.6.6

OBS\_POINT\_STAT\_INPUT\_DIR 4.6.15

POINT\_STAT\_OUTPUT\_DIR 4.6.16

[filename\_templates]

FCST\_POINT\_STAT\_INPUT\_TEMPLATE 4.6.6

OBS\_POINT\_STAT\_INPUT\_TEMPLATE 4.6.15

POINT\_STAT\_VERIFICATION\_MASK\_TEMPLATE (optional) 4.6.16

[config]

POINT\_STAT\_OFFSETS 4.6.16

FCST\_POINT\_STAT\_INPUT\_DATATYPE 4.6.6

OBS\_POINT\_STAT\_INPUT\_DATATYPE 4.6.15

POINT\_STAT\_CONFIG\_FILE 4.6.16

MODEL 4.6.13

POINT\_STAT\_REGRID\_TO\_GRID 4.6.16

POINT\_STAT\_GRID 4.6.16

POINT\_STAT\_POLY 4.6.16

POINT\_STAT\_STATION\_ID 4.6.16

POINT\_STAT\_MESSAGE\_TYPE 4.6.16

FCST\_POINT\_STAT\_WINDOW\_BEGIN (optional) 4.6.6

FCST\_POINT\_STAT\_WINDOW\_END (optional) 4.6.6

OBS\_POINT\_STAT\_WINDOW\_BEGIN (optional) 4.6.15

OBS\_POINT\_STAT\_WINDOW\_END (optional) 4.6.15

POINT\_STAT\_NEIGHBORHOOD\_WIDTH (optional) 4.6.16

POINT\_STAT\_NEIGHBORHOOD\_SHAPE (optional) 4.6.16

#### Deprecated:

FCST\_INPUT\_DIR 4.6.6

OBS\_INPUT\_DIR 4.6.15

START\_HOUR 4.6.19

END\_HOUR 4.6.5

BEG\_TIME 4.6.2

FCST\_HR\_START 4.6.6

FCST\_HR\_END 4.6.6

FCST\_HR\_INTERVAL 4.6.6

OBS\_INPUT\_DIR\_REGEX 4.6.15

FCST\_INPUT\_DIR\_REGEX 4.6.6

FCST\_INPUT\_FILE\_REGEX 4.6.6

OBS\_INPUT\_FILE\_REGEX 4.6.15

OBS\_INPUT\_FILE\_TMPL 4.6.15

FCST\_INPUT\_FILE\_TMPL 4.6.6

REGRID\_TO\_GRID 4.6.18

## 3.12 RegridDataPlane

### 3.12.1 Description

Used to configure the MET tool `regrid_data_plane`.

### 3.12.2 Configuration

[dir]

FCST\_REGRID\_DATA\_PLANE\_INPUT\_DIR [4.6.6](#)

OBS\_REGRID\_DATA\_PLANE\_INPUT\_DIR [4.6.15](#)

[filename\_templates]

FCST\_REGRID\_DATA\_PLANE\_INPUT\_TEMPLATE [4.6.6](#)

OBS\_REGRID\_DATA\_PLANE\_INPUT\_TEMPLATE [4.6.15](#)

[config]

FCST\_REGRID\_DATA\_PLANE\_RUN [4.6.6](#)

OBS\_REGRID\_DATA\_PLANE\_RUN [4.6.15](#)

REGRID\_DATA\_PLANE\_SKIP\_IF\_OUTPUT\_EXISTS [4.6.18](#)

REGRID\_DATA\_PLANE\_VERIF\_GRID [4.6.18](#)

FCST\_REGRID\_DATA\_PLANE\_INPUT\_DATATYPE [4.6.6](#)

OBS\_REGRID\_DATA\_PLANE\_INPUT\_DATATYPE [4.6.15](#)

**Deprecated:**

VERIFICATION\_GRID [4.6.22](#)

## 3.13 SeriesByInit

### 3.13.1 Description

The `SeriesByInit` wrapper provides the infrastructure needed to perform a series analysis on tropical cyclone data, based on initialization times. The `SeriesByInit_wrapper` creates numerous plots that represent the field, level, and statistic for each initialization time.

### 3.13.2 Configuration

[dir]

SERIES\_BY\_INIT\_FILTERED\_OUTPUT\_DIR [4.6.19](#)

SERIES\_BY\_INIT\_OUTPUT\_DIR [4.6.19](#)

[regex\_patterns]

FCST\_TILE\_PREFIX [4.6.6](#)

ANLY\_TILE\_PREFIX [4.6.1](#)

FCST\_TILE\_REGEX [4.6.6](#)

ANLY\_TILE\_REGEX [4.6.1](#)

FCST\_NC\_TILE\_REGEX [4.6.6](#)

ANLY\_NC\_TILE\_REGEX [4.6.1](#)

FCST\_ASCII\_REGEX\_LEAD [4.6.6](#)

ANLY\_ASCII\_REGEX\_LEAD [4.6.1](#)

[config]

INIT\_BEG [4.6.9](#)

INIT\_END [4.6.9](#)

INIT\_INCREMENT [4.6.9](#)

INIT\_HOUR\_END [4.6.9](#)

INIT\_INCLUDE [4.6.9](#)

INIT\_EXCLUDE [4.6.9](#)

SERIES\_ANALYSIS\_FILTER\_OPTS [4.6.19](#)

**Deprecated:**

SERIES\_INIT\_FILTERED\_OUT\_DIR [4.6.19](#)

## 3.14 SeriesByLead

### 3.14.1 Description

The SeriesByLead wrapper provides the infrastructure needed to perform a series analysis on tropical cyclone data, based on lead (forecast hour) times. The SeriesByLead wrapper creates numerous plots that represent the field, level, and statistic for each lead (forecast) time. The SeriesByLead can be done in one of two ways: by all forecast hours or by forecast hour groupings. Performing a series analysis by valid time with forecast hour groupings can be useful when analyzing storm tracks based on time “bins” such as by days (eg. day 1, day 2, day 3, etc.).

### 3.14.2 Configuration

The input track and model data files are defined in any one of the user's METplus Wrappers configuration files. If creating a final configuration file that overrides all other config files, it is customary to define the `MODEL_DATA_DIR`, pointing to the directory where all model data resides. The full file path to the `INIT_INCLUDE` and `INIT_EXCLUDE` are used to list the times in `YYYYMMDD_HH` format to include or exclude from your time window. If these values are undefined (i.e. no value is set for the variable), then all available times in your time window will be considered. For example, if your data is available every 6 hours and you are interested in creating a series analysis from init time 20180601 to 20180615 for all available times, from 00z to 23z, you would set the following:

[dir]

`SERIES_BY_LEAD_FILTERED_OUTPUT` 4.6.19

`SERIES_BY_LEAD_OUTPUT_DIR` 4.6.19

[config]

`INIT_BEG` 4.6.9

`INIT_TIME_FMT` 4.6.9

`INIT_END` 4.6.9

`INIT_INCREMENT` 4.6.9

`SERIES_BY_LEAD_GROUP_FCSTS` 4.6.19

`LEAD_SEQ_<n>` 4.6.12

`LEAD_SEQ_<n>_LABEL` 4.6.12

`SERIES_ANALYSIS_FILTER_OPT` 4.6.19

`VAR_LIST` 4.6.22

`STAT_LIST` 4.6.19

#### Deprecated:

`SERIES_LEAD_FILTERED_OUT_DIR` 4.6.19

## 3.15 StatAnalysis

### 3.15.1 Description

The StatAnalysis wrapper encapsulates the behavior of the MET `stat_analysis` tool. It provides the infrastructure to summarize and filter the MET `.stat` files. StatAnalysis wrapper can be run in two different methods. First is to look at the STAT lines for a single date, to use this method set `LOOP_ORDER = times`. Second is to look at the STAT lines over a span of dates, to use this method set `LOOP_ORDER = processes`. To run StatAnalysis wrapper, include StatAnalysis in `PROCESS_LIST`.

### 3.15.2 Configuration

The following values must be defined in the METplus Wrappers configuration file for running with LOOP\_ORDER = times:

[dir]

STAT\_ANALYSIS\_LOOKIN\_DIR [4.6.19](#)  
 STAT\_ANALYSIS\_OUTPUT\_DIR [4.6.19](#)

[config]

LOOP\_BY [4.6.12](#)  
 [VALID/INIT]\_TIME\_FMT [4.6.22](#)  
 [VALID/INIT]\_BEG [4.6.22](#) [4.6.9](#)  
 [VALID/INIT]\_END [4.6.22](#) [4.6.9](#)  
 VALID\_HOUR\_METHOD [4.6.22](#)  
 VALID\_HOUR\_BEG [4.6.22](#)  
 VALID\_HOUR\_END [4.6.22](#)  
 VALID\_HOUR\_INCREMENT [4.6.22](#)  
 INIT\_HOUR\_METHOD [4.6.9](#)  
 INIT\_HOUR\_BEG [4.6.9](#)  
 INIT\_HOUR\_END [4.6.9](#)  
 INIT\_HOUR\_INCREMENT [4.6.9](#)  
 STAT\_ANALYSIS\_CONFIG [4.6.19](#)  
 MODEL [4.6.13](#)  
 OBTYPE [4.6.15](#)  
 JOB\_NAME [4.6.10](#)  
 JOB\_ARGS [4.6.10](#)

The following values are **optional** in the METplus Wrappers configuration file for running with LOOP\_ORDER = times:

DESC [4.6.4](#)  
 FCST\_LEAD [4.6.6](#)  
 FCST\_VAR<n>\_NAME [4.6.6](#)  
 FCST\_VAR<n>\_LEVEL [4.6.6](#)  
 OBS\_VAR<n>\_NAME [4.6.15](#)  
 OBS\_VAR<n>\_LEVEL [4.6.15](#)  
 REGION [4.6.18](#)  
 INTERP [4.6.9](#)  
 INTERP\_PTS [4.6.9](#)  
 FCST\_THRESH [4.6.6](#)  
 COV\_THRESH [4.6.3](#)  
 LINE\_TYPE [4.6.12](#)

STAT\_ANALYSIS\_DUMP\_ROW\_TMPL 4.6.19

STAT\_ANALYSIS\_OUT\_STAT\_TMPL 4.6.19

The following values **must** be defined in the METplus Wrappers configuration file for running with LOOP\_ORDER = processes:

STAT\_ANALYSIS\_OUTPUT\_DIR 4.6.19

VERIF\_CASE 4.6.22

VERIF\_TYPE 4.6.22

PLOT\_TIME 4.6.16

[VALID/INIT]\_BEG 4.6.22 4.6.9

[VALID/INIT]\_END 4.6.22 4.6.9

VALID\_HOUR\_METHOD 4.6.22

VALID\_HOUR\_BEG 4.6.22

VALID\_HOUR\_END 4.6.22

VALID\_HOUR\_INCREMENT 4.6.22

INIT\_HOUR\_METHOD 4.6.9

INIT\_HOUR\_BEG 4.6.9

INIT\_HOUR\_END 4.6.9

INIT\_HOUR\_INCREMENT 4.6.9

STAT\_ANALYSIS\_CONFIG 4.6.19

MODEL<n>\_NAME 4.6.13

MODEL<n>\_OBS\_NAME 4.6.13

MODEL<n>\_NAME\_ON\_PLOT 4.6.13

FCST\_VAR<n>\_NAME 4.6.6

FCST\_VAR<n>\_LEVELS 4.6.6

REGION\_LIST 4.6.18

LEAD\_LIST 4.6.12

INTERP 4.6.9

LINE\_TYPE 4.6.12

The following values are optional in the METplus Wrappers configuration file for running with LOOP\_ORDER = processes:

FCST\_VAR<n>\_THRESH 4.6.6

FCST\_VAR<n>\_THRESH 4.6.6

FCST\_VAR<n>\_OPTIONS 4.6.6

VAR<n>\_FOURIER\_DECOMP 4.6.22

VAR<n>\_WAVE\_NUM\_LIST 4.6.22

#### Deprecated:

STAT\_ANALYSIS\_OUT\_DIR 4.6.19

## 3.16 TcPairs

### 3.16.1 Description

The TcPairs wrapper encapsulates the behavior of the MET `tc_pairs` tool. The wrapper accepts Adeck and Bdeck (Best track) cyclone track data in extra tropical cyclone format (such as the data used by sample data provided in the METplus tutorial), or ATCF formatted track data. If data is in an extra tropical cyclone (non-ATCF) format, the data is reformatted into an ATCF format that is recognized by MET.

### 3.16.2 Configuration

[dir]

TC\_PAIRS\_ADECK\_INPUT\_DIR [4.6.20](#)

TC\_PAIRS\_BDECK\_INPUT\_DIR [4.6.20](#)

TC\_PAIRS\_EDECK\_INPUT\_DIR [4.6.20](#)

TC\_PAIRS\_OUTPUT\_DIR [4.6.20](#)

TC\_PAIRS\_REFORMAT\_DIR [4.6.20](#)

[filename\_templates]

TC\_PAIRS\_ADECK\_INPUT\_TEMPLATE [4.6.20](#)

TC\_PAIRS\_BDECK\_INPUT\_TEMPLATE [4.6.20](#)

TC\_PAIRS\_EDECK\_INPUT\_TEMPLATE [4.6.20](#)

TC\_PAIRS\_OUTPUT\_TEMPLATE [4.6.20](#)

[config]

TC\_PAIRS\_CONFIG\_FILE [4.6.20](#)

INIT\_BEG [4.6.9](#)

INIT\_END [4.6.9](#)

INIT\_INCREMENT [4.6.9](#)

INIT\_HOUR\_END [4.6.9](#)

INIT\_INCLUDE [4.6.9](#)

INIT\_EXCLUDE [4.6.9](#)

TC\_PAIRS\_READ\_ALL\_FILES [4.6.20](#)

TC\_PAIRS\_MODEL [4.6.20](#)

TC\_PAIRS\_STORM\_ID [4.6.20](#)

TC\_PAIRS\_BASIN [4.6.20](#)

TC\_PAIRS\_CYCLONE [4.6.20](#)

TC\_PAIRS\_STORM\_NAME [4.6.20](#)

TC\_PAIRS\_DLAND\_FILE 4.6.20  
TC\_PAIRS\_MISSING\_VAL\_TO\_REPLACE 4.6.20  
TC\_PAIRS\_MISSING\_VAL 4.6.20  
TC\_PAIRS\_SKIP\_IF\_REFACTOR\_EXISTS 4.6.20  
TC\_PAIRS\_SKIP\_IF\_OUTPUT\_EXISTS 4.6.20  
TC\_PAIRS\_REFORMAT\_DECK 4.6.20  
TC\_PAIRS\_REFORMAT\_TYPE 4.6.20

**Deprecated:**

ADECK\_TRACK\_DATA\_DIR 4.6.1  
BDECK\_TRACK\_DATA\_DIR 4.6.2  
TRACK\_DATA\_SUBDIR\_MOD 4.6.20  
TC\_PAIRS\_DIR 4.6.20  
TOP\_LEVEL\_DIRS 4.6.20  
MODEL 4.6.13  
STORM\_ID 4.6.19  
BASIN 4.6.2  
CYCLONE 4.6.3  
STORM\_NAME 4.6.19  
DLAND\_FILE 4.6.4  
TRACK\_TYPE 4.6.20  
ADECK\_FILE\_PREFIX 4.6.1  
BDECK\_FILE\_PREFIX 4.6.2  
MISSING\_VAL\_TO\_REPLACE 4.6.13  
MISSING\_VAL 4.6.13

## 3.17 TcStat

### 3.17.1 Description

Used to configure the MET tool `tc_stat`. This wrapper can be run by listing it in the `PROCESS_LIST`, or can be called from the `ExtractTiles` wrapper (via the MET `tc-stat` command line commands).

### 3.17.2 Configuration

[dir]

TC\_STAT\_INPUT\_DIR 4.6.20  
TC\_STAT\_OUTPUT\_DIR 4.6.20

[config]

TC\_STAT\_RUN\_VIA 4.6.20  
TC\_STAT\_CONFIG\_FILE 4.6.20  
TC\_STAT\_CMD\_LINE\_JOB 4.6.20  
TC\_STAT\_JOBS\_LIST 4.6.20  
TC\_STAT\_AMODEL 4.6.20  
TC\_STAT\_BMODEL 4.6.20  
TC\_STAT\_DESC 4.6.20  
TC\_STAT\_STORM\_ID 4.6.20  
TC\_STAT\_BASIN 4.6.20  
TC\_STAT\_CYCLONE 4.6.20  
TC\_STAT\_STORM\_NAME 4.6.20  
TC\_STAT\_INIT\_BEG 4.6.20  
TC\_STAT\_INIT\_INCLUDE 4.6.20  
TC\_STAT\_INIT\_EXCLUDE 4.6.20  
TC\_STAT\_INIT\_HOUR 4.6.20  
TC\_STAT\_VALID\_BEG 4.6.20  
TC\_STAT\_VALID\_END 4.6.20  
TC\_STAT\_VALID\_INCLUDE 4.6.20  
TC\_STAT\_VALID\_EXCLUDE 4.6.20  
TC\_STAT\_VALID\_HOUR 4.6.20  
TC\_STAT\_LEAD\_REQ 4.6.20  
TC\_STAT\_INIT\_MASK 4.6.20  
TC\_STAT\_VALID\_MASK 4.6.20  
TC\_STAT\_VALID\_HOUR 4.6.20  
TC\_STAT\_LEAD 4.6.20  
TC\_STAT\_TRACK\_WATCH\_WARN 4.6.20  
TC\_STAT\_COLUMN\_THRESH\_NAME 4.6.20  
TC\_STAT\_COLUMN\_THRESH\_VAL 4.6.20  
TC\_STAT\_COLUMN\_STR\_NAME 4.6.20  
TC\_STAT\_COLUMN\_STR\_VAL 4.6.20  
TC\_STAT\_INIT\_THRESH\_NAME 4.6.20  
TC\_STAT\_INIT\_THRESH\_VAL 4.6.20  
TC\_STAT\_INIT\_STR\_NAME 4.6.20  
TC\_STAT\_INIT\_STR\_VAL 4.6.20  
TC\_STAT\_WATER\_ONLY 4.6.20  
TC\_STAT\_LANDFALL 4.6.20  
TC\_STAT\_LANDFALL\_BEG 4.6.20  
TC\_STAT\_LANDFALL\_END 4.6.20  
TC\_STAT\_MATCH\_POINTS 4.6.20

## 3.18 TCMPRPlotter

### 3.18.1 Description

The TCMPRPlotter wrapper is a Python script that wraps the R script `plot_tmpr.R`. This script is useful for plotting the calculated statistics for the output from the MET-TC tools. This script, and other R scripts are included in the MET installation. Please refer to section 21.2.3 of the MET User's Guide for usage information.

### 3.18.2 Configuration

LOOP\_ORDER 4.6.12  
CONFIG\_FILE 4.6.3  
PREFIX 4.6.16  
TITLE 4.6.20  
SUBTITLE 4.6.19  
XLAB 4.6.24  
YLAB 4.6.25  
XLIM 4.6.24  
YLIM 4.6.25  
FILTER 4.6.6  
FILTERED\_TCST\_DATA\_FILE 4.6.6  
DEP\_VARS 4.6.4  
SCATTER\_X 4.6.4  
SCATTER\_Y 4.6.4  
SKILL\_REF 4.6.4  
SERIES 4.6.4  
SERIES\_CI 4.6.4  
LEGEND 4.6.12  
LEAD 4.6.12  
PLOT\_TYPES 4.6.16  
RP\_DIFF 4.6.18  
DEMO\_YR 4.6.4  
HFIP\_BASELINE 4.6.8  
FOOTNOTE\_FLAG 4.6.6  
PLOT\_CONFIG\_OPTS 4.6.16  
SAVE\_DATA 4.6.4

The following are TCMPR flags, if set to 'no', then don't set flag, if set to 'yes', then set the flag

NO\_EE 4.6.14  
NO\_LOG 4.6.14

SAVE 4.6.19

TCMPR\_DATA\_DIR 4.6.20

TCMPR\_PLOT\_OUTPUT\_DIR 4.6.20

**Deprecated:**

TCMPR\_PLOT\_OUT\_DIR 4.6.20

# Chapter 4

## System Configuration

This chapter is a guide on configuring METplus Wrappers.

### 4.1 Config Best Practices

Below is a list of Best Practices:

1. Set your log level to an appropriate level.
  - (a) Debug is the most verbose and is useful for developers and when you are troubleshooting problems
  - (b) Info is the less verbose than Debug and is the recommended level to initially set your log level
  - (c) Warning - only logs warnings, error or critical events
  - (d) Error - only logs errors or critical events
  - (e) Critical is the least verbose
2. Direct your logging either to stdout or to a log file.
3. Review your log file to verify that all your processes ran cleanly.
4. The order in which you list your METplus Wrappers config files matter. The last config file on the command line will over-ride any key-values defined in an earlier config file.
5. Check the master\_metplus.conf file, as it contains all the key-values based on what you have specified. This will help you determine whether you forgot to replace any `</path/to>` with valid paths or to verify that you have defined things as you expected.

## 4.2 Config File Structure

METplus Wrappers employs a hierarchy of configuration files employed in METplus Wrappers. At the lowest level are the “set-and-forget” type configuration files that reside in the `<METplus_installation_dir>/parm/metplus_config`. At the next level are the configuration files that pertain to a user’s specific needs in the `<METplus_installation_dir>/parm/use_cases/<specific_use_case>`

- Four configuration files are required for METplus Wrappers to be fully configured (i.e. all keywords are defined by either whitespace or a valid value):
  - `metplus_system`
  - `metplus_data`
  - `metplus_logging`
  - `metplus_runtime`

By default, key-values that require the user’s input are set to `</path/to>`. Make sure to replace these with the appropriate directory for your project.

- Additional configuration files are optional and the key-values defined there will over-ride any values defined in the four mandatory METplus Wrappers configuration files. These additional configuration files enables users to use a common set of configuration files and to create customized environments for their verification tasks.

## 4.3 Common Config Variables

### 4.3.1 Timing Control

This section describes the METplus Wrappers configuration variables that are used to control which times are processed. It also covers functionality that is useful for processing data in realtime by setting run times based on the clock time when METplus Wrappers is started.

#### 4.3.1.1 LOOP\_BY

METplus Wrappers can be configured to loop over a set of valid times or a set of initialization times. This is controlled by the METplus Wrappers configuration variable called `LOOP_BY`. If the value of this variable is set to `INIT` or `RETRO`, looping will be relative to initialization time. If the value is set to `VALID` or `REALTIME`, looping will be relative to valid time. Older versions of METplus Wrappers used a True/False variable named `LOOP_BY_INIT`. METplus Wrappers still supports using this variable, although it is recommended that users update their config files to use `LOOP_BY` instead. `LOOP_BY_INIT` will eventually be removed.

### 4.3.1.2 Looping by Valid Time

When looping over valid time (`LOOP_BY = VALID` or `LOOP_BY = REALTIME`), the following variables must be set:

#### **VALID\_TIME\_FMT**

This is the format of the valid times the user can configure in METplus Wrappers. The value of `VALID_BEG` and `VALID_END` must correspond to this format. Example: `VALID_TIME_FMT = %Y%m%d%H`. Using this format, the valid time range values specified must be defined as `YYYYMMDDHH`, i.e. `2019020112`.

#### **VALID\_BEG**

This is the first valid time that will be processed. The format of this variable is controlled by `VALID_TIME_FMT`. For example, if `VALID_TIME_FMT = %Y%m%d`, then `VALID_BEG` must be set to a valid time matching `YYYYMMDD`, such as `20190201`.

#### **VALID\_END**

This is the last valid time that can be processed. The format of this variable is controlled by `VALID_TIME_FMT`. For example, if `VALID_TIME_FMT = %Y%m%d`, then `VALID_END` must be set to a valid time matching `YYYYMMDD`, such as `20190202`. Note that the time specified for this variable will not necessarily be processed. It is used to determine the cutoff of run times that can be processed. For example, if METplus Wrappers is configured to start at `20190201` and end at `20190202` processing data in 48 hour increments, it will process valid time `20190201` then increment the run time to `20190203`. This is later than the `VALID_END` value, so execution will stop. However, if the increment is set to 24 hours (see `VALID_INCREMENT`), then METplus Wrappers will process valid times `20190201` and `20190202` before ending execution.

#### **VALID\_INCREMENT**

This is the number of seconds to add to each run time to determine the next run time to process. This value must be greater than or equal to 60 because METplus Wrappers currently does not support processing intervals less than one minute.

The following is a configuration that will process valid time `20190201` at 0Z until `20190202` at 0Z in 6 hour (21600 second) increments:

```
[config]
LOOP_BY = VALID
VALID_TIME_FMT = %Y%m%d%H
VALID_BEG = 2019020100
VALID_END = 2019020200
VALID_INCREMENT = 21600
```

This will process data valid on `20190201` at 0Z, 6Z, 12Z, and 18Z, as well as `20190202` at 0Z. For each of these valid times, METplus Wrappers can also loop over a set of forecast leads that are all valid at the current run time. See 'Looping Over Forecast Leads' [4.3.1.4](#) for more information.

### 4.3.1.3 Looping by Initialization Time

When looping over initialization time (`LOOP_BY = INIT` or `LOOP_BY = RETRO`), the following variables must be set:

#### **INIT\_TIME\_FMT**

This is the format of the initialization times the user can configure in METplus Wrappers. The value of `INIT_BEG` and `INIT_END` must correspond to this format. Example: `INIT_TIME_FMT = %Y%m%d%H`. Using this format, the initialization time range values specified must be defined as `YYYYMMDDHH`, i.e. `2019020112`.

#### **INIT\_BEG**

This is the first initialization time that will be processed. The format of this variable is controlled by `INIT_TIME_FMT`. For example, if `INIT_TIME_FMT = %Y%m%d`, then `INIT_BEG` must be set to an initialization time matching `YYYYMMDD`, such as `20190201`.

#### **INIT\_END**

This is the last initialization time that can be processed. The format of this variable is controlled by `INIT_TIME_FMT`. For example, if `INIT_TIME_FMT = %Y%m%d`, then `INIT_END` must be set to an initialization time matching `YYYYMMDD`, such as `20190202`. Note that the time specified for this variable will not necessarily be processed. It is used to determine the cutoff of run times that can be processed. For example, if METplus Wrappers is configured to start at `20190201` and end at `20190202` processing data in 48 hour increments, it will process `20190201` then increment the run time to `20190203`. This is later than the `INIT_END` value, so execution will stop. However, if the increment is set to 24 hours (see `INIT_INCREMENT`), then METplus Wrappers will process initialization times `20190201` and `20190202` before ending execution.

#### **INIT\_INCREMENT**

This is the number of seconds to add to each run time to determine the next run time to process. This value must be greater than or equal to 60 because METplus Wrappers currently does not support processing intervals less than one minute.

The following is a configuration that will process initialization time `20190201` at 0Z until `20190202` at 0Z in 6 hour (21600 second) increments:

```
[config]
LOOP_BY = INIT
INIT_TIME_FMT = %Y%m%d%H
INIT_BEG = 2019020100
INIT_END = 2019020200
INIT_INCREMENT = 21600
```

This will process data initialized on `20190201` at 0Z, 6Z, 12Z, and 18Z, as well as `20190202` at 0Z. For each of these initialization times, METplus Wrappers can also loop over a set of forecast leads that are all initialized at the current run time. See 'Looping Over Forecast Leads' [4.3.1.4](#) for more information.

#### 4.3.1.4 Looping over Forecast Leads

Many of the wrappers METplus Wrappers will also loop over a list of forecast leads relative to the current valid/initialization time that is being processed.

##### LEAD\_SEQ

This variable can be set to a comma-separated list of integers to define the forecast leads (hours) that will be processed relative to the initialization/valid time. Currently only hours are supported for these values. Future versions of METplus Wrappers will allow sub-hourly forecast leads. For example:

```
[config]
LEAD_SEQ = 3, 6, 9
```

If LOOP\_BY = VALID and the current run time is 20190201 at 0Z, then three times will be processed:

- 1) Initialized on 20190131 at 21Z / valid on 20190201 at 0Z
- 2) Initialized on 20190131 at 18Z / valid on 20190201 at 0Z
- 3) Initialized on 20190131 at 15Z / valid on 20190201 at 0Z

If LOOP\_BY = INIT and the current run time is 20190201 at 0Z, then three times will be processed:

- 1) Initialized on 20190201 at 0Z / valid on 20190201 at 3Z
- 2) Initialized on 20190201 at 0Z / valid on 20190201 at 6Z
- 3) Initialized on 20190201 at 0Z / valid on 20190201 at 9Z

##### Defining lists with begin\_end\_incr

You can also define LEAD\_SEQ using begin\_end\_incr(b, e, i) where b = the first value, e = the last value (inclusive), and i = the increment. For example:

```
LEAD_SEQ = begin_end_incr(0, 12, 3)
```

is equivalent to setting:

```
LEAD_SEQ = 0, 3, 6, 9, 12
```

##### INIT\_SEQ

If METplus Wrappers is configured to loop by valid time (LOOP\_BY = VALID), you can use INIT\_SEQ instead of LEAD\_SEQ. This is a list of initialization hours that are available in the data. This is useful if you know when the data is initialized and you need to use a different list of forecast leads depending on the valid time being evaluated. For example:

```
[config]
LOOP_BY = VALID
INIT_SEQ = 0, 6, 12, 18
```

At valid time 20190201\_0Z, this initialization sequence will build a forecast lead list of 0, 6, 12, 18, 24, 30, etc.

At valid time 20190201\_1Z, this initialization sequence will build a forecast lead list of 1, 7, 13, 19, 25, 31, etc.

You can also restrict the forecast leads that will be used by setting LEAD\_SEQ\_MIN and LEAD\_SEQ\_MAX. For example, if you want to only process forecast leads between 12 and 24 you can set:

```
[config]
LEAD_SEQ_MIN = 12
LEAD_SEQ_MAX = 24
```

Using the initialization sequence specified above, you will get the following results:

At valid time 20190201\_0Z, this initialization sequence will build a forecast lead list of 12, 18, 24.

At valid time 20190201\_1Z, this initialization sequence will build a forecast lead list of 13, 19.

Setting minimum and maximum values will also affect the list of forecast leads if you use LEAD\_SEQ. LEAD\_SEQ takes precedence over INIT\_SEQ, so if you have both variables set in your configuration, INIT\_SEQ will be ignored in favor of LEAD\_SEQ.

#### Grouping Forecast Leads (LEAD\_SEQ\_<n>)

This is used by SeriesByLead wrapper only. If SERIES\_BY\_LEAD\_GROUP\_FCSTS is set to True, then you can define groups of forecast leads that will be evaluated together. You can define any number of these groups by setting configuration variables LEAD\_SEQ\_1, LEAD\_SEQ\_2, LEAD\_SEQ\_3, etc. You can define the value with a comma-separated list of integers (hours) or using the begin\_end\_incr notation (See 'Defining lists with begin\_end\_incr' earlier in this section). Each list must have a corresponding label to describe it using LEAD\_SEQ\_<n>\_LABEL, i.e. LEAD\_SEQ\_1 must have the corresponding variable LEAD\_SEQ\_1\_LABEL set. For example:

```
[config]
SERIES_BY_LEAD_GROUP_FCSTS = True
LEAD_SEQ_1 = 0, 6, 12, 18
LEAD_SEQ_1_LABEL = Day1
LEAD_SEQ_2 = begin_end_incr(24, 42, 6)
LEAD_SEQ_2_LABEL = Day2
```

#### 4.3.1.5 Realtime Looping

To make running in realtime easier, METplus Wrappers supports defining the begin and end times relative to the current clock time. For example, if the current time is 20190426\_08:17 and you start METplus Wrappers with:

```
VALID_END = {now?fmt=%Y%m%d%H}
```

then the value of VALID\_END will be set to 2019042608. You can also use {today} to substitute the

current YYYYMMDD, i.e. 20190426. You cannot change the formatting for the 'today' keyword.

You can use the 'shift' keyword to shift the current time by any number of seconds. For example, if you start METplus Wrappers at the same clock time with:

```
VALID_BEG = {now?fmt=%Y%m%d%H?shift=-86400}
```

then the value of VALID\_BEG will be set to the current clock time shifted by -86400 (24 hours backwards), or 2019042508.

If VALID\_INCREMENT is set to 21600 (seconds or 6 hours), then METplus Wrappers will process valid times:

```
20190425_08
20190425_14
20190425_20
20190426_02
20190426_08
```

However, you may want to configure METplus Wrappers to process at 0Z, 6Z, 12Z, and 18Z of a given day instead of 2Z, 8Z, 14Z, and 20Z. Having to adjust the shift amount differently if you are running at 8Z or 9Z to get the times to line up would be tedious. Instead, use the 'truncate' keyword. The value set here is the number of seconds that is used to determine the interval of time to round down. If you want to process every 6 hours, set 'truncate' to 21600 (seconds).

```
VALID_BEG = {now?fmt=%Y%m%d%H?shift=-86400?truncate=21600}
```

This will round down the value to the nearest 6 hour interval of time. Starting METplus Wrappers on or after 6Z but before 12Z on 20190426 will result in VALID\_BEG = 2019042506 (clock time shifted backwards by 24 hours then truncated to 6 hours).

Starting METplus Wrappers on 20190426 at 8:16 with the following configuration:

```
[config]
LOOP_BY = VALID
VALID_TIME_FMT = %Y%m%d%H
VALID_BEG = {now?fmt=%Y%m%d%H?shift=-86400?truncate=21600}
VALID_END = {now?fmt=%Y%m%d%H}
VALID_INCREMENT = 21600
```

will process valid times starting on 20190425 at 6Z every 6 hours until the current run time is later than 20190426 at 8Z, which will result in processing:

```
20190425 at 6Z
```

20190425 at 12Z

20190425 at 18Z

20190426 at 0Z

20190426 at 6Z

**IMPORTANT NOTE:** When using the 'now' keyword, the value of `VALID_TIME_FMT` must be set to the same value as the 'fmt' used in the 'now' item in `VALID_BEG` and `VALID_END`. The relevant values are highlighted in bold in the above example.

### 4.3.2 Field Info

This section describes how METplus Wrappers configuration variables can be used to define field information that is sent to the MET applications to read forecast and observation fields.

#### **FCST\_VAR<n>\_NAME**

Set this to the name of a forecast variable that you want to evaluate. <n> is any integer greater or equal to 1, i.e.

```
[config]
FCST_VAR1_NAME = TMP
FCST_VAR2_NAME = RH
```

#### **FCST\_VAR<n>\_LEVELS**

Set this to a comma-separated list of levels or a single value. `FCST_VAR1_LEVELS` corresponds to `FCST_VAR1_NAME`, `FCST_VAR2_LEVELS` corresponds to `FCST_VAR2_NAME`, etc.

For example:

```
[config]
FCST_VAR1_NAME = TMP
FCST_VAR1_LEVELS = P500, P750
```

will process TMP at P500 and TMP at P750.

#### **OBS\_VAR<n>\_NAME**

Set this to the corresponding observation variable that you want to evaluate with `FCST_VAR<n>_NAME`. If this value is not set for a given <n> value, then it will be assumed that the same name as the forecast name will be used.

#### **OBS\_VAR<n>\_LEVELS**

Set this to a comma-separated list of levels or a single value. If `OBS_VAR<n>_LEVELS` and `OBS_VAR<n>_NAME` are not set, it will be assumed that the same name/level as the forecast data will be used.

For example, setting

```
[config]
FCST_VAR1_NAME = TMP
FCST_VAR1_LEVELS = P500
FCST_VAR2_NAME = RH
FCST_VAR2_LEVELS = P750, P250
```

without setting OBS\_VAR1\_NAME or OBS\_VAR2\_NAME is the equivalent of setting:

```
[config]
FCST_VAR1_NAME = TMP
FCST_VAR1_LEVELS = P500
FCST_VAR2_NAME = RH
FCST_VAR2_LEVELS = P750, P250
OBS_VAR1_NAME = TMP
OBS_VAR1_LEVELS = P500
OBS_VAR2_NAME = RH
OBS_VAR2_LEVELS = P750, P250
```

This will compare:

TMP/P500 in the forecast data to TMP/P500 in the observation data

RH/P750 in the forecast data to RH/P750 in the observation data

RH/P250 in the forecast data to RH/P250 in the observation data

If you set:

```
[config]
FCST_VAR1_NAME = TMP
FCST_VAR1_LEVELS = P500, P750
OBS_VAR1_NAME = TEMP
OBS_VAR1_LEVELS = "(0,*)", "(1,*)"
```

This will compare:

Forecast TMP/P500 to observation TEMP at (0,\*)

Forecast TMP/P750 to observation TEMP at (1,\*)

**IMPORTANT NOTE: NetCDF level values that contain (\*,\*) notation must be surrounded by quotation marks so it will not be misinterpreted as a list of items.**

The number of level items must be equal in each list for a given comparison. If you define separate names for a forecast and observation, you will need to define separate levels for both even if they are equivalent. For example, setting FCST\_VAR1\_NAME, FCST\_VAR1\_LEVELS, and OBS\_VAR1\_NAME, but not setting OBS\_VAR1\_LEVELS will result in an error.

The field information will be formatted to match the field info dictionary in the MET config files and passed to the appropriate config file to evaluate the data. The previous configuration comparing TMP (P500 and P750) and TEMP will generate the following in the MET config file:

```
fcst = {
field = [ { name="TMP"; level="P500"; } ];
}

obs = {
field = [ {name="TEMP"; level="(0,*)"; } ];
}
```

for the first run and:

```
fcst = {
field = [ { name="TMP"; level="P750"; } ];
}

obs = {
field = [ {name="TEMP"; level="(1,*)"; } ];
}
```

for the second run. Note that some MET applications allow multiple fields to be specified for a single run. If the MET tool allows it and METplus Wrappers is configured accordingly, these two comparisons would be configured in a single run.

**[FCST/OBS]\_VAR<n>\_THRESH:** Set this to a comma-separated list of threshold values to use in the comparison. Each of these values must begin with a comparison operator (>, >=, ==, !=, <, <=, gt, ge, eq, ne, lt, or le). For example, setting:

```
[config]
FCST_VAR1_NAME = TMP
FCST_VAR1_LEVELS = P500
FCST_VAR1_THRESH = le0.5, gt0.4, gt0.5, gt0.8
```

will add the following information to the MET config file:

```
fcst = {
field = [ { name="TMP"; level="P500"; cat_thresh=[ le0.5, gt0.4, gt0.5, gt0.8 ]; } ];
}
```

Note that the value for FCST\_VAR<n>\_THRESH will not be copied to OBS\_VAR<n>\_THRESH if it is not set. These items are independent of each other.

**[FCST/OBS]\_VAR<n>\_OPTIONS:** Set this to add additional information to field dictionary in the MET config file. The item must end with a semi-colon. For example:

```
[config]
FCST_VAR1_NAME = TMP
FCST_VAR1_LEVELS = P500
FCST_VAR1_OPTIONS = GRIB_lvl_typ = 105;
```

will add the following to the MET config file:

```
fcst = {
field = [ { name="TMP"; level="P500"; GRIB_lvl_typ = 105; } ];
}
```

You can specify multiple items as long as they are separated by and end with a semi-colon.

```
FCST_VAR1_NAME = TMP
FCST_VAR1_LEVELS = P500
FCST_VAR1_OPTIONS = GRIB_lvl_typ = 105; ens_phist_bin_size = 0.05;
```

will add the following to the MET config file:

```
fcst = {
field = [ { name="TMP"; level="P500"; GRIB_lvl_typ = 105; ens_phist_bin_size = 0.05; } ];
}
```

Note that the value for FCST\_VAR<n>\_OPTIONS will not be copied to OBS\_VAR<n>\_OPTIONS if it is not set. These items are independent of each other.

**ENS\_VAR<n>\_[NAME/LEVELS/THRESH/OPTIONS]:** Used with EnsembleStat only. Users may want to define the ens dictionary item in the MET EnsembleStat config file differently than the fcst dictionary item. If this is the case, you can use these variables. If it is not set, the values set in the corresponding FCST\_VAR<n>\_[NAME/LEVELS/THRESH/OPTIONS] will be used in the ens dictionary.

### 4.3.3 Directory and Filename Template Info

The METplus Wrappers use directory and filename template configuration variables to find the desired files for a given run.

#### Using Templates to Find Observation Data:

The following configuration variables describe input observation data:

```
[dir]
OBS_GRID_STAT_INPUT_DIR = /my/path/to/grid_stat/input/obs
[filename_templates]
OBS_GRID_STAT_INPUT_TEMPLATE = {valid?fmt=%Y%m%d}/prefix.{valid?fmt=%Y%m%d%H}.ext
```

The input directory is the top level directory containing all of the observation data. The template contains items with keywords that will be substituted with time values for each run. After the values are substituted,

METplus Wrappers will check to see if the desired file exists relative to the input directory. At valid time 20190201\_12Z, the full desired path of the observation input data to grid\_stat is:

```
/my/path/to/grid_stat/input/obs/20190201/prefix.2019020112.ext
```

Note that the template contains a dated subdirectory. This cannot go in the directory variable because it changes based on the run time.

METplus Wrappers does not need be configured to loop by valid time to find files using a template containing valid time information. For example, at initialization time 20190201\_12Z and forecast lead 3, the valid time is calculated to be 20190201\_15Z and the full desired path of the observation input data to grid\_stat is:

```
/my/path/to/grid_stat/input/obs/20190201/prefix.2019020115.ext
```

'init' and 'valid' are keywords to notate initialization and valid times respectively. 'lead', 'offset', 'da\_init', and 'cycle' are also keywords that can be used to find forecast data and data assimilation data.

### Using Templates to Find Forecast Data

Most forecast files contain the initialization time and the forecast lead in the filename. The keywords 'init' and 'lead' can be used to describe the template of these files:

```
[dir]
FCST_GRID_STAT_INPUT_DIR = /my/path/to/grid_stat/input/fcst
[filename_templates]
FCST_GRID_STAT_INPUT_TEMPLATE = prefix.{init?fmt=%Y%m%d%H}_f{lead?fmt=%3H}.ext
```

For a valid time of 20190201\_0Z and a forecast lead of 3, METplus Wrappers will look for the following forecast file with initialization time of 20190131\_21Z:

```
/my/path/to/grid_stat/input/fcst/prefix.2019013121_f003.ext
```

### Using Templates to Find Data Assimilation Data

Some data assimilation files contain offset and da\_init (data assimilation initialization) values in the filename. These values are used to determine the valid time of the data. Consider the following configuration:

```
[config]
PB2NC_OFFSETS = 6, 3
[dir]
PB2NC_INPUT_DIR = /my/path/to/prepbuf
[filename_templates]
PB2NC_INPUT_TEMPLATE = prefix.{da_init?fmt=%Y%m%d}_f{cycle?fmt=%H}_off{offset?fmt=%2H}.ext
```

The PB2NC\_OFFSETS list tells METplus Wrappers the order to prioritize files with offsets in the name.

At valid time 20190201\_12Z, METplus Wrappers will check if the following file exists:

```
/my/path/to/prepbufr/prefix.20190201_18_off06.ext
```

The offset is added to the valid time to get the data assimilation initialization time. Note that 'cycle' can be used interchangeably with 'da\_init.' It is generally used to specify the hour of the data that was generated. If that file doesn't exist, it will check:

```
/my/path/to/prepbufr/prefix.20190201_15_off03.ext
```

### Shifting Times in Filename Templates

Users can use the 'shift' keyword to adjust the time referenced in the filename template relative to the run time. For example, the input files contain data from 1Z on the date specified in the filename until 1Z the following day. This means that for a 0Z run you want to use the file from the previous day and for the 1Z to 23Z runs you want to use the file that corresponds to the current day.

```
[filename_templates]
OBS_POINT_STAT_INPUT_TEMPLATE = {valid?fmt=%Y%m%d?shift=-3600}.ext
```

Running at valid time 20190201\_12Z, shifting the valid time backwards by 1 hour will result in 20190201\_11Z, so it will substitute the current day into the template, giving you 20190201.ext. Running at valid time 20190201\_0Z, the shift will result in a file time of 20190131\_23Z, so 20190131.ext will be generated by the template.

### Using Windows to Find Valid Files

The [FCST/OBS]\_FILE\_WINDOW\_[BEGIN/END] configuration variables can be used if the time information in the input data does not exactly line up with the run time but you still want to process the data. The default value of the file window begin and end variables are both 0 (seconds). If both values are set to 0, METplus Wrappers will require that a file matching the template with the exact time requested exists. If either value is non-zero, METplus Wrappers will examine all of the files under the input directory that match the template, pull out the time information from the files, and use the file with time closest to the run time. For example, consider this configuration:

```
[config]
OBS_FILE_WINDOW_BEGIN = -7200
OBS_FILE_WINDOW_END = 7200
[dir]
OBS_GRID_STAT_INPUT_DIR = /my/grid_stat/input/obs
[filename_templates]
OBS_GRID_STAT_INPUT_TEMPLATE = {valid?fmt=%Y%m%d}/pre.{valid?fmt=%Y%m%d}_{valid?fmt=%H}.ext
```

The run time is 20190201\_0Z and the following files exist in the input directory:

```

/my/grid_stat/input/obs/20190131/pre.20190131_22.ext
/my/grid_stat/input/obs/20190131/pre.20190131_23.ext
/my/grid_stat/input/obs/20190201/othertype.20190201_00.ext
/my/grid_stat/input/obs/20190201/pre.20190201_01.ext
/my/grid_stat/input/obs/20190201/pre.20190201_03.ext

```

\* The first file matches the template and the file time is within the window, so the filename and time difference relative to the valid time (7200 seconds or 2 hours) is saved.

\* The second file matches the template, the file time is within the window, and the time difference is less than the closest file, so the filename and time difference from the valid time (3600 seconds or 1 hour) is saved.

\* The third file does not match the template, so it is ignored.

\* The fourth file matches the template and is within the time range, but it is the same distance away from the valid time as the closest file. GridStat only allows one file to be processed, so it is ignored (PB2NC is currently the only wrapper to allow multiples files to be processed).

\* The fifth file matches the template but it is valid outside of the -2 to +2 hour window range, so it is ignored.

METplus Wrappers uses /my/grid\_stat/input/obs/20190131/pre.20190131\_23.ext as the input to grid\_stat

### Wrapper Specific Windows

A user may need to specify a different window on a wrapper-by-wrapper basis. If this is the case, you can override the file window values for each wrapper. Consider the following configuration:

```

[config]
PROCESS_LIST = PcpCombine, GridStat, EnsembleStat
OBS_FILE_WINDOW_BEGIN = 0
OBS_FILE_WINDOW_END = 0
OBS_GRID_STAT_FILE_WINDOW_BEGIN = -1800
OBS_GRID_STAT_FILE_WINDOW_END = 1800
OBS_ENSEMBLE_STAT_FILE_WINDOW_END = 3600

```

Using this configuration, PcpCombine will use (0,0) and require exact file times. GridStat will use (-1800, 1800) for observation data. EnsembleStat will use (0, 3600) for observation data. OBS\_ENSEMBLE\_STAT\_FILE\_WINDOW was not set, so it will use OBS\_FILE\_WINDOW\_BEGIN.

## 4.4 Config Quick Start Example

### Track and Intensity Use case with sample data

- Create a directory where you wish to store the sample data. Sample datasets are specific to each use

case (see 2.5.1) and are required in order to be able to run the use case.

- Retrieve the sample data from the GitHub repository:
  - In your browser, navigate to <https://github.com/NCAR/METplus/releases>
  - locate the latest release and click on the *sample\_data-cyclone\_track\_feature.tgz* link associated with that release
  - save it to the directory you created above, hereafter referred to as `INPUT_DATA_DIRECTORY`
  - cd to your `$INPUT_DATA_DIRECTORY` and uncompress the tarball: `tar xvfz sample_data-cyclone_track_feature.tgz`
  - when you perform a listing of the `sample_data` directory, the `INPUT_DATA_DIRECTORY/sample_data/GFS` contains the data you will need for this use case
  
- Set up the configuration file:
  - Your METplus Wrappers install directory will hereafter be referred to as `METplus_INSTALL`
  - Verify that all the `<path/to>` values are replaced with valid paths in the `METplus_INSTALL/parm/metplus_conf/` and `METplus_INSTALL/parm/metplus_conf/metplus_system.conf` files
  - Two configuration files are used in this use case, `track_and_intensity.conf` file and `tcmp_mean_median.conf` to take cyclone track data, and using TcPairs which wraps the MET TC-Pairs tool (to match ADeck and BDeck cyclone tracks to generate matched pairs and error statistics). The TCM-PRPlotter is then used (wraps the MET tool `plot_tcmp.R`) to generate a mean and median plots for these matched pairs.
  - In your editor, open the `METplus_INSTALL/METplus/parm/use_cases/track_and_intensity.conf` file:
    - \* You will replace any `</path/to>` with actual paths by setting the following:
    - \* `OUTPUT_BASE` to where you wish to save the output:
      - `TC_PAIRS_ADECK_INPUT_DIR` to `INPUT_DATA_DIRECTORY/sample_data/GFS/track_data`
    - \* save your changes and exit your editor
    - \* In your editor, open the `METplus_INSTALL/METplus/parm/use_cases/track_and_intensity/examples/tcmp`
    - \* Verify that `PROCESS_LIST` is set to TcPairs, TCMPRPlotter. This instructs METplus Wrappers to run the TcPairs wrapper first (TC-Pairs) followed by the TCMPR plotter wrapper (`plot_TCMPR.R`).
  
- Run the use case:
  - Make sure you have set the following environment in your `.cshrc` (C shell) or `.bashrc` (Bash):
    - \* csh: `setenv RSCRIPTS_BASE $MET_BASE/scripts/Rscripts`
    - \* bash: `export RSCRIPTS_BASE $MET_BASE/scripts/Rscripts`
    - \* Refer to Section 2.7 for the full instructions on setting up the rest of your environment
    - \* on your command line, run:
      - `master_metplus.py -c use_cases/track_and_intensity/track_and_intensity.conf -c use_cases/track_and`

- \* When complete, you will have a log file in the output directory you specified, and under the `tc_pairs` directory you will see `.tcst` files under the `201412` subdirectory. These are the matched pairs created by the MET tool `Tc-pairs` and can be viewed in any text editor.
- \* Plots are generated under the `tcmpr_plots` subdirectory, in `.png` format. You should have the following plots which can be viewed by any graphics viewers such as `'display'` on Linux/Unix hosts:
  - `AMAX_WIND-BMAX_WIND_mean.png`
  - `AMAX_WIND-BMAX_WIND_median.png`
  - `AMSLP-BMSLP_mean.png`
  - `AMSLP-BMSLP_median.png`
  - `TK_ERR_mean.png`
  - `TK_ERR_median.png`

## 4.5 User Defined Config

You can define your own custom config variables that will be set as environment variables when METplus is run. MET config files can read environment variables, so this is a good way to customize information that is read by those files. To create add a custom config variable, add a section to one of your METplus config files called [user\_env\_vars]. Under this header, add as many variables as you'd like. For example, if you added the following to your METplus config:

```
[user_env_vars]
VAR_NAME = some_text_for_feb_1_1987_run
```

and you added the following to a MET config file that is used:

```
output_prefix = ${VAR_NAME}
```

then at run time, the MET application will be run with the configuration:

```
output_prefix = some_text_for_feb_1_1987_run
```

You can also reference other variables in the METplus config file. For example:

```
[config]
INIT_BEG = 1987020104
...
[user_env_vars]
USE_CASE_TIME_ID = {INIT_BEG}
```

This is the equivalent of calling

```
export USE_CASE_TIME_ID=1987020104
```

at the beginning of your METplus run. You can access the variable in the MET config file with \${USE\_CASE\_TIME\_ID}.

## 4.6 A-Z Config Glossary

This glossary was created from the two commands:

```
$ cat METplus/parm/metplus_config/*.conf METplus/parm/use_cases/*.conf METplus/parm/use_cases/*/*/*.conf
> allopts.conf
$ grep = allopts.conf | grep -v \# | sort | uniq > uniqueopts.conf
```

General form of glossary entry:

**CONFIG\_NAME\_HERE**

...Some description here...

*Used by:* Which METplus utility is this used by?

*Family:* Which family? [dir], [config], [filename\_temUpdates], [exe], [regex\_pattern], etc...

*Default:* If it makes sense to include a default value (or value shipped in a release), do it here

---

**4.6.1 A****[deprecated] ADECK\_FILE\_PREFIX**

Please use TC\_PAIRS\_ADECK\_TEMPLATE..

*Used by:* TcPairs

*Family:* [config]

*Default:* Varies

---

**[deprecated] ADECK\_TRACK\_DATA\_DIR**

Please use TC\_PAIRS\_ADECK\_INPUT\_DIR.

*Used by:* TcPairs

*Family:* [dir]

*Default:* Varies

---

**[deprecated] AMODEL**

Please use TC\_STAT\_AMODEL.

*Used by:* CyclonePlotter, TcStat

*Family:* [config]

*Default:*

---

**ANLY\_ASCII\_REGEX\_LEAD**

The regular expression describing the analysis (obs) file name (in ASCII format) of the intermediate file generated when running a series by lead case.

*Used by:* SeriesByLead

*Family:* [regex\_pattern]

*Default:*

---

**ANLY\_NC\_TILE\_REGEX**

The regular expression used to search the input files that are in netCDF format and used in the series by analysis task.

*Used by:* SeriesByLead, SeriesByInit

*Family:* [regex\_pattern]

*Default:*

---

#### **ANLY\_TILE\_PREFIX**

The prefix to the filename for the analysis file that is created as part of a series analysis.

*Used by:* ExtractTiles, SeriesByLead

*Family:* [regex\_pattern]

*Default:*

---

#### **ANLY\_TILE\_REGEX**

The regular expression for the analysis input file the file is in GRIB2.

*Used by:* SeriesByLead, SeriesByInit

*Family:* [regex\_pattern]

*Default:*

### **4.6.2 B**

---

#### **BACKGROUND\_MAP**

Control whether or not a background map shows up for series analysis plots. Set to 'yes' if background map desired.

*Used by:* SeriesByLead, SeriesByInit

*Family:* [config]

*Default:* no

---

#### **[deprecated] BASIN**

Please use TC\_PAIRS\_BASIN or TC\_STAT\_BASIN.

*Used by:* TcPairs, TcStat

*Family:* [config]

*Default:* Varies

---

#### **[deprecated] BDECK\_FILE\_PREFIX**

Please use TC\_PAIRS\_BDECK\_TEMPLATE.

*Used by:* TcPairs

*Family:* [config]

*Default:* Varies

---

**[deprecated] BDECK\_TRACK\_DATA\_DIR**

Please use TC\_PAIRS\_BDECK\_INPUT\_DIR.

*Used by:* TcPairs

*Family:* [dir]

*Default:* Varies

---

**[deprecated] BEG\_TIME**

Please use INIT\_BEG or VALID\_BEG instead. Beginning time for analysis in YYYYMMDD format.

*Used by:* PB2NC, PointStat

*Family:* [config]

*Default:* Varies

---

**[deprecated] BMODEL**

Please use TC\_STAT\_BMODEL.

*Used by:* TcStat

*Family:* [config]

*Default:*

---

### 4.6.3 C

---

**CI\_METHOD**

The method for creating confidence intervals. Valid options are EMC, or NONE.

*Used by:* MakePlots

*Family:* [config]

*Default:*

---

Control the size of the circle marker in the cyclone plotter.

*Used by:* CyclonePlotter

*Family:* [config]

*Default:* 41

---

**CLOCK\_TIME**

Automatically set by METplus with the time that METplus was started. Setting this variable has no effect as it will be overwritten. Can be used for reference in metplus\_final.conf or used with other config variables.

CIRCLE\_MA

*Used by:* All

*Family:* [config]

*Default:* Set automatically to current clock time in %Y%m%d%H%M%S format

---

### CONFIG\_DIR

Directory containing config files relevant to MET tools.

*Used by:* EnsembleStat, GridStat, Mode, StatAnalysis

*Family:* [dir]

*Default:* Varies

---

### CONFIG\_FILE

Specific configuration file name to use for MET tools.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

### CONVERT

Path to the ImageMagick “convert” executable.

*Used by:* PB2NC, PointStat, SeriesByInit, SeriesByLead

*Family:* [exe]

*Default:* /path/to

---

### [deprecated] CONVERT\_EXE

Please use CONVERT.

*Used by:* PB2NC, PointStat, SeriesByInit, SeriesByLead

*Family:* [exe]

*Default:* /path/to

---

### COV\_THRESH

Specify the values of the COV\_THRESH column in the MET .stat file to use. This is optional in the METplus configuration file for running with LOOP\_ORDER = times.

*Used by:* StatAnalysis

*Family:* [config]

*Default:*

---

Control the size of the cross marker in the cyclone plotter.

*Used by:* CyclonePlotter

*Family:* [config]

*Default:* 51

---

**CUT**

Path to the Linux “cut” executable.

*Used by:* PB2NC, PointStat

*Family:* [exe]

*Default:* /path/to

---

**[deprecated] CUT\_EXE**

Please use CUT.

*Used by:* PB2NC, PointStat

*Family:* [exe]

*Default:* /path/to

---

**[deprecated] CYCLONE**

Please use TC\_PAIRS\_CYCLONE or TC\_STAT\_CYCLONE..

*Used by:* TcPairs, TcStat

*Family:* [config]

*Default:* Varies

---

**CYCLONE\_INIT\_DATE**

Initialization date for the cyclone forecasts in YYYYMMDD format.

*Used by:* CyclonePlotter

*Family:* [config]

*Default:* Varies

---

**CYCLONE\_INIT\_HR**

Initialization hour for the cyclone forecasts in HH format.

*Used by:* CyclonePlotter

*Family:* [config]

*Default:* Varies

---

**CYCLONE\_INPUT\_DIR**

Input directory for the cyclone plotter. This should be the output directory for the MET TC Pairs utility.

*Used by:* CyclonePlotter

*Family:* [dir]

*Default:* Varies

---

**CYCLONE\_MODEL**

Define the model being used for the tropical cyclone forecasts.

*Used by:* CyclonePlotter

*Family:* [config]

*Default:* Varies

---

#### **CYCLONE\_OUT\_DIR**

Specify the directory where the output from the cyclone plotter should go.

*Used by:* CyclonePlotter

*Family:* [dir]

*Default:* Varies

---

#### **CYCLONE\_PLOT\_TITLE**

Title string for the cyclone plotter.

*Used by:* CyclonePlotter

*Family:* [config]

*Default:* Varies

---

### **4.6.4 D**

---

#### **DEMO\_YR**

The demo year. This is an optional value used by the plot\_TCMPR.R script, (which is wrapped by TCM-PRPlotter). Please refer to Chapter 21 in the MET User's Guide for more details.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

#### **DEP\_VARS**

Corresponds to the optional flag -dep in the plot\_TCMPR.R script, which is wrapped by TCMPRPlotter. The value to this flag is a comma-separated list (no whitespace) of dependent variable columns to plot ( e.g. AMSLP-BMSLP, AMAX\_WIND-BMAX\_WIND, TK\_ERR). If this is undefined, then the default plot for TK\_ERR (track error) is generated. Note, if you want the track error plot generated, in addition to other plots, then you need to explicitly list this with the other variables. Please refer to Chapter 21 in the MET User's Guide for more details.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

**DESC**

A single value or list of values used in the `stat_analysis` data stratification. Specifies the values of the `DESC` column in the MET `.stat` file to use. This is optional in the METplus configuration file for running with `LOOP_ORDER = times`

*Used by:* StatAnalysis

*Family:* [config]

*Default:* Varies

---

[deprecated]

**DLAND\_FILE**

Please use `TC_PAIRS_DLAND_FILE`.

*Used by:* TcPairs

*Family:* [config]

*Default:* Varies

---

**DLAT**

The value that defines the resolution of the data (in decimal degrees).

*Used by:* ExtractTiles, SeriesByLead

*Family:* [config]

*Default:* 0.5

---

**DLON**

The longitude value, in degrees. Set to the value that defines the resolution of the data (in decimal degrees).

*Used by:* ExtractTiles, SeriesByLead

*Family:* [config]

*Default:* 0.5

---

**DO\_NOT\_RUN\_EXE**

True/False. If True, applications will not run and will only output command that would have been called.

*Used by:* All

*Family:* [config]

*Default:* False

---

**4.6.5 E****EGREP**

Path to the Linux “`egrep`” executable.

*Used by:* PB2NC, PointStat

*Family:* [exe]

*Default:* /path/to

---

#### [deprecated] EGREP\_EXE

Please use EGREP.

*Used by:* PB2NC, PointStat

*Family:* [exe]

*Default:* /path/to

---

#### [deprecated] END\_DATE

Please use INIT\_END or VALID\_END instead.

*Used by:* PB2NC, PointStat

*Family:* [config]

*Default:* Varies

---

#### [deprecated] END\_HOUR

Ending hour for analysis with format HH.

*Used by:* PB2NC, PointStat

*Family:* [config]

*Default:* Varies

---

#### [deprecated] END\_TIME

Ending date string for analysis with format YYYYMMDD.

*Used by:* PB2NC, PointStat

*Family:* [config]

*Default:* Varies

---

#### ENSEMBLE\_STAT\_CONFIG\_FILE

Specify the absolute path to the configuration file for the MET ensemble\_stat tool.

*Used by:* EnsembleStat

*Family:* [config]

*Default:*

---

#### ENSEMBLE\_STAT\_ENS\_THRESH

Threshold for the ratio of the number of valid ensemble fields to the total number of expected ensemble members. This value is passed into the ensemble\_stat config file to make sure the percentage of files that are valid meets the expectation.

*Used by:* EnsembleStat

*Family:* [config]

*Default:* 1.0

---

#### **ENSEMBLE\_STAT\_GRID\_VX**

Used to set the regrid dictionary item 'to\_grid' in the MET ensemble\_stat config file. See the MET User's Guide for more information.

*Used by:* EnsembleStat

*Family:* [config]

*Default:* FCST

---

#### **ENSEMBLE\_STAT\_MET\_OBS\_ERROR\_TABLE**

*Used by:* EnsembleStat

*Family:* [config]

*Default:*

---

#### **ENSEMBLE\_STAT\_N\_MEMBERS**

Expected number of ensemble members found. This should correspond to the number of items in FCST\_ENSEMBLE\_STAT. If this number differs from the number of files are found for a given run, then ensemble\_stat will not run for that time.

*Used by:* EnsembleStat

*Family:* [config]

*Default:*

---

#### **ENSEMBLE\_STAT\_ONCE\_PER\_FIELD**

If True, run ensemble\_stat separately for each field name/level combination specified in the configuration file. See 4.3.2 for more information on how fields are specified. If False, run ensemble\_stat once with all of the fields specified.

*Used by:* EnsembleStat

*Family:* [config]

*Default:* False

---

#### **ENSEMBLE\_STAT\_OUTPUT\_DIR**

Specify the output directory where files from the MET ensemble\_stat tool are written.

*Used by:* EnsembleStat

*Family:* [dir]

*Default:* Varies

---

#### **ENSEMBLE\_STAT\_OUTPUT\_TEMPLATE**

Sets the subdirectories below ENSEMBLE\_STAT\_OUTPUT\_DIR using a template to allow run time information. If LOOP\_BY = VALID, default value is valid time YYYYMMDDHHMM/ensemble\_stat. If LOOP\_BY = INIT, default value is init time YYYYMMDDHHMM/ensemble\_stat.

**Used by:** EnsembleStat

**Family:** [filename\_templates]

**Default:** Varies

### ENS\_VAR<n>\_LEVELS

Define the levels for the <n>th ensemble variable to be used in the analysis where <n> is an integer  $\geq 1$ . The value can be a single item or a comma separated list of items. You can define NetCDF levels, such as (0,\*,\*), but you will need to surround these values with quotation marks so that the commas in the item are not interpreted as an item delimiter. Some examples:

```
ENS_VAR1_LEVELS = A06, P500
ENS_VAR2_LEVELS = "(0,*,*)", "(1,*,*)"
```

There can be <n> number of these variables defined in configuration files, simply increment the “\_VAR1\_” string to match the total number of variables being used, e.g.:

```
ENS_VAR1_LEVELS
ENS_VAR2_LEVELS
...
ENS_VAR<n>_LEVELS
```

See [4.3.2](#) for more information.

**Used by:** EnsembleStat

**Family:** [config]

**Default:** Varies

### ENS\_VAR<n>\_NAME

Define the name for the <n>th ensemble variable to be used in the analysis where <n> is an integer  $\geq 1$ . There can be <n> number of these variables defined in configuration files, simply increment the “\_VAR1\_” string to match the total number of variables being used, e.g.:

```
ENS_VAR1_NAME
ENS_VAR2_NAME
...
ENS_VAR<n>_NAME
```

See [4.3.2](#) for more information.

**Used by:** EnsembleStat

**Family:** [config]

**Default:** Varies

### ENS\_VAR<n>\_OPTIONS

Define the options for the <n>th ensemble variable to be used in the analysis where <n> is an integer  $\geq 1$ .

1. These addition options will be applied to every name/level/threshold combination for VAR<n>. There can be <n> number of these variables defined in configuration files, simply increment the “\_VAR1\_” string to match the total number of variables being used, e.g.:

```
ENS_VAR1_OPTIONS
ENS_VAR2_OPTIONS
...
ENS_VAR<n>_OPTIONS
```

See 4.3.2 for more information.

**Used by:** EnsembleStat

**Family:** [config]

**Default:** Varies

### ENS\_VAR<n>\_THRESH

Define the threshold(s) for the <n>th ensemble variable to be used in the analysis where <n> is an integer >= 1. The value can be a single item or a comma separated list of items that must start with a comparison operator (>, >=, ==, !=, <, <=, gt, ge, eq, ne, lt, le). There can be <n> number of these variables defined in configuration files, simply increment the “\_VAR1\_” string to match the total number of variables being used, e.g.:

```
ENS_VAR1_THRESH
ENS_VAR2_THRESH
...
ENS_VAR<n>_THRESH
```

See 4.3.2 for more information.

**Used by:** EnsembleStat

**Family:** [config]

**Default:** Varies

### EVENT\_EQUALIZATION

If event equalization is to be used (True) or not (False). If set to True, if any of the listed models are missing data for a particular time, data for all models will be masked out for this time. If set to False, there are no changes to the data.

**Used by:** MakePlots

**Family:** [config]

**Default:** True

[deprecated]

### EXTRACT\_OUT\_DIR

Please use EXTRACT\_TILES\_OUTPUT\_DIR. Set the output directory for the METplus extract\_tiles utility.

*Used by:* ExtractTiles, SeriesByInit, SeriesByLead

*Family:* [dir]

*Default:* Varies

---

#### **EXTRACT\_TILES\_FILTER\_OPTS**

Control what options are passed to the METplus extract\_tiles utility.

*Used by:* ExtractTiles

*Family:* [config]

*Default:* Varies

---

**EXTRACT\_T**

Set the output directory for the METplus extract\_tiles utility.

*Used by:* ExtractTiles, SeriesByInit, SeriesByLead

*Family:* [dir]

*Default:* Varies

#### **EXTRACT\_TILES\_VAR\_LIST**

Control what variables the METplus extract\_tiles utility runs on. Additional filtering by summary (via the MET tc\_stat tool). Please refer to Chapter 20 in the MET Users Guide (TC-STAT Tools) for all the available options for filtering by summary method in tc-stat. If no additional filtering is required, simply leave the value to EXTRACT\_TILES\_FILTER\_OPTS blank/empty in the METplus configuration file.

*Used by:* ExtractTiles

*Family:* [config]

*Default:* Varies

---

### **4.6.6 F**

#### **[deprecated] FCST\_EXACT\_VALID\_TIME**

No longer used. Please use FCST\_WINDOW\_BEGIN and FCST\_WINDOW\_END instead. If both of those variables are set to 0, the functionality is the same as FCST\_EXACT\_VALID\_TIME = True.

*Used by:* GridStat Mode, MTD

*Family:* [config]

*Default:* False

---

#### **[deprecated] FCST\_<n>\_FIELD\_NAME**

Please use FCST\_PCP\_COMBINE\_<n>\_FIELD\_NAME where N >=1 instead.

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### **FCST\_ASCII\_REGEX\_LEAD**

Regular expression used to find the forecast file (ASCII format) generated as an intermediate step in the series by lead use case.

*Used by:* SeriesByLead

*Family:* [regex\_pattern]

*Default:* Varies

---

#### **FCST\_ENSEMBLE\_STAT\_FILE\_WINDOW\_BEGIN**

See OBS\_ENSEMBLE\_STAT\_FILE\_WINDOW\_BEGIN [4.6.15](#).

*Used by:* EnsembleStat

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_BEGIN

---

#### **FCST\_ENSEMBLE\_STAT\_FILE\_WINDOW\_END**

See OBS\_ENSEMBLE\_STAT\_FILE\_WINDOW\_END [4.6.15](#).

*Used by:* EnsembleStat

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_END

---

#### **FCST\_ENSEMBLE\_STAT\_INPUT\_DIR**

Input directory for forecast files to use with the MET tool ensemble\_stat. A corresponding variable exists for observation data called OBS\_ENSEMBLE\_STAT\_INPUT\_DIR.

*Used by:* EnsembleStat

*Family:* [dir]

*Default:* Varies

---

#### **FCST\_ENSEMBLE\_STAT\_INPUT\_TEMPLATE**

Template used to specify forecast input filenames for the MET tool ensemble\_stat. A corresponding variable exists for observation data called OBS\_ENSEMBLE\_STAT\_INPUT\_TEMPLATE.

*Used by:* EnsembleStat

*Family:* [filename\_templates]

*Default:* Varies

---

#### **FCST\_FILE\_WINDOW\_BEGIN**

See OBS\_FILE\_WINDOW\_BEGIN [4.6.15](#).

*Used by:* EnsembleStat, GridStat, Mode, MTD, PB2NC, PointStat

*Family:* [config]

*Default:* Varies

---

#### **FCST\_FILE\_WINDOW\_END**

See OBS\_FILE\_WINDOW\_END [4.6.15](#).

*Used by:* EnsembleStat, GridStat, Mode, MTD, PB2NC, PointStat

*Family:* [config]

*Default:* Varies

---

#### **[deprecated] FCST\_GEMPAK\_INPUT\_DIR**

Input directory for GEMPAK formatted forecast files. Use GEMPAKTOCF\_INPUT\_DIR if GempakToCF is in the PROCESS\_LIST.

*Used by:* GempakToCF

*Family:* [dir]

*Default:* Varies

---

#### **[deprecated] FCST\_GEMPAK\_TEMPLATE**

Template used to specify input filenames for GEMPAK formatted forecast files. Use GEMPAKTOCF\_INPUT\_TEMPLATE if GempakToCF is in the PROCESS\_LIST.

*Used by:* GempakToCF

*Family:* [filename\_templates]

*Default:* Varies

---

#### **FCST\_GRID\_STAT\_FILE\_WINDOW\_BEGIN**

See OBS\_GRID\_STAT\_FILE\_WINDOW\_BEGIN [4.6.15](#).

*Used by:* GridStat

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_BEGIN

---

#### **FCST\_GRID\_STAT\_FILE\_WINDOW\_END**

See OBS\_GRID\_STAT\_FILE\_WINDOW\_END [4.6.15](#).

*Used by:* GridStat

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_END

---

#### **FCST\_GRID\_STAT\_INPUT\_DATATYPE**

Specify the data type of the input directory for forecast files used with the MET grid\_stat tool. Currently valid options are NETCDF, GRIB, and GEMPAK. If set to GEMPAK, data will automatically be converted to NetCDF via GempakToCF. A corresponding variable exists for observation data called OBS\_GRID\_STAT\_INPUT\_DATATYPE.

*Used by:* GridStat

*Family:* [config]

*Default:* Varies

---

#### **FCST\_GRID\_STAT\_INPUT\_DIR**

Input directory for forecast files to use with the MET tool `grid_stat`. A corresponding variable exists for observation data called `OBS_GRID_STAT_INPUT_DIR`.

*Used by:* GridStat

*Family:* [dir]

*Default:* Varies

---

#### **FCST\_GRID\_STAT\_INPUT\_TEMPLATE**

Template used to specify forecast input filenames for the MET tool `grid_stat`. A corresponding variable exists for observation data called `OBS_GRID_STAT_INPUT_TEMPLATE`.

*Used by:* GridStat

*Family:* [filename\_templates]

*Default:* Varies

---

#### **FCST\_GRID\_STAT\_PROB\_THRESH**

Threshold values to be used for probabilistic data in `grid_stat`. The value can be a single item or a comma separated list of items that must start with a comparison operator (>, >=, ==, !=, <, <=, gt, ge, eq, ne, lt, le). A corresponding variable exists for observation data called `OBS_GRID_STAT_PROB_THRESH`.

*Used by:* GridStat

*Family:* [config]

*Default:* ==0.1

---

#### **[deprecated] FCST\_HR\_END**

Please use `LEAD_SEQ` instead.

*Used by:*

*Family:* [config]

*Default:*

---

#### **[deprecated] FCST\_HR\_INTERVAL**

Please use `LEAD_SEQ` instead.

*Used by:*

*Family:* [config]

*Default:*

---

#### **[deprecated] FCST\_HR\_START**

Please use `LEAD_SEQ` instead.

*Used by:*

*Family:* [config]

*Default:*

---

**[deprecated] FCST\_INIT\_INTERVAL**

Specify the stride for forecast initializations.

*Used by:* EnsembleStat, GridStat, Mode

*Family:* [config]

*Default:* Varies

---

**[deprecated] FCST\_INPUT\_DIR\_REGEX**

Please use FCST\_POINT\_STAT\_INPUT\_DIR instead.

*Used by:* PointStat

*Family:* [regex\_pattern]

*Default:* Varies

---

**[deprecated] FCST\_INPUT\_DIR**

Specify the input directory for the forecast files. Use FCST\_[MET-APP]\_INPUT\_DIR instead, i.e. FCST\_GRID\_STAT\_IN

*Used by:* GridStat, Mode, PointStat, PcpCombine

*Family:* [dir]

*Default:* Varies

---

**[deprecated] FCST\_INPUT\_FILE\_REGEX**

Regular expression to use when identifying which forecast file to use.

*Used by:* PointStat

*Family:* [regex\_pattern]

*Default:* Varies

---

**[deprecated] FCST\_INPUT\_FILE\_TMPL**

Please use FCST\_POINT\_STAT\_INPUT\_TEMPLATE instead.

*Used by:* PointStat

*Family:* [filename\_templates]

*Default:* Varies

---

**[deprecated] FCST\_IS\_DAILY\_FILE**

Please use FCST\_PCP\_COMBINE\_IS\_DAILY\_FILE instead.

Acceptable values: true/false

*Used by:* PcpCombine

*Family:* [config]

*Default:* False

---

### **FCST\_IS\_PROB**

Specify whether the forecast data are probabilistic or not.

Acceptable values: true/false

*Used by:* EnsembleStat, GridStat, Mode, MTD, PointStat

*Family:* [config]

*Default:* False

---

### **FCST\_LEAD**

Specify the values of the FCST\_LEAD column in the MET .stat file to use. This is optional in the METplus configuration file for running with LOOP\_ORDER = times

*Used by:* StatAnalysis

*Family:* [config]

*Default:* Varies

---

### **[deprecated] FCST\_LEVEL**

Please use FCST\_PCP\_COMBINE\_INPUT\_LEVEL instead.

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

### **[deprecated] FCST\_MAX\_FORECAST**

Please use LEAD\_SEQ\_MAX instead. Specify the maximum forecast lead time to use for the analysis.

*Used by:* EnsembleStat, GridStat, Mode

*Family:* [config]

*Default:* Varies

---

### **FCST\_MODE\_CONV\_RADIUS**

Comma separated list of convolution radius values used by mode for forecast fields. A corresponding variable exists for observation data called OBS\_MODE\_CONV\_RADIUS.

*Used by:* Mode

*Family:* [config]

*Default:*

---

### **FCST\_MODE\_CONV\_THRESH**

Comma separated list of convolution threshold values used by mode for forecast fields. A corresponding variable exists for observation data called OBS\_MODE\_CONV\_THRESH.

*Used by:* Mode

*Family:* [config]

*Default:*

---

#### **FCST\_MODE\_FILE\_WINDOW\_BEGIN**

See OBS\_MODE\_FILE\_WINDOW\_BEGIN [4.6.15](#).

*Used by:* Mode

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_BEGIN

---

#### **FCST\_MODE\_FILE\_WINDOW\_END**

See OBS\_MODE\_FILE\_WINDOW\_END [4.6.15](#).

*Used by:* Mode

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_END

---

#### **FCST\_MODE\_MERGE\_FLAG**

Sets the merge\_flag value in the mode config file for forecast fields. Valid values are NONE, THRESH, ENGINE, and BOTH. A corresponding variable exists for observation data called OBS\_MODE\_MERGE\_FLAG.

*Used by:* Mode

*Family:* [config]

*Default:*

---

#### **FCST\_MODE\_MERGE\_THRESH**

Comma separated list of merge threshold values used by mode for forecast fields. A corresponding variable exists for observation data called OBS\_MODE\_MERGE\_THRESH.

*Used by:* Mode

*Family:* [config]

*Default:*

---

#### **FCST\_MODE\_INPUT\_DATATYPE**

Specify the data type of the input directory for forecast files used with the MET mode tool. Currently valid options are NETCDF, GRIB, and GEMPAK. If set to GEMPAK, data will automatically be converted to NetCDF via GempakToCF. A corresponding variable exists for observation data called OBS\_MODE\_INPUT\_DATATYPE.

*Used by:* Mode

*Family:* [config]

*Default:* Varies

---

#### **FCST\_MODE\_INPUT\_DIR**

Input directory for forecast files to use with the MET tool mode. A corresponding variable exists for observation data called OBS\_MODE\_INPUT\_DIR.

*Used by:* Mode

*Family:* [dir]

*Default:* Varies

---

#### FCST\_MODE\_INPUT\_TEMPLATE

Template used to specify forecast input filenames for the MET tool mode. A corresponding variable exists for observation data called OBS\_MODE\_INPUT\_TEMPLATE.

*Used by:* Mode

*Family:* [filename\_templates]

*Default:* Varies

---

#### FCST\_MTD\_CONV\_RADIUS

Comma separated list of convolution radius values used by mode-TD for forecast files. A corresponding variable exists for observation data called OBS\_MTD\_CONV\_RADIUS.

*Used by:* MTD

*Family:* [config]

*Default:*

---

#### FCST\_MTD\_CONV\_THRESH

Comma separated list of convolution threshold values used by mode-TD for forecast files. A corresponding variable exists for observation data called OBS\_MTD\_CONV\_THRESH.

*Used by:* MTD

*Family:* [config]

*Default:*

---

#### FCST\_MTD\_FILE\_WINDOW\_BEGIN

See OBS\_MTD\_FILE\_WINDOW\_BEGIN [4.6.15](#).

*Used by:* MTD

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_BEGIN

---

#### FCST\_MTD\_FILE\_WINDOW\_END

See OBS\_MTD\_FILE\_WINDOW\_END [4.6.15](#).

*Used by:* MTD

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_END

---

**FCST\_MTD\_INPUT\_DATATYPE**

Specify the data type of the input directory for forecast files used with the MET mode-TD tool. Currently valid options are NETCDF, GRIB, and GEMPAK. If set to GEMPAK, data will automatically be converted to NetCDF via GempakToCF. A corresponding variable exists for observation data called OBS\_MTD\_INPUT\_DATATYPE.

*Used by:* MTD

*Family:* [config]

*Default:* Varies

---

**FCST\_MTD\_INPUT\_DIR**

Input directory for forecast files to use with the MET tool mode-TD. A corresponding variable exists for observation data called OBS\_MTD\_INPUT\_DIR.

*Used by:* MTD

*Family:* [dir]

*Default:* Varies

---

**FCST\_MTD\_INPUT\_TEMPLATE**

Template used to specify forecast input filenames for the MET tool mode-TD. A corresponding variable exists for observation data called OBS\_MTD\_INPUT\_TEMPLATE.

*Used by:* MTD

*Family:* [filename\_templates]

*Default:* Varies

---

**[deprecated] FCST\_NATIVE\_DATA\_TYPE**

Specify the data format of the forecast data. Use FCST\_PCP\_COMBINE\_INPUT\_DATATYPE instead

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

**FCST\_NC\_TILE\_REGEX**

Define the regular expression for input forecast files that are in netCDF.

*Used by:* SeriesByLead, SeriesByInit

*Family:* [regex\_pattern]

*Default:* Varies

---

**FCST\_PCP\_COMBINE\_<n>\_FIELD\_NAME**

This variable is used to define a <n> hour accumulation NetCDF field in the forecast dataset used in the MET tool pcp\_combine. <n> must be an integer >= 1. A corresponding variable exists for observation data called OBS\_PCP\_COMBINE\_<n>\_FIELD\_NAME.

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### **FCST\_PCP\_COMBINE\_DATA\_INTERVAL**

Specify the accumulation interval of the forecast dataset used by the MET pcp\_combine tool when processing daily input files. A corresponding variable exists for observation data called OBS\_PCP\_COMBINE\_DATA\_INTERVAL.

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### **FCST\_PCP\_COMBINE\_DERIVE\_LOOKBACK**

Specify how far to look back in time in hours to find files for running the MET pcp\_combine tool in derive mode. A corresponding variable exists for observation data called OBS\_PCP\_COMBINE\_DERIVE\_LOOKBACK.

*Used by:* PcpCombine

*Family:* [config]

*Default:* ADD

---

#### **FCST\_PCP\_COMBINE\_INPUT\_DATATYPE**

Specify the data type of the input directory for forecast files used with the MET pcp\_combine tool. Currently valid options are NETCDF, GRIB, and GEMPAK. Required by pcp\_combine if FCST\_PCP\_COMBINE\_RUN is True. Replaces deprecated variable FCST\_NATIVE\_DATA\_TYPE. A corresponding variable exists for observation data called OBS\_PCP\_COMBINE\_INPUT\_DATA\_TYPE.

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### **FCST\_PCP\_COMBINE\_INPUT\_DIR**

Specify the input directory for forecast files used with the MET pcp\_combine tool. A corresponding variable exists for observation data called OBS\_PCP\_COMBINE\_INPUT\_DIR.

*Used by:* PcpCombine

*Family:* [dir]

*Default:* Varies

---

#### **FCST\_PCP\_COMBINE\_INPUT\_LEVEL**

Specify what accumulation level should be used from the forecast data for the analysis. Used only when running pcp\_combine with SUBTRACT mode set or processing accumulation files that do not have the accumulation specified in the filename template. A corresponding variable exists for observation data called OBS\_PCP\_COMBINE\_INPUT\_LEVEL.

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### **FCST\_PCP\_COMBINE\_INPUT\_TEMPLATE**

Template used to specify input filenames for forecast files used by the MET `pcp_combine` tool. A corresponding variable exists for observation data called `OBS_PCP_COMBINE_INPUT_TEMPLATE`.

*Used by:* PcpCombine

*Family:* [filename\_templates]

*Default:* Varies

---

#### **FCST\_PCP\_COMBINE\_IS\_DAILY\_FILE**

Specify whether the forecast file is a daily file or not. A corresponding variable exists for observation data called `OBS_PCP_COMBINE_IS_DAILY_FILE`.

Acceptable values: true/false

*Used by:* PcpCombine

*Family:* [config]

*Default:* False

---

#### **FCST\_PCP\_COMBINE\_METHOD**

Specify the method to be used with the MET `pcp_combine` tool processing forecast data.

Valid options are ADD, SUM, SUBTRACT, and DERIVE. A corresponding variable exists for observation data called `OBS_PCP_COMBINE_METHOD`.

*Used by:* PcpCombine

*Family:* [config]

*Default:* None

---

#### **FCST\_PCP\_COMBINE\_MIN\_FORECAST**

Specify the minimum forecast lead time to use when finding the lowest forecast lead to use in `pcp_combine`. A corresponding variable exists for observation data called `OBS_PCP_COMBINE_MIN_FORECAST`.

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### **FCST\_PCP\_COMBINE\_MAX\_FORECAST**

Specify the maximum forecast lead time to use when finding the lowest forecast lead to use in `pcp_combine`. A corresponding variable exists for observation data called `OBS_PCP_COMBINE_MAX_FORECAST`.

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### **FCST\_PCP\_COMBINE\_OUTPUT\_DIR**

Specify the output directory for forecast files generated by the MET `pcp_combine` tool. A corresponding variable exists for observation data called `OBS_PCP_COMBINE_OUTPUT_DIR`.

*Used by:* PcpCombine

*Family:* [dir]

*Default:* Varies

---

#### **FCST\_PCP\_COMBINE\_OUTPUT\_TEMPLATE**

Template used to specify output filenames for forecast files generated by the MET `pcp_combine` tool. A corresponding variable exists for observation data called `OBS_PCP_COMBINE_OUTPUT_TEMPLATE`.

*Used by:* PcpCombine

*Family:* [filename\_templates]

*Default:* Varies

---

#### **FCST\_PCP\_COMBINE\_RUN**

Specify whether to run the MET `pcp_combine` tool on forecast data or not. A corresponding variable exists for observation data called `OBS_PCP_COMBINE_RUN`.

Acceptable values: true/false

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### **FCST\_PCP\_COMBINE\_STAT\_LIST**

List of statistics to process when using the MET `pcp_combine` tool on forecast data in derive mode. A corresponding variable exists for observation data called `OBS_PCP_COMBINE_STAT_LIST`.

Acceptable values: sum, min, max, range, mean, stdev, vld\_count

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### **FCST\_PCP\_COMBINE\_TIMES\_PER\_FILE**

Specify the number of accumulation intervals of the forecast dataset used by the MET `pcp_combine` tool when processing daily input files. A corresponding variable exists for observation data called `OBS_PCP_COMBINE_TIMES`.

*Used by:* PcpCombine

*Family:* [config]

*Default:* ~~Varies~~

---

#### **FCST\_POINT\_STAT\_FILE\_WINDOW\_BEGIN**

See `OBS_POINT_STAT_FILE_WINDOW_BEGIN` [4.6.15](#).

*Used by:* PointStat

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_BEGIN

---

#### FCST\_POINT\_STAT\_FILE\_WINDOW\_END

See OBS\_POINT\_STAT\_FILE\_WINDOW\_END [4.6.15](#).

*Used by:* PointStat

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_END

---

#### FCST\_POINT\_STAT\_INPUT\_DATATYPE

Specify the data type of the input directory for forecast files used with the MET point\_stat tool. Currently valid options are NETCDF, GRIB, and GEMPAK. If set to GEMPAK, data will automatically be converted to NetCDF via GempakToCF. A corresponding variable exists for observation data called OBS\_POINT\_STAT\_INPUT\_DATATYPE.

*Used by:* PointStat

*Family:* [config]

*Default:* Varies

---

#### FCST\_POINT\_STAT\_INPUT\_DIR

Input directory for forecast files to use with the MET tool point\_stat. A corresponding variable exists for observation data called OBS\_POINT\_STAT\_INPUT\_DIR.

*Used by:* PointStat

*Family:* [dir]

*Default:* Varies

---

#### FCST\_POINT\_STAT\_INPUT\_TEMPLATE

Template used to specify forecast input filenames for the MET tool point\_stat. A corresponding variable exists for observation data called OBS\_POINT\_STAT\_INPUT\_TEMPLATE.

*Used by:* GriPointStat

*Family:* [filename\_templates]

*Default:* Varies

---

#### FCST\_REGRID\_DATA\_PLANE\_INPUT\_DATATYPE

Specify the data type of the input directory for forecast files used with the MET regrid\_data\_plane tool. Currently valid options are NETCDF, GRIB, and GEMPAK. Required by pcp\_combine. A corresponding variable exists for observation data called OBS\_REGRID\_DATA\_PLANE\_INPUT\_DATATYPE.

*Used by:* RegridDataPlane

*Family:* [config]

*Default:* Varies

---

**FCST\_REGRID\_DATA\_PLANE\_INPUT\_DIR**

Specify the input directory for forecast files used with the MET regrid\_data\_plane tool. A corresponding variable exists for observation data called OBS\_REGRID\_DATA\_PLANE\_INPUT\_DIR.

*Used by:* RegridDataPlane

*Family:* [dir]

*Default:* Varies

---

**FCST\_REGRID\_DATA\_PLANE\_INPUT\_TEMPLATE**

Template used to specify input filenames for forecast data used by the MET regrid\_data\_plane tool. If not set, METplus will use FCST\_REGRID\_DATA\_PLANE\_TEMPLATE. A corresponding variable exists for observation data called OBS\_REGRID\_DATA\_PLANE\_INPUT\_TEMPLATE.

*Used by:* RegridDataPlane

*Family:* [filename\_templates]

*Default:* Varies

---

**FCST\_REGRID\_DATA\_PLANE\_OUTPUT\_TEMPLATE**

Template used to specify output filenames for forecast data used by the MET regrid\_data\_plane tool. If not set, METplus will use FCST\_REGRID\_DATA\_PLANE\_TEMPLATE. A corresponding variable exists for observation data called OBS\_REGRID\_DATA\_PLANE\_OUTPUT\_TEMPLATE.

*Used by:* RegridDataPlane

*Family:* [filename\_templates]

*Default:* Varies

---

**FCST\_REGRID\_DATA\_PLANE\_TEMPLATE**

Template used to specify filenames for forecast data used by the MET regrid\_data\_plane tool. To specify different templates for input and output files, use FCST\_REGRID\_DATA\_PLANE\_INPUT\_TEMPLATE and FCST\_REGRID\_DATA\_PLANE\_OUTPUT\_TEMPLATE. A corresponding variable exists for observation data called OBS\_REGRID\_DATA\_PLANE\_TEMPLATE.

*Used by:* RegridDataPlane

*Family:* [filename\_templates]

*Default:* Varies

---

**FCST\_REGRID\_DATA\_PLANE\_OUTPUT\_DIR**

Specify the output directory for forecast files used with the MET regrid\_data\_plane tool. A corresponding variable exists for observation data called OBS\_REGRID\_DATA\_PLANE\_OUTPUT\_DIR.

*Used by:* RegridDataPlane

*Family:* [dir]

*Default:* Varies

---

**FCST\_THRESH**

Specify the values of the FCST\_THRESH column in the MET .stat file to use. This is optional in the METplus configuration file for running with LOOP\_ORDER = times.

*Used by:* StatAnalysis

*Family:* [config]

*Default:*

---

**FCST\_TILE\_PREFIX**

Prefix for forecast tile files. Used to create filename of intermediate files that are created while performing a series analysis.

*Used by:* ExtractTiles, SeriesByLead

*Family:* [regex\_pattern]

*Default:* Varies

---

**FCST\_TILE\_REGEX**

Regular expression for forecast input files that are in GRIB2.

*Used by:* SeriesByInit, SeriesByLead

*Family:* [regex\_pattern]

*Default:* Varies

---

**[deprecated] FCST\_VAR**

Define the name of the forecast variable to be used in the analysis. See FCST\_VAR<n>\_NAME, FCST\_VAR<n>\_LEVELS, FCST\_VAR<n>\_THRESH, and FCST\_VAR<n>\_OPTIONS where <n> = integer >= 1.

*Used by:* EnsembleStat, MakePlots

*Family:* [config]

*Default:* Varies

---

**FCST\_VAR\_LEVEL**

Specify the values of the FCST\_VAR\_LEVEL column in the MET .stat file to use.

*Used by:* StatAnalysis

*Family:* [config]

*Default:* Varies

---

**FCST\_VAR\_NAME**

Specify the values of the FCST\_VAR\_NAME column in the MET .stat file to use. This is optional in the METplus configuration file for running with LOOP\_ORDER = times

*Used by:* StatAnalysis

*Family:* [config]

*Default:* Varies

---

**FCST\_VAR<n>\_LEVELS**

Define the levels for the <n>th forecast variable to be used in the analysis where <n> is an integer  $\geq 1$ . The value can be a single item or a comma separated list of items. You can define NetCDF levels, such as (0,\*,\*), but you will need to surround these values with quotation marks so that the commas in the item are not interpreted as an item delimiter. Some examples:

```
FCST_VAR1_LEVELS = A06, P500
FCST_VAR2_LEVELS = "(0,*,*)", "(1,*,*)"
```

If FCST\_VAR<n>\_LEVELS is not set but OBS\_VAR<n>\_LEVELS is, the same information will be used for both variables. There can be <n> number of these variables defined in configuration files, simply increment the “\_VAR1\_” string to match the total number of variables being used, e.g.:

```
FCST_VAR1_LEVELS
FCST_VAR2_LEVELS
...
FCST_VAR<n>_LEVELS
```

See [4.3.2](#) for more information.

**Used by:** GridStat, EnsembleStat, PointStat, Mode, MTD, PcpCombine

**Family:** [config]

**Default:** Varies

---

**FCST\_VAR<n>\_NAME**

Define the name for the <n>th forecast variable to be used in the analysis where <n> is an integer  $\geq 1$ . If FCST\_VAR<n>\_NAME is not set but OBS\_VAR<n>\_NAME is, the same information will be used for both variables. There can be <n> number of these variables defined in configuration files, simply increment the “\_VAR1\_” string to match the total number of variables being used, e.g.:

```
FCST_VAR1_NAME
FCST_VAR2_NAME
...
FCST_VAR<n>_NAME
```

See [4.3.2](#) for more information.

**Used by:** GridStat, EnsembleStat, PointStat, Mode, MTD, PcpCombine

**Family:** [config]

**Default:** Varies

---

**FCST\_VAR<n>\_OPTIONS**

Define the options for the <n>th forecast variable to be used in the analysis where <n> is an integer  $\geq 1$ . These addition options will be applied to every name/level/threshold combination for VAR<n>. If

FCST\_VAR<n>\_OPTIONS is not set but OBS\_VAR<n>\_OPTIONS is, the same information will be used for both variables. There can be <n> number of these variables defined in configuration files, simply increment the “\_VAR1\_” string to match the total number of variables being used, e.g.:

```
FCST_VAR1_OPTIONS
FCST_VAR2_OPTIONS
...
FCST_VAR<n>_OPTIONS
```

See [4.3.2](#) for more information.

**Used by:** GridStat, EnsembleStat, PointStat, Mode, MTD, PcpCombine

**Family:** [config]

**Default:** Varies

---

### FCST\_VAR<n>\_THRESH

Define the threshold(s) for the <n>th forecast variable to be used in the analysis where <n> is an integer  $\geq 1$ . The value can be a single item or a comma separated list of items that must start with a comparison operator (>,>=,==,!=,<,<=,gt,ge,eq,ne,lt,le). If FCST\_VAR<n>\_THRESH is not set but OBS\_VAR<n>\_THRESH is, the same information will be used for both variables. There can be <n> number of these variables defined in configuration files, simply increment the “\_VAR1\_” string to match the total number of variables being used, e.g.:

```
FCST_VAR1_THRESH
FCST_VAR2_THRESH
...
FCST_VAR<n>_THRESH
```

See [4.3.2](#) for more information.

**Used by:** GridStat, EnsembleStat, PointStat, Mode, MTD, PcpCombine

**Family:** [config]

**Default:** Varies

---

### FCST\_WINDOW\_BEGIN

See OBS\_WINDOW\_BEGIN [4.6.15](#).

**Used by:** EnsembleStat, GridStat, Mode, MTD, PB2NC, PointStat

**Family:** [config]

**Default:** Varies

---

### FCST\_WINDOW\_END

See OBS\_WINDOW\_END [4.6.15](#).

**Used by:** EnsembleStat, GridStat, Mode, MTD, PB2NC, PointStat

**Family:** [config]

*Default:* Varies

---

### **FHR\_BEG**

Specify the first forecast lead time to use in the analysis. Use in combination with FHR\_END and FHR\_INC.

*Used by:* SeriesByLead

*Family:* [config]

*Default:* Varies

---

### **FHR\_END**

Specify the last forecast lead time to use in the analysis. Use in combination with FHR\_BEG and FHR\_INC.

*Used by:* SeriesByLead

*Family:* [config]

*Default:* Varies

---

### **FHR\_GROUP\_BEG**

Define which forecast lead time should be first in a group of forecast leads to use in the analysis. Use in combination with FHR\_GROUP\_END and FHR\_INC.

Example:

FHR\_GROUP\_BEG = 24

FHR\_GROUP\_END = 42

FHR\_INC = 6

List of forecast leads processed: [24, 30, 36, 42]

*Used by:* SeriesByLead

*Family:* [config]

*Default:* Varies

---

### **FHR\_GROUP\_END**

Define which forecast lead time should be the last in a group of forecast leads to use in the analysis. Use in combination with FHR\_GROUP\_BEG and FHR\_INC.

Example:

FHR\_GROUP\_BEG = 24

FHR\_GROUP\_END = 42

FHR\_INC = 6

List of forecast leads processed: [24, 30, 36, 42]

*Used by:* SeriesByLead

*Family:* [config]

*Default:* Varies

---

**FHR\_GROUP\_LABELS**

Label strings to use for the forecast groups.

*Used by:* SeriesByLead

*Family:* [config]

*Default:* Varies

---

**FHR\_INC**

Stride to use for incrementing forecast lead times used in the analysis. Use in combination with FHR\_BEG and FHR\_END or FHR\_GROUP\_BEG and FHR\_GROUP\_END.

*Used by:* SeriesByLead

*Family:* [config]

*Default:* Varies

---

**FILTER**

Corresponds to the optional -filter argument to the plot\_TCMPR.R script which is wrapped by TCM-PRPlotter. This is a list of filtering options for the tc\_stat tool.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

**FILTERED\_TCST\_DATA\_FILE**

Corresponds to the optional -tcst argument to the plot\_TCMPR.R script which is wrapped by TCM-PRPlotter. This is a tcst data file to be used instead of running the tc\_stat tool. Indicate a full path to the data file.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

**FOOTNOTE\_FLAG**

This corresponds to the optional -footnote flag in the plot\_TCMPR.R script which is wrapped by TCM-PRPlotter. According to the plot\_TCMPR.R usage, this flag is used to disable footnote (date).

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

**[deprecated] FORECAST\_TMPL**

Please use TC\_PAIRS\_ADECK\_TEMPLATE.

*Used by:* TcPairs

*Family:* [filename\_templates]

*Default:* Varies

---

#### 4.6.7 G

---

##### **GEMPAKTOCF\_CLASSPATH**

Path to the GempakToCF binary file and the NetCDF jar file required to run GempakToCF.

*Used by:* GempakToCF

*Family:* [exe]

*Default:* Varies

---

##### **GEMPAKTOCF\_INPUT\_DIR**

Specify the input directory for the tool used to convert GEMPAK files to netCDF.

*Used by:* GempakToCF

*Family:* [dir]

*Default:* Varies

---

##### **GEMPAKTOCF\_INPUT\_TEMPLATE**

Filename template used for input files to the tool used to convert GEMPAK files to netCDF.

*Used by:* GempakToCF

*Family:* [filename\_templates]

*Default:* Varies

---

##### **GEMPAKTOCF\_OUTPUT\_DIR**

Specify the output directory for files generated by the tool used to convert GEMPAK files to netCDF.

*Used by:* GempakToCF

*Family:* [dir]

*Default:* Varies

---

##### **GEMPAKTOCF\_OUTPUT\_TEMPLATE**

Filename template used for output files from the tool used to convert GEMPAK files to netCDF.

*Used by:* GempakToCF

*Family:* [filename\_templates]

*Default:* Varies

---

##### **GEMPAKTOCF\_SKIP\_IF\_OUTPUT\_EXISTS**

If True, do not run GempakToCF if output file already exists. Set to False to overwrite files.

*Used by:* GempakToCF

*Family:* [config]

*Default:* Varies

---

### **GENERATE\_TRACK\_ASCII**

Specify whether or not to produce an ASCII file containing all of the tracks in the plot.

Acceptable values: true/false

*Used by:* CyclonePlotter

*Family:* [conf]

*Default:* Varies

---

### **[deprecated] GEN\_SEQ**

*Used by:*

*Family:*

*Default:*

---

### **GFS\_ANLY\_FILE\_TMPL**

Filename template used to identify the GFS analysis file.

*Used by:* ExtractTiles, SeriesByLead

*Family:* [filename\_templates]

*Default:* Varies

---

### **GFS\_FCST\_FILE\_TMPL**

Filename templated used to identify the GFS forecast files.

*Used by:* ExtractTiles, SeriesByLead

*Family:* [filename\_templates]

*Default:* Varies

---

### **[deprecated] GRID\_STAT\_CONFIG**

Please use GRID\_STAT\_CONFIG\_FILE instead. Specify the absolute path to the configuration file used by the MET grid\_stat tool.

*Used by:* GridStat

*Family:* [config]

*Default:* Varies

---

### **GRID\_STAT\_CONFIG\_FILE**

Specify the absolute path to the configuration file used by the MET grid\_stat tool.

*Used by:* GridStat

*Family:* [config]

*Default:* Varies

---

#### **GRID\_STAT\_ONCE\_PER\_FIELD**

True/False. If True, grid\_stat will run once to process all name/level/threshold combinations specified. If False, it will run once for each name/level. Some cases require this to be set to False, for example processing probabilistic forecasts or precipitation accumulations.

*Used by:* GridStat

*Family:* [config]

*Default:* False

---

#### **[deprecated] GRID\_STAT\_OUT\_DIR**

Specify the output directory where files from the MET grid\_stat tool are written. Please use GRID\_STAT\_OUTPUT\_DIR instead.

*Used by:* GridStat

*Family:* [dir]

*Default:* Varies

---

#### **GRID\_STAT\_OUTPUT\_DIR**

Specify the output directory where files from the MET grid\_stat tool are written.

*Used by:* GridStat

*Family:* [dir]

*Default:* Varies

---

#### **GRID\_STAT\_OUTPUT\_TEMPLATE**

Sets the subdirectories below GRID\_STAT\_OUTPUT\_DIR using a template to allow run time information. If LOOP\_BY = VALID, default value is valid time YYYYMMDDHHMM/grid\_stat. If LOOP\_BY = INIT, default value is init time YYYYMMDDHHMM/grid\_stat.

*Used by:* GridStat

*Family:* [filename\_templates]

*Default:* Varies

---

#### **GRID\_STAT\_VERIFICATION\_MASK\_TEMPLATE**

Template used to specify the verification mask filename for the MET tool grid\_stat.

*Used by:* GridStat

*Family:* [filename\_templates]

*Default:* Varies

---

### 4.6.8 H

---

#### **HFIP\_BASELINE**

Corresponds to the optional `-hfip_bsln` flag in the `plot_TCMPR.R` script which is wrapped by `TCMPRPlotter`. This is a string that indicates whether to add the HFIP baseline, and indicates the version (no, 0, 5, 10 year goal).

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

### 4.6.9 I

---

#### **INIT\_BEG**

Specify the beginning initialization time to be used in the analysis. Format can be controlled by `INIT_TIME_FMT`.

See [4.3.1.3](#) for more information.

*Used by:* All

*Family:* [config]

*Default:* Varies

---

#### **INIT\_END**

Specify the ending initialization time to be used in the analysis. Format can be controlled by `INIT_TIME_FMT`.

See [4.3.1.3](#) for more information.

*Used by:* All

*Family:* [config]

*Default:* Varies

---

#### **INIT\_EXCLUDE**

Specify which, if any, forecast initializations to exclude from the analysis.

*Used by:* TcPairs, TcStat

*Family:* [config]

*Default:* Varies

---

#### **INIT\_HOUR\_BEG**

Specify the beginning initialization hour to be used in the analysis. Format is HHMM.

**Used by:** MakePlots, StatAnalysis

**Family:** [config]

**Default:** Varies

---

**INIT\_HOUR**

Specify the ending initialization hour to be used in the analysis. Format is HH or HHMM.

**Used by:** ExtractTiles, MakePlots, StatAnalysis, TcPairs, TcStat

**Family:** [config]

**Default:** Varies

---

### INIT\_HOUR\_INCREMENT

Specify a time increment for valid times for use in the analysis. This is an integer defined in seconds.

**Used by:** MakePlots, StatAnalysis

**Family:** [config]

**Default:** Varies

---

**INIT\_HOUR\_METHOD**

Specify the method for the treatment of valid hours. Valid options are LOOP or GROUP. LOOP will consider the initialization hours individually, and GROUP will consider them as a whole.

**Used by:** MakePlots, StatAnalysis

**Family:** [config]

**Default:** Varies

---

### INIT\_INCLUDE

Specify which forecast initializations to include in the analysis.

**Used by:** TcPairs, TcStat

**Family:** [config]

**Default:** Varies

---

### INIT\_INCREMENT

Control the increment or stride to use when stepping between forecast initializations. Units are seconds.

See [4.3.1.3](#) for more information.

**Used by:** All

**Family:** [config]

**Default:** Varies

---

### INIT\_SEQ

Specify a list of initialization hours that are used to build a sequence of forecast lead times to include in the analysis. Used only when looping by valid time (LOOP\_BY = VALID). Comma separated list format, e.g.:  
0, 6, 12

See [4.3.1.4](#) for more information.

**Used by:** EnsembleStat, GridStat, Mode, MTD, PB2NC, PcpCombine, PointStat, RegridDataPlane, SeriesByLead

**Family:** [config]

**Default:** Varies

---

#### INIT\_TIME\_FMT

Specify a formatting string to use for INIT\_BEG and INIT\_END.

See [4.3.1.3](#) for more information.

**Used by:** All

**Family:**

**Default:**

---

#### INTERP

Specify the interpolation used to create the MET .stat files. This is optional in the METplus configuration file for running with LOOP\_ORDER = times.

**Used by:** MakePlots, StatAnalysis

**Family:** [config]

**Default:**

---

#### INTERP\_PTS

Corresponds to the interpolation in the MET .stat files. This is optional in the METplus configuration file for running with LOOP\_ORDER = times.

**Used by:** MakePlots, StatAnalysis

**Family:** [config]

**Default:**

---

#### INTERVAL\_T

Define the interval time in hours (HH) to be used by the MET pb2nc tool.

**Used by:** PB2NC

**Family:** [config]

**Default:** Varies

---

### 4.6.10 J

---

#### JOB\_ARGS

Specify stat\_analysis job arguments to run. The job arguments that are to be run with the corresponding

JOB\_NAME. If using `-dump_row`, use `-dump_row [dump_row_filename]`. If using `-out_stat`, `-out_stat [out_stat_filename]`. For more information on these job arguments, please see the MET Users Guide.

*Used by:* StatAnalysis

*Family:* [config]

*Default:*

---

#### JOB\_NAME

Specify `stat_analysis` job name to run. Valid options are `filter`, `summary`, `aggregate`, `aggregate_stat`, `go_index`, and `ramp`. For more information on these job names and what they do, please see the MET Users Guide.

*Used by:* StatAnalysis

*Family:* [config]

*Default:*

---

#### 4.6.11 K

#### 4.6.12 L

---

#### LAT\_ADJ

Specify a latitude adjustment, in degrees to be used in the analysis. In the `ExtractTiles` wrapper, this corresponds to the 2m portion of the 2n x 2m subregion tile.

*Used by:* `ExtractTiles`, `SeriesByLead`

*Family:* [config]

*Default:* Varies

---

#### LEAD

For `CyclonePlotter`, this refers to the column of interest in the input ASCII cyclone file.

In the `TCMPRPlotter`, this corresponds to the optional `-lead` argument in the `plot_TCMPR.R` script (which is wrapped by `TCMPRPlotter`). This argument is set to a comma-separated list of lead times (h) to be plotted.

In `TcStat`, this corresponds to the name of the column of interest in the input ASCII data file.

*Used by:* `CyclonePlotter`, `TCMPRPlotter`, `TcStat`

*Family:* [config]

*Default:* Varies

---

#### LEAD\_LIST

Specify a list of forecast leads to include in the analysis. Comma separated list format, e.g.:

0, 24, 48, 72, 96, 120

**Used by:** MakePlots, StatAnalysis

**Family:** [config]

**Default:** Varies

---

### LEAD\_SEQ

Specify the sequence of forecast lead times to include in the analysis. Comma separated list format, e.g.:

0, 6, 12

See [4.3.1.4](#) for more information.

**Used by:** EnsembleStat, GridStat, Mode, MTD, PB2NC, PcpCombine, PointStat, RegridDataPlane, SeriesByLead

**Family:** [config]

**Default:** Varies

---

### LEAD\_SEQ\_MIN

Minimum forecast lead to be processed. Used primarily with INIT\_SEQ but also affects LEAD\_SEQ.

See [4.3.1.4](#) for more information.

**Used by:** EnsembleStat, GridStat, Mode, MTD, PB2NC, PcpCombine, PointStat, RegridDataPlane, SeriesByLead

**Family:** [config]

**Default:** Varies

---

### LEAD\_SEQ\_MAX

Maximum forecast lead to be processed. Used primarily with INIT\_SEQ but also affects LEAD\_SEQ.

See [4.3.1.4](#) for more information.

**Used by:** EnsembleStat, GridStat, Mode, MTD, PB2NC, PcpCombine, PointStat, RegridDataPlane, SeriesByLead

**Family:** [config]

**Default:** Varies

---

### LEAD\_SEQ\_<n>

Required when SERIES\_BY\_LEAD\_GROUP\_FCSTS=True. Not necessary otherwise. Specify the sequence of forecast lead times to include in the analysis. Comma separated list format, e.g.:

0, 6, 12. <n> corresponds to the bin in which the user wishes to aggregate series by lead results.

**Used by:** SeriesByLead

**Family:** [config]

**Default:** Varies

---

### LEAD\_SEQ\_<n>\_LABEL

Required when SERIES\_BY\_LEAD\_GROUP\_FCSTS=True. Specify the label of the corresponding bin of series by lead results.

**Used by:** SeriesByLead

**Family:** [config]

**Default:**

---

### LEGEND

The text to be included in the legend of your plot.

**Used by:** TCMPRPlotter

**Family:** [config]

**Default:** Varies

---

### LINE\_TYPE

Specify the MET STAT line types to be considered. This is optional in the METplus configuration file for running with LOOP\_ORDER = times.

**Used by:** TCMPRPlotter

**Family:** [config]

**Default:**

---

### LOG\_DIR

Specify the directory where log files from MET and METplus should be written.

**Used by:** All

**Family:** [dir]

**Default:** Varies

---

### LOG\_LEVEL

Specify the level of logging.

Everything above this level is sent to standard output. To quiet the output to a comfortable level, set this to "ERROR".

Options (ordered MOST verbose to LEAST verbose):

NOTSET

DEBUG

INFO

WARNING

ERROR

CRITICAL

*Used by:* All  
*Family:* [config]  
*Default:* Varies

---

### LOG\_METPLUS

Control the filename of the METplus log file. Control the timestamp appended to the filename with LOG\_TIMESTAMP\_TEMPLATE. To turn OFF all logging, do not set this option.

*Used by:* All  
*Family:* [config]  
*Default:* Varies

---

### LOG\_MET\_OUTPUT\_TO\_METPLUS

Control whether logging output from the MET tools is sent to the METplus log file, or individual log files for each MET tool.

*Used by:* All  
*Family:* [config]  
*Default:* yes/no

---

### LOG\_MET\_VERBOSITY

Control the verbosity of the logging from the MET tools.

0 = Least amount of logging (lowest verbosity)

5 = Most amount of logging (highest verbosity)

*Used by:* All  
*Family:* [config]  
*Default:* 2

---

### LOG\_TIMESTAMP\_TEMPLATE

Set the timestamp template for the METplus log file. Use Python strftime directives, e.g. %Y%m%d for YYYYMMDD.

*Used by:* All  
*Family:* [config]  
*Default:* %Y%m%d

---

### LOG\_TIMESTAMP\_USE\_DATETIME

True/False. Determines which time to use for the log filenames. If True, use INIT\_BEG if LOOP\_BY\_INIT is True or VALID\_BEG if LOOP\_BY\_INIT is False. If False, use current time.

*Used by:* All  
*Family:* [config]  
*Default:* False

---

**LON\_ADJ**

Specify a longitude adjustment, in degrees to be used in the analysis. In the ExtractTiles wrapper, this corresponds to the 2n portion of the 2n x 2m subregion tile.

*Used by:* ExtractTiles, SeriesByLead

*Family:* [config]

*Default:* Varies

---

**LOOP\_BY**

Control whether the analysis is processed across valid or initialization times.

See [4.3.1.1](#) for more information.

*Used by:* All

*Family:* [config]

*Default:* true

---

**LOOP\_ORDER**

Control the looping order for METplus. Valid options are “times” or “processes”. “times” runs all items in the PROCESS\_LIST for a single run time, then repeat until all times have been evaluated. “processes” runs each item in the PROCESS\_LIST for all times specified, then repeat for the next item in the PROCESS\_LIST.

*Used by:* All

*Family:* [config]

*Default:* Varies

---

**4.6.13 M**

---

**METPLUS\_BASE**

This variable will automatically be set by METplus when it is started. It will be set to the location of METplus that is currently being run. Setting this variable in a config file will have no effect and will report a warning that it is being overridden.

*Used by:* All

*Family:* [dir]

*Default:* Location METplus is being run from

---

**METPLUS\_CONF**

Provide the absolute path to the METplus final configuration file. This file will contain every configuration option and value used when METplus was run.

*Used by:* All  
*Family:* [config]  
*Default:* Varies

---

### **MET\_BASE**

The base directory where your MET installation resides.

*Used by:* CyclonePlotter, ExtractTiles, PB2NC, PointStat, SeriesByInit, SeriesByLead, TCMPRPlotter, TcPairs  
*Family:* [dir]  
*Default:*

---

### **[deprecated] MET\_BIN**

The location of MET binaries.

*Used by:*  
*Family:*  
*Default:*

---

### **MET\_BUILD\_BASE**

The base directory of the MET install. Only needed if using MET version 6.0

*Used by:* TCMPRPlotter  
*Family:* [dir]  
*Default:* Varies

---

### **MET\_INSTALL\_DIR**

The base directory of the MET install. To be defined when using MET version 6.1 and beyond. Used to get the full path of the MET executable when calling from METplus Wrappers.

*Used by:* All  
*Family:* [dir]  
*Default:* Varies

---

### **[deprecated] MISSING\_VAL**

Please use TC\_PAIRS\_MISSING\_VAL.

*Used by:* TcPairs  
*Family:* [config]  
*Default:* Varies

---

### **[deprecated] MISSING\_VAL\_TO\_REPLACE**

Please use TC\_PAIRS\_MISSING\_VAL\_TO\_REPLACE.

*Used by:* TcPairs  
*Family:* [config]

**Default:** Varies

---

## MODEL

Specify the model name. This is the model name listed in the MET .stat files.

**Used by:** EnsembleStat, GridStat, PointStat, PcpCombine, StatAnalysis, TcPairs

**Family:** [config]

**Default:** Varies

---

## MODEL1\_NAME

Define the model name for the first model to be used in the analysis. This is the model name listed in the MET .stat files. There can be N number of models defined in configuration files, simply increment the “MODEL1\_” string to match the total number of models being used, e.g.:

MODEL1\_NAME

MODEL2\_NAME

.

.

.

MODELN\_NAME

**Used by:** MakePlots, StatAnalysis

**Family:** [config]

**Default:** Varies

---

## MODEL1\_NAME\_ON\_PLOT

Define the name the first model will be listed as on the plots. There can be N number of models defined in configuration files, simply increment the “MODEL1\_” string to match the total number of models being used, e.g.:

MODEL1\_NAME\_ON\_PLOT

MODEL2\_NAME\_ON\_PLOT

.

.

.

MODELN\_NAME\_ON\_PLOT

**Used by:** MakePlots, StatAnalysis

**Family:** [config]

**Default:** Varies

---

## MODEL1\_OBS\_NAME

Define the observation name that was used to compare the first model to be. This is the observation name listed in the MET .stat files. There can be N number of observation names defined in configuration files, simply increment the “MODEL1\_” string to match the total number of models being used, e.g.:

MODEL1\_OBS\_NAME

MODEL2\_OBS\_NAME

.  
.  
.

MODELN\_OBS\_NAME

*Used by:* MakePlots, StatAnalysis

*Family:* [config]

*Default:* Varies

---

### MODEL1\_STAT\_DIR

Define the stat file directory for the first model to be used in the analysis. There can be N number of model directories defined in configuration files, simply increment the “MODEL1\_” string to match the total number of models being used, e.g.:

MODEL1\_DIR

MODEL2\_DIR

.  
.  
.

MODELN\_DIR

*Used by:* StatAnalysis

*Family:* [config]

*Default:* Varies

---

### MODEL\_DATA\_DIR

Specify the directory where the model data are located.

*Used by:* ExtractTiles, SeriesByLead

*Family:* [dir]

*Default:* Varies

---

### [deprecated] MODEL\_NAME

Please use MODEL instead.

*Used by:* PointStat

*Family:* [config]

*Default:* Varies

---

### [deprecated] MODE\_CONFIG

Please use MODE\_CONFIG\_FILE instead. Path to mode configuration file.

*Used by:* Mode

*Family:* [config]

*Default:* Varies

---

**MODE\_CONFIG\_FILE**

Path to mode configuration file.

*Used by:* Mode

*Family:* [config]

*Default:* Varies

---

**MODE\_CONV\_RADIUS**

Comma separated list of convolution radius values used by mode for both forecast and observation fields. Has the same behavior as setting FCST\_MODE\_CONV\_RADIUS and OBS\_MODE\_CONV\_RADIUS to the same value.

*Used by:* Mode

*Family:* [config]

*Default:*

---

**MODE\_CONV\_THRESH**

Comma separated list of convolution threshold values used by mode for both forecast and observation fields. Has the same behavior as setting FCST\_MODE\_CONV\_THRESH and OBS\_MODE\_CONV\_THRESH to the same value.

*Used by:* Mode

*Family:* [config]

*Default:*

---

**MODE\_FCST\_CONV\_RADIUS**

Comma separated list of convolution radius values used by mode for forecast fields.

*Used by:* Mode

*Family:* [config]

*Default:*5

---

**MODE\_FCST\_CONV\_THRESH**

Comma separated list of convolution threshold values used by mode for forecast fields.

*Used by:* Mode

*Family:* [config]

*Default:*5

---

**MODE\_FCST\_MERGE\_FLAG**

Sets the merge\_flag value in the mode config file for forecast fields. Valid values are NONE, THRESH, ENGINE, and BOTH.

*Used by:* Mode

*Family:* [config]

*Default:* THRESH

---

#### **MODE\_FCST\_MERGE\_THRESH**

Comma separated list of merge threshold values used by mode for forecast fields.

*Used by:* Mode

*Family:* [config]

*Default:* >0.45

---

#### **MODE\_MERGE\_CONFIG\_FILE**

Path to mode merge config file.

*Used by:* Mode

*Family:* [config]

*Default:* Varies

---

#### **MODE\_MERGE\_FLAG**

Sets the merge\_flag value in the mode config file for both forecast and observation fields. Has the same behavior as setting MODE\_FCST\_MERGE\_FLAG and MODE\_OBS\_MERGE\_FLAG to the same value. Valid values are NONE, THRESH, ENGINE, and BOTH.

*Used by:* Mode

*Family:* [config]

*Default:* THRESH

---

#### **MODE\_MERGE\_THRESH**

Comma separated list of merge threshold values used by mode for forecast and observation fields. Has the same behavior as setting MODE\_FCST\_MERGE\_THRESH and MODE\_OBS\_MERGE\_THRESH to the same value.

*Used by:* Mode

*Family:* [config]

*Default:* >0.45

---

#### **[deprecated]MODE\_OBS\_CONV\_RADIUS**

Please use OBS\_CONV\_MODE\_RADIUS instead. Comma separated list of convolution radius values used by mode for observation fields.

*Used by:* Mode

*Family:* [config]

*Default:* 5

---

#### **[deprecated]MODE\_OBS\_CONV\_THRESH**

Please use OBS\_MODE\_CONV\_THRESH instead. Comma separated list of convolution threshold values used by mode for observation fields.

*Used by:* Mode

*Family:* [config]

*Default:* 5

---

#### [deprecated] **MODE\_OBS\_MERGE\_FLAG**

Please use OBS\_MODE\_MERGE\_FLAG instead. Sets the merge\_flag value in the mode config file for observation fields. Valid values are NONE, THRESH, ENGINE, and BOTH.

*Used by:* Mode

*Family:* [config]

*Default:* THRESH

---

#### [deprecated] **MODE\_OBS\_MERGE\_THRESHOLD**

Please use OBS\_MODE\_MERGE\_THRESHOLD\_INSTEAD. Comma separated list of merge threshold values used by mode for observation fields.

*Used by:* Mode

*Family:* [config]

*Default:* >0.45

---

#### [deprecated] **MODE\_OUT\_DIR**

Please use MODE\_OUTPUT\_DIR instead. Output directory to write mode files.

*Used by:* Mode

*Family:* [dir]

*Default:* Varies

---

#### **MODE\_OUTPUT\_DIR**

Output directory to write mode files.

*Used by:* Mode

*Family:* [dir]

*Default:* Varies

---

#### **MODE\_OUTPUT\_TEMPLATE**

Sets the subdirectories below MODE\_OUTPUT\_DIR using a template to allow run time information. If LOOP\_BY = VALID, default value is valid time YYYYMMDDHHMM/mode. If LOOP\_BY = INIT, default value is init time YYYYMMDDHHMM/mode.

*Used by:* Mode

*Family:* [filename\_templates]

*Default:* Varies

---

#### **MODE\_VERIFICATION\_MASK\_TEMPLATE**

Template used to specify the verification mask filename for the MET tool mode.

*Used by:* Mode

*Family:* [filename\_templates]

*Default:* Varies

---

### MODE\_QUILT

True/False. If True, run all permutations of radius and threshold.

*Used by:* Mode

*Family:* [config]

*Default:* False

---

### [deprecated]MTD\_CONFIG

Please use MTD\_CONFIG\_FILE instead. Path to mode-TD configuration file.

*Used by:* MTD

*Family:* [config]

*Default:* Varies

---

### MTD\_CONFIG\_FILE

Path to mode-TD configuration file.

*Used by:* MTD

*Family:* [config]

*Default:* Varies

---

### MTD\_CONV\_RADIUS

Comma separated list of convolution radius values used by mode-TD for both forecast and observation files. Has the same behavior as setting FCST\_MTD\_CONV\_RADIUS and OBS\_MTD\_CONV\_RADIUS to the same value.

*Used by:* MTD

*Family:* [config]

*Default:*

---

### MTD\_CONV\_THRESH

Comma separated list of convolution threshold values used by mode-TD for both forecast and observation files. Has the same behavior as setting FCST\_MTD\_CONV\_THRESH and OBS\_MTD\_CONV\_THRESH to the same value.

*Used by:* MTD

*Family:* [config]

*Default:*

---

### MTD\_FCST\_CONV\_RADIUS

Comma separated list of convolution radius values used by mode-TD for forecast files.

*Used by:* MTD

*Family:* [config]

*Default:* 5

---

#### **MTD\_MIN\_VOLUME**

Sets min\_volume in the MET Mode-TD config file. Refer to the MET User's Guide for more information.

*Used by:* MTD

*Family:* [config]

*Default:*

---

#### **MTD\_SINGLE\_RUN**

Set to True to only process one data set (forecast or observation) in Mode-TD. If True, must set MTD\_SINGLE\_RUN\_SRC to either 'FCST' or 'OBS'.

*Used by:* MTD

*Family:* [config]

*Default:*

---

#### **MTD\_SINGLE\_RUN\_SRC**

Used only if MTD\_SINGLE\_RUN is set to True. Valid options are 'FCST' or 'OBS'.

*Used by:* MTD

*Family:* [config]

*Default:*

---

#### **MTD\_FCST\_CONV\_THRESH**

Comma separated list of convolution threshold values used by mode-TD for forecast files.

*Used by:* MTD

*Family:* [config]

*Default:* >0.5

---

#### **MTD\_OBS\_CONV\_RADIUS**

Comma separated list of convolution radius values used by mode-TD for observation files.

*Used by:* MTD

*Family:* [config]

*Default:* 5

---

#### **MTD\_OBS\_CONV\_THRESH**

Comma separated list of convolution threshold values used by mode-TD for observation files.

*Used by:* MTD

*Family:* [config]

*Default:* >0.5

---

#### [deprecated] MTD\_OUT\_DIR

Please use MTD\_OUTPUT\_DIR.

*Used by:* MTD

*Family:* [dir]

*Default:* Varies

---

#### MTD\_OUTPUT\_DIR

Output directory to write mode-TD files.

*Used by:* MTD

*Family:* [dir]

*Default:* Varies

---

#### MTD\_OUTPUT\_TEMPLATE

Sets the subdirectories below MTD\_OUTPUT\_DIR using a template to allow run time information. If LOOP\_BY = VALID, default value is valid time YYYYMMDDHHMM/mtd. If LOOP\_BY = INIT, default value is init time YYYYMMDDHHMM/mtd.

*Used by:* MTD

*Family:* [filename\_templates]

*Default:* Varies

---

#### MTD\_SINGLE\_DATA\_SRC

Only used if MTD\_SINGLE\_RUN is True. Determines which data set to process. Valid options are FCST and OBS.

*Used by:* MTD

*Family:* [config]

*Default:* FCST

---

#### MTD\_SINGLE\_RUN

Run mode-TD with -single option. Must set MTD\_SINGLE\_DATA\_SRC to specify which data set to process.

*Used by:* MTD

*Family:* [config]

*Default:* False

---

#### 4.6.14 N

---

##### NCAP2

Path to the “ncap2” executable.

*Used by:* PB2NC, PointStat, SeriesByLead

*Family:* [exe]

*Default:* /path/to

---

##### [deprecated] NCAP2\_EXE

Please use NCAP2.

*Used by:* PB2NC, PointStat, SeriesByLead

*Family:* [exe]

*Default:* /path/to

---

##### NCDUMP

Path to the “ncdump” executable.

*Used by:* PB2NC, PointStat, SeriesByLead

*Family:* [exe]

*Default:* /path/to

---

##### [deprecated] NCDUMP\_EXE

Please use NCDUMP.

*Used by:* PB2NC, PointStat, SeriesByLead

*Family:* [exe]

*Default:* /path/to

---

##### NC\_FILE\_TMPL

File template used to match netCDF files used for analysis.

*Used by:* PB2NC

*Family:* [filename\_templates]

*Default:* Varies

---

##### NLAT

The number of latitude points, set to a whole number. This defines the number of latitude points to incorporate into the subregion (density).

*Used by:* ExtractTiles, SeriesByLead

*Family:* [config]

*Default:* Varies

---

**NLON**

The number of longitude points, set to a whole number. This defines the number of longitude points to incorporate into the subregion (density).

*Used by:* ExtractTiles, SeriesByLead

*Family:* [config]

*Default:* Varies

---

**NO\_EE**

Set the “NO\_EE” flag for the TC Matched Pairs plotting utility.

Acceptable values: yes/no

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* no

---

**NO\_LOG**

Set the “NO\_LOG” flag for the TC Matched Pairs plotting utility.

Acceptable values: yes/no

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* no

---

**4.6.15 O**

---

**[deprecated] OBS\_<n>\_FIELD\_NAME**

Please use OBS\_PCP\_COMBINE\_<n>\_FIELD\_NAME instead. This variable is used to define a <n> hour accumulation NetCDF field in the observation dataset used in the MET tool pcp\_combine. <n> must be an integer  $\geq 1$ .

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

**[deprecated] OBS\_BUFR\_VAR\_LIST**

Please use PB2NC\_OBS\_BUFR\_VAR\_LIST instead. Specify which BUFR codes to use from the observation dataset when using the MET pb2nc tool. Format is comma separated list, e.g.:

PMO, TOB, TDO

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### [deprecated] OBS\_DATA\_INTERVAL

Specify the accumulation interval of the observation dataset used by the MET pcp\_combine tool.

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### OBS\_ENSEMBLE\_STAT\_GRID\_INPUT\_DATATYPE

Specify the data type of the input directory for grid observation files used with the MET ensemble\_stat tool. Currently valid options are NETCDF, GRIB, and GEMPAK. If set to GEMPAK, data will automatically be converted to NetCDF via GempakToCF. A similar variable exists for forecast data called FCST\_ENSEMBLE\_STAT\_INPUT\_DATATYPE.

*Used by:* EnsembleStat

*Family:* [config]

*Default:* Varies

---

#### OBS\_ENSEMBLE\_STAT\_GRID\_INPUT\_DIR

Input directory for grid observation files to use with the MET tool ensemble\_stat. A similar variable exists for forecast data called FCST\_ENSEMBLE\_STAT\_INPUT\_DIR.

*Used by:* EnsembleStat

*Family:* [dir]

*Default:* Varies

---

#### OBS\_ENSEMBLE\_STAT\_GRID\_INPUT\_TEMPLATE

Template used to specify grid observation input filenames for the MET tool ensemble\_stat. A similar variable exists for forecast data called FCST\_ENSEMBLE\_STAT\_INPUT\_TEMPLATE.

*Used by:* EnsembleStat

*Family:* [filename\_templates]

*Default:* Varies

---

#### OBS\_ENSEMBLE\_STAT\_POINT\_INPUT\_DATATYPE

Specify the data type of the input directory for point observation files used with the MET ensemble\_stat tool. Currently valid options are NETCDF, GRIB, and GEMPAK. If set to GEMPAK, data will automatically be converted to NetCDF via GempakToCF. A similar variable exists for forecast data called FCST\_ENSEMBLE\_STAT\_INPUT\_DATATYPE.

*Used by:* EnsembleStat

*Family:* [config]

*Default:* Varies

---

#### **OBS\_ENSEMBLE\_STAT\_POINT\_INPUT\_DIR**

Input directory for point observation files to use with the MET tool ensemble\_stat. A similar variable exists for forecast data called FCST\_ENSEMBLE\_STAT\_INPUT\_DIR.

*Used by:* EnsembleStat

*Family:* [dir]

*Default:* Varies

---

#### **OBS\_ENSEMBLE\_STAT\_POINT\_INPUT\_TEMPLATE**

Template used to specify point observation input filenames for the MET tool ensemble\_stat. A similar variable exists for forecast data called FCST\_ENSEMBLE\_STAT\_INPUT\_TEMPLATE.

*Used by:* EnsembleStat

*Family:* [filename\_templates]

*Default:* Varies

---

#### **OBS\_ENSEMBLE\_STAT\_FILE\_WINDOW\_BEGIN**

Used to control the lower bound of the window around the valid time to determine if a file should be used for processing by EnsembleStat. See 4.3.3 subsection called 'Use Windows to Find Valid Files.' Units are seconds. If OBS\_ENSEMBLE\_STAT\_FILE\_WINDOW\_BEGIN is not set in the config file, the value of OBS\_FILE\_WINDOW\_BEGIN will be used instead. If both file window begin and window end values are set to 0, then METplus will require an input file with an exact time match to process.

*Used by:* EnsembleStat

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_BEGIN

---

#### **OBS\_ENSEMBLE\_STAT\_FILE\_WINDOW\_END**

Used to control the upper bound of the window around the valid time to determine if a file should be used for processing by EnsembleStat. See 4.3.3 subsection called 'Use Windows to Find Valid Files.' Units are seconds. If OBS\_ENSEMBLE\_STAT\_FILE\_WINDOW\_END is not set in the config file, the value of OBS\_FILE\_WINDOW\_END will be used instead. If both file window begin and window end values are set to 0, then METplus will require an input file with an exact time match to process.

*Used by:* EnsembleStat

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_END

---

#### **OBS\_FILE\_WINDOW\_BEGIN**

Used to control the lower bound of the window around the valid time to determine if a file should be used for processing. See 4.3.3 subsection called 'Use Windows to Find Valid Files.' Units are seconds.

This value will be used for all wrappers that look for an observation file unless it is overridden by a wrapper specific configuration variable. For example, if `OBS_GRID_STAT_FILE_WINDOW_BEGIN` is set, the GridStat wrapper will use that value. If `PB2NC_FILE_WINDOW_BEGIN` is not set, then the PB2NC wrapper will use `OBS_FILE_WINDOW_BEGIN`.

A corresponding variable exists for forecast data called `FCST_FILE_WINDOW_BEGIN`.

**Used by:** EnsembleStat, GridStat, Mode, MTD, PB2NC, PointStat

**Family:** [config]

**Default:** Varies

---

### **OBS\_FILE\_WINDOW\_END**

Used to control the upper bound of the window around the valid time to determine if a file should be used for processing. See 4.3.3 subsection called 'Use Windows to Find Valid Files.' Units are seconds.

This value will be used for all wrappers that look for an observation file unless it is overridden by a wrapper specific configuration variable. For example, if `OBS_GRID_STAT_WINDOW_END` is set, the GridStat wrapper will use that value. If `PB2NC_WINDOW_END` is not set, then the PB2NC wrapper will use `OBS_WINDOW_END`.

A corresponding variable exists for forecast data called `FCST_FILE_WINDOW_END`.

**Used by:** EnsembleStat, GridStat, Mode, MTD, PB2NC, PointStat

**Family:** [config]

**Default:** Varies

---

### **OBS\_GRID\_STAT\_FILE\_WINDOW\_BEGIN**

Used to control the lower bound of the window around the valid time to determine if a file should be used for processing by GridStat. See 4.3.3 subsection called 'Use Windows to Find Valid Files.' Units are seconds. If `OBS_GRID_STAT_FILE_WINDOW_BEGIN` is not set in the config file, the value of `OBS_FILE_WINDOW_BEGIN` will be used instead. If both file window begin and window end values are set to 0, then METplus will require an input file with an exact time match to process.

**Used by:** GridStat

**Family:** [config]

**Default:** `OBS_FILE_WINDOW_BEGIN`

---

### **OBS\_GRID\_STAT\_FILE\_WINDOW\_END**

Used to control the upper bound of the window around the valid time to determine if a file should be used for processing by GridStat. See 4.3.3 subsection called 'Use Windows to Find Valid Files.' Units are seconds. If `OBS_GRID_STAT_FILE_WINDOW_END` is not set in the config file, the value of `OBS_FILE_WINDOW_END` will be used instead. If both file window begin and window end values are set to 0, then METplus will require an input file with an exact time match to process.

**Used by:** GridStat

**Family:** [config]

**Default:** `OBS_FILE_WINDOW_END`

---

**OBS\_GRID\_STAT\_INPUT\_DATATYPE**

See FCST\_GRID\_STAT\_INPUT\_DATATYPE 4.6.6.

*Used by:* GridStat

*Family:* [config]

*Default:* Varies

---

**OBS\_GRID\_STAT\_INPUT\_DIR**

See FCST\_GRID\_STAT\_INPUT\_DIR 4.6.6.

*Used by:* GridStat

*Family:* [dir]

*Default:* Varies

---

**OBS\_GRID\_STAT\_INPUT\_TEMPLATE**

See FCST\_GRID\_STAT\_INPUT\_TEMPLATE 4.6.6.

*Used by:* GridStat

*Family:* [filename\_templates]

*Default:* Varies

---

**OBS\_GRID\_STAT\_PROB\_THRESH**

See FCST\_GRID\_STAT\_PROB\_THRESH 4.6.6.

*Used by:* GridStat

*Family:* [config]

*Default:* ==0.1

---

**[deprecated] OBS\_GEMPAK\_INPUT\_DIR**

Specify the input directory for GEMPAK formatted observation files. Use GEMPAKTOCF\_INPUT\_DIR if running GempakToCF from the PROCESS\_LIST.

*Used by:* PcpCombine

*Family:* [dir]

*Default:* Varies

---

**[deprecated] OBS\_GEMPAK\_TEMPLATE**

Filename template used to filter GEMPAK formatted observation files. Use GEMPAKTOCF\_INPUT\_TEMPLATE if running GempakToCF from the PROCESS\_LIST.

*Used by:* PcpCombine

*Family:* [filename\_templates]

*Default:* Varies

---

**[deprecated] OBS\_INPUT\_DIR**

Please use OBS\_POINT\_STAT\_INPUT\_DIR instead. Specify the input directory for observation files.

*Used by:* PointStat

*Family:* [dir]

*Default:* Varies

---

**[deprecated] OBS\_INPUT\_DIR\_REGEX**

Please use OBS\_POINT\_STAT\_INPUT\_DIR instead. Specify the regular expression to use when searching for observation file input directories.

*Used by:* PointStat

*Family:* [regex\_pattern]

*Default:* Varies

---

**[deprecated] OBS\_INPUT\_FILE\_REGEX**

Please use OBS\_POINT\_STAT\_INPUT\_TEMPLATE instead. Regular expression used to filter observation input files used in the analysis.

*Used by:* PointStat,

*Family:* [regex\_pattern]

*Default:* Varies

---

**[deprecated] OBS\_INPUT\_FILE\_TEMPL**

Please use OBS\_POINT\_STAT\_INPUT\_TEMPLATE instead. Specify the filename template to use for observation input files.

*Used by:* PointStat,

*Family:* [filename\_templates]

*Default:* Varies

---

**[deprecated] OBS\_IS\_DAILY\_FILE**

Please use OBS\_PCP\_COMBINE\_IS\_DAILY\_FILE instead. Specify whether the forecast file is a daily file or not.

Acceptable values: true/false

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

**OBS\_IS\_PROB**

Used when setting OBS\_\* variables to process forecast data for comparisons with mtd. Specify whether the observation data are probabilistic or not. See FCST\_IS\_PROB [4.6.6](#).

Acceptable values: true/false

*Used by:* EnsembleStat, GridStat, Mode, MTD, PointStat

*Family:* [config]

*Default:* False

---

### [deprecated] OBS\_LEVEL

Please use OBS\_PCP\_COMBINE\_INPUT\_LEVEL instead. Specify what accumulation level should be used from the observation data for the analysis. See FCST\_LEVEL for more information

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

### OBS\_MODE\_CONV\_RADIUS

See FCST\_MODE\_CONV\_RADIUS [4.6.6](#).

*Used by:* Mode

*Family:* [config]

*Default:*

---

### OBS\_MODE\_CONV\_THRESH

See FCST\_MODE\_CONV\_THRESH [4.6.6](#).

*Used by:* Mode

*Family:* [config]

*Default:*

---

### OBS\_MODE\_FILE\_WINDOW\_BEGIN

Used to control the lower bound of the window around the valid time to determine if a file should be used for processing by Mode. See [4.3.3](#) subsection called 'Use Windows to Find Valid Files.' Units are seconds. If OBS\_MODE\_FILE\_WINDOW\_BEGIN is not set in the config file, the value of OBS\_FILE\_WINDOW\_BEGIN will be used instead. If both file window begin and window end values are set to 0, then METplus will require an input file with an exact time match to process.

*Used by:* Mode

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_BEGIN

---

### OBS\_MODE\_FILE\_WINDOW\_END

Used to control the upper bound of the window around the valid time to determine if a file should be used for processing by Mode. See [4.3.3](#) subsection called 'Use Windows to Find Valid Files.' Units are seconds. If OBS\_MODE\_FILE\_WINDOW\_END is not set in the config file, the value of OBS\_FILE\_WINDOW\_END

will be used instead. If both file window begin and window end values are set to 0, then METplus will require an input file with an exact time match to process.

*Used by:* Mode

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_END

---

#### **OBS\_MODE\_MERGE\_FLAG**

See FCST\_MODE\_MERGE\_FLAG [4.6.6](#).

*Used by:* Mode

*Family:* [config]

*Default:*

---

#### **OBS\_MODE\_MERGE\_THRESH**

See FCST\_MODE\_MERGE\_THRESH [4.6.6](#).

*Used by:* Mode

*Family:* [config]

*Default:*

---

#### **OBS\_MODE\_INPUT\_DATATYPE**

See FCST\_MODE\_INPUT\_DATATYPE [4.6.6](#).

*Used by:* Mode

*Family:* [config]

*Default:* Varies

---

#### **OBS\_MODE\_INPUT\_DIR**

See FCST\_MODE\_INPUT\_DIR [4.6.6](#).

*Used by:* Mode

*Family:* [dir]

*Default:* Varies

---

#### **OBS\_MODE\_INPUT\_TEMPLATE**

See FCST\_MODE\_INPUT\_TEMPLATE [4.6.6](#).

*Used by:* Mode

*Family:* [filename\_templates]

*Default:* Varies

---

#### **OBS\_MTD\_CONV\_RADIUS**

See FCST\_MTD\_CONV\_RADIUS [4.6.6](#).

*Used by:* MTD

*Family:* [config]

*Default:*

---

### **OBS\_MTD\_CONV\_THRESH**

See FCST\_MTD\_CONV\_THRESH 4.6.6.

*Used by:* MTD

*Family:* [config]

*Default:*

---

### **OBS\_MTD\_FILE\_WINDOW\_BEGIN**

Used to control the lower bound of the window around the valid time to determine if a file should be used for processing by MTD. See 4.3.3 subsection called 'Use Windows to Find Valid Files.' Units are seconds. If OBS\_MTD\_FILE\_WINDOW\_BEGIN is not set in the config file, the value of OBS\_FILE\_WINDOW\_BEGIN will be used instead. If both file window begin and window end values are set to 0, then METplus will require an input file with an exact time match to process.

*Used by:* MTD

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_BEGIN

---

### **OBS\_MTD\_FILE\_WINDOW\_END**

Used to control the upper bound of the window around the valid time to determine if a file should be used for processing by MTD. See 4.3.3 subsection called 'Use Windows to Find Valid Files.' Units are seconds. If OBS\_MTD\_FILE\_WINDOW\_END is not set in the config file, the value of OBS\_FILE\_WINDOW\_END will be used instead. If both file window begin and window end values are set to 0, then METplus will require an input file with an exact time match to process.

*Used by:* MTD

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_END

---

### **OBS\_MTD\_INPUT\_DATATYPE**

See FCST\_MTD\_INPUT\_DATATYPE 4.6.6.

*Used by:* MTD

*Family:* [config]

*Default:* Varies

---

### **OBS\_MTD\_INPUT\_DIR**

See FCST\_MTD\_INPUT\_DIR 4.6.6.

*Used by:* MTD

*Family:* [dir]

*Default:* Varies

---

**OBS\_MTD\_INPUT\_TEMPLATE**

See FCST\_MTD\_INPUT\_TEMPLATE [4.6.6](#).

*Used by:* MTD

*Family:* [filename\_templates]

*Default:* Varies

---

**[deprecated] OBS\_NAME**

No longer used. Provide a string to identify the observation dataset name.

*Used by:* PointStat

*Family:* [config]

*Default:* Varies

---

**[deprecated] OBS\_NATIVE\_DATA\_TYPE**

Specify the data format of the observation data. Use OBS\_PCP\_COMBINE\_INPUT\_DATATYPE instead.

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

**OBS\_PCP\_COMBINE\_<n>\_FIELD\_NAME**

See FCST\_PCP\_COMBINE\_<n>\_FIELD\_NAME [4.6.6](#).

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

**OBS\_PCP\_COMBINE\_DATA\_INTERVAL**

See FCST\_PCP\_COMBINE\_DATA\_INTERVAL [4.6.6](#).

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

**OBS\_PCP\_COMBINE\_DERIVE\_LOOKBACK**

See FCST\_PCP\_COMBINE\_DERIVE\_LOOKBACK [4.6.6](#).

*Used by:* PcpCombine

*Family:* [config]

*Default:* ADD

---

**OBS\_PCP\_COMBINE\_INPUT\_DATATYPE**

See FCST\_PCP\_COMBINE\_INPUT\_DATA\_TYPE 4.6.6.

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

**OBS\_PCP\_COMBINE\_INPUT\_DIR**

See FCST\_PCP\_COMBINE\_INPUT\_DIR 4.6.6.

*Used by:* PcpCombine

*Family:* [dir]

*Default:* Varies

---

**OBS\_PCP\_COMBINE\_INPUT\_LEVEL**

See FCST\_PCP\_COMBINE\_INPUT\_LEVEL 4.6.6.

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

**OBS\_PCP\_COMBINE\_INPUT\_TEMPLATE**

See FCST\_PCP\_COMBINE\_INPUT\_TEMPLATE 4.6.6.

*Used by:* PcpCombine

*Family:* [filename\_templates]

*Default:* Varies

---

**OBS\_PCP\_COMBINE\_IS\_DAILY\_FILE**

See FCST\_PCP\_COMBINE\_IS\_DAILY\_FILE 4.6.6.

Acceptable values: true/false

*Used by:* PcpCombine

*Family:* [config]

*Default:* False

---

**OBS\_PCP\_COMBINE\_METHOD**

See FCST\_PCP\_COMBINE\_METHOD 4.6.6.

*Used by:* PcpCombine

*Family:* [config]

*Default:* None

---

**OBS\_PCP\_COMBINE\_MIN\_FORECAST**

See FCST\_PCP\_COMBINE\_MIN\_FORECAST 4.6.6.

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### **OBS\_PCP\_COMBINE\_MAX\_FORECAST**

See FCST\_PCP\_COMBINE\_MAX\_FORECAST [4.6.6](#).

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### **OBS\_PCP\_COMBINE\_OUTPUT\_DIR**

See FCST\_PCP\_COMBINE\_OUTPUT\_DIR [4.6.6](#).

*Used by:* PcpCombine

*Family:* [dir]

*Default:* Varies

---

#### **OBS\_PCP\_COMBINE\_OUTPUT\_TEMPLATE**

See FCST\_PCP\_COMBINE\_OUTPUT\_TEMPLATE [4.6.6](#).

*Used by:* PcpCombine

*Family:* [filename\_templates]

*Default:* Varies

---

#### **OBS\_PCP\_COMBINE\_RUN**

See FCST\_PCP\_COMBINE\_RUN [4.6.6](#).

Acceptable values: true/false

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### **OBS\_PCP\_COMBINE\_STAT\_LIST**

See FCST\_PCP\_COMBINE\_STAT\_LIST [4.6.6](#).

Acceptable values: sum, min, max, range, mean, stdev, vld\_count

*Used by:* PcpCombine

*Family:* [config]

*Default:* Varies

---

#### **OBS\_PCP\_COMBINE\_TIMES\_PER\_FILE**

See FCST\_PCP\_COMBINE\_TIMES\_PER\_FILE [4.6.6](#).

*Used by:* PcpCombine

*Family:* [config]

*Default:*

---

#### **OBS\_POINT\_STAT\_FILE\_WINDOW\_BEGIN**

Used to control the lower bound of the window around the valid time to determine if a file should be used for processing by PointStat. See 4.3.3 subsection called 'Use Windows to Find Valid Files.' Units are seconds. If OBS\_POINT\_STAT\_FILE\_WINDOW\_BEGIN is not set in the config file, the value of OBS\_FILE\_WINDOW\_BEGIN will be used instead. If both file window begin and window end values are set to 0, then METplus will require an input file with an exact time match to process.

*Used by:* PointStat

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_BEGIN

---

#### **OBS\_POINT\_STAT\_FILE\_WINDOW\_END**

Used to control the upper bound of the window around the valid time to determine if a file should be used for processing by PointStat. See 4.3.3 subsection called 'Use Windows to Find Valid Files.' Units are seconds. If OBS\_POINT\_STAT\_FILE\_WINDOW\_END is not set in the config file, the value of OBS\_FILE\_WINDOW\_END will be used instead. If both file window begin and window end values are set to 0, then METplus will require an input file with an exact time match to process.

*Used by:* PointStat

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_END

---

#### **OBS\_POINT\_STAT\_INPUT\_DATATYPE**

See FCST\_POINT\_STAT\_INPUT\_DATATYPE 4.6.6.

*Used by:* PointStat

*Family:* [config]

*Default:* Varies

---

#### **OBS\_POINT\_STAT\_INPUT\_DIR**

See FCST\_POINT\_STAT\_INPUT\_DIR 4.6.6.

*Used by:* PointStat

*Family:* [dir]

*Default:* Varies

---

#### **OBS\_POINT\_STAT\_INPUT\_TEMPLATE**

See FCST\_POINT\_STAT\_INPUT\_TEMPLATE 4.6.6.

*Used by:* GriPointStat

*Family:* [filename\_templates]

*Default:* Varies

---

#### **OBS\_POINT\_STAT\_WINDOW\_BEGIN**

Passed to the point\_stat MET config file to determine the range of data within a file that should be used for processing.

Units are seconds. If the variable is not set, point\_stat will use OBS\_WINDOW\_BEGIN.

*Used by:* PointStat

*Family:* [config]

*Default:* Varies

---

#### **OBS\_POINT\_STAT\_WINDOW\_END**

Passed to the point\_stat MET config file to determine the range of data within a file that should be used for processing. Units are seconds. If the variable is not set, point\_stat will use OBS\_WINDOW\_END.

*Used by:* PB2NC, PointStat

*Family:* [config]

*Default:* Varies

---

#### **OBS\_REGRID\_DATA\_PLANE\_INPUT\_DATATYPE**

See FCST\_REGRID\_DATA\_PLANE\_INPUT\_DATATYPE [4.6.6](#).

*Used by:* RegridDataPlane

*Family:* [config]

*Default:* Varies

---

#### **OBS\_REGRID\_DATA\_PLANE\_INPUT\_DIR**

See FCST\_REGRID\_DATA\_PLANE\_INPUT\_DIR [4.6.6](#).

*Used by:* RegridDataPlane

*Family:* [dir]

*Default:* Varies

---

#### **OBS\_REGRID\_DATA\_PLANE\_INPUT\_TEMPLATE**

See FCST\_REGRID\_DATA\_PLANE\_INPUT\_TEMPLATE [4.6.6](#).

*Used by:* RegridDataPlane

*Family:* [filename\_templates]

*Default:* Varies

---

#### **OBS\_REGRID\_DATA\_PLANE\_OUTPUT\_TEMPLATE**

See FCST\_REGRID\_DATA\_PLANE\_OUTPUT\_TEMPLATE [4.6.6](#).

*Used by:* RegridDataPlane

*Family:* [filename\_templates]

*Default:* Varies

---

### **OBS\_REGRID\_DATA\_PLANE\_TEMPLATE**

See FCST\_REGRID\_DATA\_PLANE\_TEMPLATE 4.6.6.

*Used by:* RegridDataPlane

*Family:* [filename\_templates]

*Default:* Varies

---

### **OBS\_REGRID\_DATA\_PLANE\_OUTPUT\_DIR**

See FCST\_REGRID\_DATA\_PLANE\_OUTPUT\_DIR 4.6.6.

*Used by:* RegridDataPlane

*Family:* [dir]

*Default:* Varies

---

### **[deprecated] OBS\_VAR**

Specify the string for the observation variable used in the analysis. See OBS\_VAR<n>\_NAME, OBS\_VAR<n>\_LEVELS, OBS\_VAR<n>\_OPTIONS and OBS\_VAR<n>\_THRESH where n = integer >= 1.

*Used by:* GridStat

*Family:* [config]

*Default:* Varies

---

### **OBS\_VAR\_LEVEL**

Specify the values of the OBS\_VAR\_LEVEL column in the MET .stat file to use.

*Used by:* StatAnalysis

*Family:* [config]

*Default:* Varies

---

### **OBS\_VAR\_NAME**

Specify the values of the OBS\_VAR\_NAME column in the MET .stat file to use. This is optional in the METplus configuration file for running with LOOP\_ORDER = times

*Used by:* StatAnalysis

*Family:* [config]

*Default:* Varies

---

### **OBS\_VAR<n>\_LEVELS**

Define the levels for the <n>th observation variable to be used in the analysis where <n> is an integer >= 1. The value can be a single item or a comma separated list of items. You can define NetCDF levels, such as (0,\*,\*), but you will need to surround these values with quotation marks so that the commas in the item are not interpreted as an item delimiter. Some examples:

```
OBS_VAR1_LEVELS = A06, P500
OBS_VAR2_LEVELS = "(0,*,*)", "(1,*,*)"
```

If OBS\_VAR<n>\_LEVELS is not set but FCST\_VAR<n>\_LEVELS is, the same information will be used for both variables. There can be <n> number of these variables defined in configuration files, simply increment the “\_VAR1\_” string to match the total number of variables being used, e.g.:

```
OBS_VAR1_LEVELS
OBS_VAR2_LEVELS
...
OBS_VAR<n>_LEVELS
```

**Used by:** GridStat, EnsembleStat, PointStat, Mode, MTD, PcpCombine

**Family:** [config]

**Default:** Varies

---

#### OBS\_VAR<n>\_NAME

Define the name for the <n>th observation variable to be used in the analysis where <n> is an integer >= 1. If OBS\_VAR<n>\_NAME is not set but FCST\_VAR<n>\_NAME is, the same information will be used for both variables. There can be <n> number of these variables defined in configuration files, simply increment the “\_VAR1\_” string to match the total number of variables being used, e.g.:

```
OBS_VAR1_NAME
OBS_VAR2_NAME
...
OBS_VAR<n>_NAME
```

**Used by:** GridStat, EnsembleStat, PointStat, Mode, MTD, PcpCombine

**Family:** [config]

**Default:** Varies

---

#### OBS\_VAR<n>\_OPTIONS

Define the options for the <n>th observation variable to be used in the analysis where <n> is an integer >= 1. These addition options will be applied to every name/level/threshold combination for VAR<n>. If OBS\_VAR<n>\_OPTIONS is not set but FCST\_VAR<n>\_OPTIONS is, the same information will be used for both variables. There can be <n> number of these variables defined in configuration files, simply increment the “\_VAR1\_” string to match the total number of variables being used, e.g.:

```
OBS_VAR1_OPTIONS
OBS_VAR2_OPTIONS
...
OBS_VAR<n>_OPTIONS
```

**Used by:** GridStat, EnsembleStat, PointStat, Mode, MTD, PcpCombine

**Family:** [config]

*Default:* Varies

---

### **OBS\_VAR<n>\_THRESH**

Define the threshold(s) for the <n>th observation variable to be used in the analysis where <n> is an integer  $\geq 1$ . The value can be a single item or a comma separated list of items that must start with a comparison operator (>, >=, ==, !=, <, <=, gt, ge, eq, ne, lt, le). If OBS\_VAR<n>\_THRESH is not set but FCST\_VAR<n>\_THRESH is, the same information will be used for both variables. There can be <n> number of these variables defined in configuration files, simply increment the “\_VAR1\_” string to match the total number of variables being used, e.g.:

OBS\_VAR1\_THRESH

OBS\_VAR2\_THRESH

...

OBS\_VAR<n>\_THRESH

*Used by:* GridStat, EnsembleStat, PointStat, Mode, MTD, PcpCombine

*Family:* [config]

*Default:* Varies

---

### **[deprecated] OBS\_WINDOW\_BEG**

Please use OBS\_WINDOW\_BEGIN.

*Used by:* PB2NC, PointStat

*Family:* [config]

*Default:* Varies

---

### **OBS\_WINDOW\_BEGIN**

Passed to the MET config file to determine the range of data within a file that should be used for processing. Units are seconds. This value will be used for all wrappers that look for an observation file unless it is overridden by a wrapper specific configuration variable. For example, if OBS\_POINT\_STAT\_WINDOW\_BEGIN is set, the PointStat wrapper will use that value. If PB2NC\_WINDOW\_BEGIN is not set, then the PB2NC wrapper will use OBS\_WINDOW\_BEGIN.

A corresponding variable exists for forecast data called FCST\_WINDOW\_BEGIN.

*Used by:* PB2NC, PointStat

*Family:* [config]

*Default:* Varies

---

### **OBS\_WINDOW\_END**

Passed to the MET config file to determine the range of data within a file that should be used for processing. Units are seconds. This value will be used for all wrappers that look for an observation file unless it is overridden by a wrapper specific configuration variable. For example, if OBS\_POINT\_STAT\_WINDOW\_END is set, the PointStat wrapper will use that value. If PB2NC\_WINDOW\_END is not set, then the PB2NC wrapper will use OBS\_WINDOW\_END.

A corresponding variable exists for forecast data called FCST\_WINDOW\_END.

*Used by:* PB2NC, PointStat

*Family:* [config]

*Default:* Varies

---

### OBTYPE

Provide a string to represent the type of observation data used in the analysis. This is the observation time listed in the MET .stat files and is used in setting output filename.

*Used by:* EnsembleStat, GridStat, Mode, MTD, PointStat, StatAnalysis

*Family:* [config]

*Default:* Varies

---

### [deprecated] OB\_TYPE

Please use OBTYPE instead.

*Used by:* EnsembleStat, GridStat, Mode, StatAnalysis

*Family:* [config]

*Default:* Varies

---

### OUTPUT\_BASE

Provide a path to the top level output directory for METplus.

*Used by:* All

*Family:* [dir]

*Default:* Varies

---

### [deprecated] OVERWRITE\_NC\_OUTPUT

Please use PB2NC\_SKIP\_IF\_OUTPUT\_EXISTS instead. Specify whether to overwrite the netCDF output or not when using the MET pb2nc tool.

Acceptable values: yes/no

*Used by:* PB2NC

*Family:* [config]

*Default:* yes

---

### OVERWRITE\_TRACK

Specify whether to overwrite the track data or not.

Acceptable values: yes/no

*Used by:* ExtractTiles

*Family:* [config]

*Default:* no

---

#### 4.6.16 P

---

##### PARM\_BASE

This variable will automatically be set by METplus when it is started. Specifies the top level METplus parameter file directory. You can override this value by setting the environment variable METPLUS\_PARM\_BASE to another directory containing a copy of the METplus parameter file directory. If the environment variable is not set, the parm directory corresponding to the calling script is used. It is recommended that this variable is not set by the user. If it is set and is not equivalent to the value determined by METplus, execution will fail.

*Used by:* All

*Family:* [dir]

*Default:* {METPLUS\_BASE}/parm

---

##### PB2NC\_CONFIG\_FILE

Specify the absolute path to the configuration file for the MET pb2nc tool.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

##### PB2NC\_FILE\_WINDOW\_BEGIN

Used to control the lower bound of the window around the valid time to determine if a file should be used for processing by PB2NC. See 4.3.3 subsection called 'Use Windows to Find Valid Files.' Units are seconds. If PB2NC\_FILE\_WINDOW\_BEGIN is not set in the config file, the value of OBS\_FILE\_WINDOW\_BEGIN will be used instead. If both file window begin and window end values are set to 0, then METplus will require an input file with an exact time match to process.

*Used by:* PB2NC

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_BEGIN

---

##### PB2NC\_FILE\_WINDOW\_END

Used to control the upper bound of the window around the valid time to determine if a file should be used for processing by PB2NC. See 4.3.3 subsection called 'Use Windows to Find Valid Files.' Units are seconds. If PB2NC\_FILE\_WINDOW\_END is not set in the config file, the value of OBS\_FILE\_WINDOW\_END will be used instead. If both file window begin and window end values are set to 0, then METplus will require an input file with an exact time match to process.

*Used by:* PB2NC

*Family:* [config]

*Default:* OBS\_FILE\_WINDOW\_END

---

**PB2NC\_GRID**

Specify a grid to use with the MET pb2nc tool.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

**PB2NC\_INPUT\_DATATYPE**

Specify the data type of the input directory for prepbufr files used with the MET pb2nc tool. Currently valid options are NETCDF, GRIB, and GEMPAK. If set to GEMPAK, data will automatically be converted to NetCDF via GempakToCF.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

**PB2NC\_MESSAGE\_TYPE**

Specify which PREPBUFR (PB) message types to convert using the MET pb2nc tool.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

**PB2NC\_OBS\_BUFR\_VAR\_LIST**

Specify which BUFR codes to use from the observation dataset when using the MET pb2nc tool. Format is comma separated list, e.g.:

PMO, TOB, TDO

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

**PB2NC\_OFFSETS**

A list of potential offsets (in hours) that can be found in the prepbufr input template. METplus will check if a file with a given offset exists in the order specified in this list, to be sure to put favored offset values first.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

**PB2NC\_OUTPUT\_DIR**

Specify the directory where files will be written from the MET pb2nc tool.

*Used by:* PB2NC

*Family:* [dir]

*Default:* Varies

---

#### **PB2NC\_POLY**

Specify a polygon to be used with the MET pb2nc tool.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### **PB2NC\_SKIP\_IF\_OUTPUT\_EXISTS**

If True, do not run PB2NC if output file already exists. Set to False to overwrite files.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### **PB2NC\_STATION\_ID**

Specify the ID of the station to use with the MET pb2nc tool.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### **PB2NC\_TIME\_SUMMARY\_FLAG**

Specify the time summary flag item in the MET pb2nc config file. Refer to the MET User's Guide for more information.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### **PB2NC\_TIME\_SUMMARY\_BEG**

Specify the time summary beg item in the MET pb2nc config file. Refer to the MET User's Guide for more information.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### **PB2NC\_TIME\_SUMMARY\_END**

Specify the time summary end item in the MET pb2nc config file. Refer to the MET User's Guide for more information.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### **PB2NC\_TIME\_SUMMARY\_VAR\_NAMES**

Specify the time summary obs\_var list item in the MET pb2nc config file. Refer to the MET User's Guide for more information.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### **PB2NC\_TIME\_SUMMARY\_TYPES**

Specify the time summary type list item in the MET pb2nc config file. Refer to the MET User's Guide for more information.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### **PB2NC\_WINDOW\_BEGIN**

Passed to the pb2nc MET config file to determine the range of data within a file that should be used for processing.

Units are seconds. If the variable is not set, pb2nc will use OBS\_WINDOW\_BEGIN.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### **PB2NC\_WINDOW\_END**

Passed to the pb2nc MET config file to determine the range of data within a file that should be used for processing. Units are seconds. If the variable is not set, pb2nc will use OBS\_WINDOW\_END.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### **[deprecated] PCP\_COMBINE\_METHOD**

SPlease use [FCST/OBS]\_PCP\_COMBINE\_METHOD instead.

*Used by:* PcpCombine

*Family:* [config]

*Default:* ADD

---

#### **PCP\_COMBINE\_SKIP\_IF\_OUTPUT\_EXISTS**

If True, do not run pcp\_combine if output file already exists. Set to False to overwrite files.

*Used by:* PcpCombine

*Family:* [config]

*Default:* False

---

### PLOTTING\_OUTPUT\_DIR

Specify the output directory where plots will be saved. This is the base directory where the output from running `make_plots_wrapper` will be put.

*Used by:* MakePlots

*Family:* [dir]

*Default:* Varies

---

### PLOTTING\_SCRIPTS\_DIR

Specify the directory where the plotting scripts are located. It is recommended to set this to `{METPLUS_BASE}/ush/plotting_scripts`.

*Used by:* MakePlots

*Family:* [dir]

*Default:* Varies

---

### PLOT\_CONFIG\_OPTS

Specify plot configuration options for the TC Matched Pairs plotting tool.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

### PLOT\_STATS\_LIST

This is a list of the statistics to calculate and create plots for. Specify the list in a comma-separated list, e.g.:

acc, bias, rmse

The list of valid options varies depending on line type that was used during the filtering of `stat_analysis_wrapper`. For SL1L2, VL1L2 valid options are bias, rms, mss, rsd, rmse\_md, rmse\_pv, pcor, fbar, and fbar\_obar. For SAL1L2, VAL1L2, the valid options is acc. For VCNT, bias, fbar, fbar\_obar, speed\_err, dir\_err, rmsve, vdiff\_speed, vdiff\_dir, rsd, fbar\_speed, fbar\_dir, fbar\_obar\_speed, and fbar\_obar\_dir.

*Used by:* MakePlots

*Family:* [config]

*Default:* Varies

---

### PLOT\_TIME

In StatAnalysis, this specifies the way to treat the date information, where valid options are valid and init.

*Used by:* StatAnalysis

*Family:* [config]

*Default:* Varies

---

### **PLOT\_TYPES**

Specify what plot types are desired for the TC Matched Pairs plotting tool. By default, a boxplot is generated if this is undefined in the configuration file. If other plots are requested and a boxplot is also desired, you must explicitly list *boxplot* in your list of plot types. Supported plot types: BOXPLOT, POINT, MEAN, MEDIAN, RELPERF (relative performance), RANK (time series of ranks for the first model), SCATTER, SKILL\_MN (mean skill scores) and SKILL\_MD (median skill scores).

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

### **POINT\_STAT\_CONFIG\_FILE**

Specify the absolute path to the configuration file to be used with the MET point\_stat tool.

*Used by:* PointStat

*Family:* [config]

*Default:* Varies

---

### **POINT\_STAT\_GRID**

Specify the grid to use with the MET point\_stat tool.

*Used by:* PointStat

*Family:* [config]

*Default:* Varies

---

### **POINT\_STAT\_MESSAGE\_TYPE**

Specify which PREPBUFR message types to process with the MET point\_stat tool.

*Used by:* PointStat

*Family:* [config]

*Default:* Varies

---

### **POINT\_STAT\_OUTPUT\_DIR**

Specify the directory where output files from the MET point\_stat tool are written.

*Used by:* PointStat

*Family:* [dir]

*Default:* Varies

---

### **POINT\_STAT\_OUTPUT\_TEMPLATE**

Sets the subdirectories below POINT\_STAT\_OUTPUT\_DIR using a template to allow run time information. If LOOP\_BY = VALID, default value is valid time YYYYMMDDHHMM/point\_stat. If LOOP\_BY = INIT, default value is init time YYYYMMDDHHMM/point\_stat.

*Used by:* PointStat

*Family:* [filename\_templates]

*Default:* Varies

---

### **POINT\_STAT\_POLY**

Specify a polygon to use with the MET point\_stat tool.

*Used by:* PointStat

*Family:* [config]

*Default:* Varies

---

### **POINT\_STAT\_STATION\_ID**

Specify the ID of a specific station to use with the MET point\_stat tool.

*Used by:* PointStat

*Family:* [config]

*Default:* Varies

---

### **POINT\_STAT\_VERIFICATION\_MASK\_TEMPLATE**

Template used to specify the verification mask filename for the MET tool point\_stat.

*Used by:* PointStat

*Family:* [filename\_templates]

*Default:* Varies

---

### **PREFIX**

This corresponds to the optional -prefix flag of the plot\_TCM-PR.R script (which is wrapped by TCM-PRPlotter). This is the output file name prefix.

*Used by:* TCM-PRPlotter

*Family:* [config]

*Default:* Varies

---

### **[deprecated] PREPBUFR\_DATA\_DIR**

Please use PB2NC\_INPUT\_DIR instead. Specify the directory where the PREPBUFR data are located for the MET pb2nc tool.

*Used by:* PB2NC

*Family:* [dir]

*Default:* Varies

---

### **[deprecated] PREPBUFR\_DIR\_REGEX**

Regular expression to use when searching for PREPBUFR data.

*Used by:* PB2NC

*Family:* [regex\_pattern]

*Default:* Varies

---

#### [deprecated] PREPBUFR\_FILE\_REGEX

Regular expression to use when searching for PREPBUFR files.

*Used by:* PB2NC

*Family:* [regex\_pattern]

*Default:* Varies

---

#### [deprecated] PREPBUFR\_MODEL\_DIR\_NAME

Please put the value previously used here in the PB2NC\_INPUT\_DIR path. Specify the name of the model being used with the MET pb2nc tool.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### PROCESS\_LIST

Specify the list of processes for METplus to perform, in a comma separated list.

*Used by:* All

*Family:* [config]

*Default:* Varies

---

#### [deprecated] PROJ\_DIR

A directory for generic use. The user can store input files (if INPUT\_BASE is not defined), intermediate files, and any other project-related files.

*Used by:* PB2NC, PointStat, TcStat

*Family:* [dir]

*Default:* Varies

---

### 4.6.17 Q

### 4.6.18 R

---

#### [deprecated] REFERENCE\_TMPL

Please use TC\_PAIRS\_BDECK\_TEMPLATE.

*Used by:* TcPairs

*Family:* [filename\_templates]

*Default:* Varies

---

## REGION

Specify the values of the VX\_MASK column in the MET .stat file to use. This is optional in the METplus configuration file for running with LOOP\_ORDER = times

*Used by:* StatAnalysis

*Family:* [config]

*Default:*

---

REGION\_LIS

A list of the regions of interest. This is the list of regions for plotting verification.

*Used by:* MakePlots, StatAnalysis

*Family:* [config]

*Default:* Varies

---

## REGRID\_DATA\_PLANE\_METHOD

Sets the method used by regrid\_data\_plane. See MET User's Guide for more information.

*Used by:* RegridDataPlane

*Family:* [config]

*Default:*

---

## REGRID\_DATA\_PLANE\_SKIP\_IF\_OUTPUT\_EXISTS

If True, do not run regrid\_data\_plane if output file already exists. Set to False to overwrite files.

*Used by:* RegridDataPlane

*Family:* [config]

*Default:* False

---

## REGRID\_DATA\_PLANE\_WIDTH

Sets the width used by regrid\_data\_plane. See MET User's Guide for more information.

*Used by:* RegridDataPlane

*Family:* [config]

*Default:* 1

---

## REGRID\_DATA\_PLANE\_VERIF\_GRID

Specify the absolute path to a file containing information about the desired output grid from the MET regrid\_data\_plane tool.

*Used by:* RegridDataPlane

*Family:* [config]

*Default:* Varies

---

**REGRID\_TO\_GRID**

If supported, provide the output grid that is desired from the MET tool being used in the analysis.

*Used by:* MakePlots, PointStat

*Family:* [config]

*Default:* Varies

---

**REGRID\_USING\_MET\_TOOL**

Specify whether to regrid using the MET regrid\_data\_plane tool or not.

Acceptable values: yes/no

*Used by:* SeriesByInit, SeriesByLead

*Family:* [config]

*Default:* yes

---

**RM**

Specify the path to the Linux “rm” executable.

*Used by:* PB2NC, PointStat, SeriesByLead

*Family:* [exe]

*Default:* /path/to

---

**[deprecated] RM\_EXE**

Please use RM.

*Used by:* PB2NC, PointStat, SeriesByLead

*Family:* [exe]

*Default:* /path/to

---

**RP\_DIFF**

This corresponds to the optional -rp\_diff flag of the plot\_TCM-PR.R script (which is wrapped by TCM-PRPlotter). This a comma-separated list of thresholds to specify meaningful differences for the relative performance plot.

*Used by:* TCM-PRPlotter

*Family:* [config]

*Default:* Varies

---

**4.6.19 S**

---

**SAVE**

Corresponds to the optional `-save` flag in `plot_TCMPR.R` (which is wrapped by `TCMPRPlotter`). This is a yes/no value to indicate whether to save the image (yes).

*Used by:* `TCMPRPlotter`

*Family:* [config]

*Default:* Varies

---

### SAVE\_DATA

Corresponds to the optional `-save_data` flag in `plot_TCMPR.R` (which is wrapped by `TCMPRPlotter`). Indicates whether to save the filtered track data to a file instead of deleting it.

*Used by:* `TCMPRPlotter`

*Family:* [config]

*Default:* Varies

---

### SCATTER\_X

Corresponds to the optional `-scatter_x` flag in `plot_TCMPR.R` (which is wrapped by `TCMPRPlotter`). This is a comma-separated list of x-axis variable columns to plot.

*Used by:* `TCMPRPlotter`

*Family:* [config]

*Default:* Varies

---

### SCATTER\_Y

Corresponds to the optional `-scatter_y` flag in `plot_TCMPR.R` (which is wrapped by `TCMPRPlotter`). This is a comma-separated list of y-axis variable columns to plot.

*Used by:* `TCMPRPlotter`

*Family:* [config]

*Default:* Varies

---

### SCRUB\_STAGING\_DIR

Remove staging directory after METplus has completed running if set to True. Set to False to preserve data for subsequent runs.

*Used by:* All

*Family:* [config]

*Default:* False

---

### SERIES

Corresponds to the optional `-series` flag in `plot_TCMPR.R` (which is wrapped by `TCMPRPlotter`). This is the column whose unique values define the series on the plot, optionally followed by a comma-separated list of values, including: ALL, OTHER, and colon-separated groups.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

#### **SERIES\_ANALYSIS\_BY\_INIT\_CONFIG\_FILE**

Specify the absolute path for the configuration file to use with the MET series\_analysis tool by initialization time.

*Used by:* SeriesByInit

*Family:* [config]

*Default:* Varies

---

#### **SERIES\_ANALYSIS\_BY\_LEAD\_CONFIG\_FILE**

Specify the absolute path for the configuration file to use with the MET series\_analysis tool by lead time.

*Used by:* SeriesByLead

*Family:* [config]

*Default:* Varies

---

#### **SERIES\_ANALYSIS\_FILTER\_OPTS**

Filtering options to be applied during series analysis. Filter options are performed by invoking the MET tc\_stat tool within the METplus wrapper. Refer to Chapter 20 of the MET User's Guide for the syntax to use for performing filtering via the MET tc\_stat tool.

*Used by:* SeriesByLead, SeriesByInit

*Family:* [config]

*Default:* Varies

---

#### **SERIES\_BY\_LEAD\_FILTERED\_OUTPUT\_DIR**

Specifies the directory where filtered files will be written from the MET series\_analysis tool when processing by lead time.

*Used by:* SeriesByLead

*Family:* [config]

*Default:* Varies

---

#### **SERIES\_BY\_LEAD\_GROUP\_FCSTS**

Set to *True* to aggregate the series by lead results into bins of time.

*Used by:* SeriesByLead

*Family:* [config]

*Default:* Varies

---

#### **SERIES\_CI**

Corresponds to the optional -series\_ci flag in plot\_TCMPR.R (which is wrapped by TCMPRPlotter). This

is a list of true/false for confidence intervals. This list can be optionally followed by a comma-separated list of values, including ALL, OTHER, and colon-separated groups.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

#### **SERIES\_INIT\_FILTERED\_OUT\_DIR**

Specify the directory where filtered files will be written from the MET series\_analysis tool when processing by initialization time.

*Used by:* SeriesByInit

*Family:* [dir]

*Default:* Varies

---

#### **SERIES\_INIT\_OUT\_DIR**

Specify the directory where files will be written from the MET series analysis tool when processing by initialization time.

*Used by:* SeriesByInit

*Family:* [dir]

*Default:* Varies

---

#### **[deprecated] SERIES\_LEAD\_FILTERED\_OUT\_DIR**

Please use SERIES\_BY\_LEAD\_FILTERED\_OUTPUT\_DIR. Specify the directory where filtered files will be written from the MET series\_analysis tool when processing by lead time.

*Used by:* SeriesByLead

*Family:* [dir]

*Default:* Varies

---

#### **SERIES\_LEAD\_OUT\_DIR**

Specify the directory where files will be written from the MET series analysis tool when processing by lead time.

*Used by:* SeriesByLead

*Family:* [dir]

*Default:* Varies

---

#### **SKILL\_REF**

This corresponds to the optional -skill\_ref flag in plot\_TCMPR.R (which is wrapped by TCMPRPlotter). This is the identifier for the skill score reference.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

### **START\_DATE**

Specify the start data for the analysis time period. Format is YYYYMMDDHH.

*Used by:* PB2NC, PointStat

*Family:* [config]

*Default:* Varies

---

### **STAGING\_DIR**

Directory to uncompress or convert data into for use in METplus.

*Used by:* All

*Family:* [dir]

*Default:* OUTPUT\_BASE/stage

---

### **START\_HOUR**

Specify the start hour for the analysis time period. Format is HH.

*Used by:* PB2NC, PointStat

*Family:* [config]

*Default:* Varies

---

### **STAT\_ANALYSIS\_CONFIG**

Specify the absolute path for the configuration file used with the MET stat\_analysis tool. It is recommended to set this to {PARM\_BASE}/use\_cases/plotting/met\_config/STATAnalysisConfig.

*Used by:* StatAnalysis

*Family:* [config]

*Default:* Varies

---

### **STAT\_ANALYSIS\_DUMP\_ROW\_TMPL**

Specify the template to use for the stat\_analysis dump\_row file. A user customized template to use for the dump\_row file. If left blank and a dump\_row file is requested, a default version will be used. This is optional in the METplus configuration file for running with LOOP\_ORDER = times.

*Used by:* StatAnalysis

*Family:* [filename\_templates]

*Default:*

---

### **STAT\_ANALYSIS\_LOOKIN\_DIR**

Specify the input directory where the MET stat\_analysis tool will find input files. This is the directory that the stat\_analysis wrapper will use to build the argument to -lookin for the MET stat\_analysis tool. It can contain wildcards, i.e. \*.

*Used by:* StatAnalysis

*Family:* [dir]

*Default:* Varies

---

### **STAT\_ANALYSIS\_OUT\_STAT\_TMPL**

Specify the template to use for the stat\_analysis out\_stat file. A user customized template to use for the out\_stat file. If left blank and a out\_stat file is requested, a default version will be used. This is optional in the METplus configuration file for running with LOOP\_ORDER = times.

*Used by:* StatAnalysis

*Family:* [filename\_templates]

*Default:*

---

STAT\_ANAL

This is the base directory where the output from running stat\_analysis\_wrapper will be put.

*Used by:* StatAnalysis

*Family:* [dir]

*Default:* Varies

---

### **STAT\_FILES\_INPUT\_DIR**

Specify the directory where stat files exist that plots can be generated from. This is the directory where the files from running previously running stat\_analysis\_wrapper are located. These are the files used as the data to create the plots. It is recommended to set this to {STAT\_ANALYSIS\_OUTPUT\_DIR}.

*Used by:* MakePlots

*Family:* [dir]

*Default:* Varies

---

### **STAT\_LIST**

Specify a list of statistics to be computed by the MET series\_analysis tool.

*Used by:* SeriesByInit, SeriesByLead

*Family:* [config]

*Default:* Varies

---

### **[deprecated] STORM\_ID**

Please use TC\_PAIRS\_STORM\_ID or TC\_STAT\_STORM\_ID.

*Used by:* CyclonePlotter, TcPairs, TcStat

*Family:* [config]

*Default:* Varies

---

### **[deprecated] STORM\_NAME**

Please use TC\_PAIRS\_STORM\_NAME.

*Used by:* TcPairs

*Family:* [config]

*Default:* Varies

---

## **SUBTITLE**

The subtitle of the plot.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

## 4.6.20 T

---

### **TCMPR\_DATA\_DIR**

Provide the input directory for the track data for the TC Matched Pairs plotting tool.

*Used by:* TCMPRPlotter

*Family:* [dir]

*Default:* Varies

---

### **TCMPR\_PLOT\_OUT\_DIR**

Provide the output directory where the TC Matched Pairs plotting tool will create files.

*Used by:* TCMPRPlotter

*Family:* [dir]

*Default:* Varies

---

### **TC\_PAIRS\_ADECK\_INPUT\_DIR**

Directory that contains the ADECK files.

*Used by:* TcPairs

*Family:* [dir]

*Default:* Varies

---

### **TC\_PAIRS\_ADECK\_TEMPLATE**

Template of the file names of ADECK data.

*Used by:* TcPairs

*Family:* [filename\_templates]

*Default:* Varies

---

**TC\_PAIRS\_BASIN**

Control what basins are desired for tropical cyclone analysis.

Per the MET users' guide, acceptable basin ID's are:

WP = Western Northern Pacific

IO = Northern Indian Ocean

SH = Southern Hemisphere

CP = Central Northern Pacific

EP = Eastern Northern Pacific

AL = Northern Atlantic

SL = Southern Atlantic

*Used by:* TcPairs

*Family:* [config]

*Default:* Varies

---

**TC\_PAIRS\_BDECK\_INPUT\_DIR**

Directory that contains the BDECK files.

*Used by:* TcPairs

*Family:* [dir]

*Default:* Varies

---

**TC\_PAIRS\_BDECK\_TEMPLATE**

Template of the file names of BDECK data.

*Used by:* TcPairs

*Family:* [filename\_templates]

*Default:* Varies

---

**TC\_PAIRS\_CONFIG\_FILE**

Provide the absolute path to the configuration file for the MET tc\_pairs tool.

*Used by:* TcPairs

*Family:* [config]

*Default:* Varies

---

**TC\_PAIRS\_CYCLONE**

Specify which cyclone numbers to include in the tropical cyclone analysis. Per the MET users' guide, this can be any number 01-99 (HH format). Use a space or comma separated list, or leave unset if all cyclones are desired.

*Used by:* TcPairs

*Family:* [config]

*Default:* Varies

---

The file generated by the MET tool `tc_dland`, containing the gridded representation of the minimum distance to land. Please refer to Chapter 18 of the MET User's Guide for more information about the `tc_dland` tool.

*Used by:* TcPairs

*Family:* [config]

*Default:* Varies

---

#### **TC\_PAIRS\_EDECK\_INPUT\_DIR**

Directory that contains the EDECK files.

*Used by:* TcPairs

*Family:* [dir]

*Default:* Varies

---

#### **TC\_PAIRS\_EDECK\_TEMPLATE**

Template of the file names of EDECK data.

*Used by:* TcPairs

*Family:* [filename\_templates]

*Default:* Varies

---

#### **[deprecated] TC\_PAIRS\_DIR**

Please use `TC_PAIRS_OUTPUT_DIR`.

*Used by:* TcPairs

*Family:* [dir]

*Default:* Varies

---

#### **[deprecated] TC\_PAIRS\_FORCE\_OVERWRITE**

Please use `TC_PAIRS_SKIP_IF_OUTPUT_EXISTS`.

Acceptable values: yes/no

*Used by:* TcPairs

*Family:* [config]

*Default:* no

---

#### **TC\_PAIRS\_MISSING\_VAL**

Specify the missing value code.

*Used by:* TcPairs

*Family:* [config]

*Default:* Varies

---

#### **TC\_PAIRS\_MISSING\_VAL\_TO\_REPLACE**

Specify the missing value code to replace.

*Used by:* TcPairs

*Family:* [config]

*Default:* Varies

---

#### **TC\_PAIRS\_OUTPUT\_DIR**

Specify the directory where the MET tc\_pairs tool will write files.

*Used by:* TcPairs

*Family:* [dir]

*Default:* Varies

---

#### **TC\_PAIRS\_OUTPUT\_TEMPLATE**

Template of the output file names created by tc\_pairs.

*Used by:* TcPairs

*Family:* [filename\_templates]

*Default:* Varies

---

#### **TC\_PAIRS\_READ\_ALL\_FILES**

Specify whether to pass the value specified in TC\_PAIRS\_[A/B/E]DECK\_INPUT\_DIR to the MET tc\_pairs utility or have the wrapper search for valid files in that directory based on the value of TC\_PAIRS\_[A/B/E]DECK\_T and pass them individually to tc\_pairs. Set to false or no to have the wrapper find valid files. This can speed up execution time of tc\_pairs.

Acceptable values: yes/no

*Used by:* TcPairs

*Family:* [config]

*Default:* no

---

#### **TC\_PAIRS\_REFORMAT\_DECK**

Set to true or yes if using cyclone data that needs to be reformatted to match the ATCF (Automated Tropical Cyclone Forecasting) format. If set to true or yes, you will need to set TC\_PAIRS\_REFORMAT\_TYPE to specify which type of reformatting to perform.

*Used by:* TcPairs

*Family:* [config]

*Default:* Varies

---

#### **TC\_PAIRS\_REFORMAT\_DIR**

Specify the directory to write reformatted track data to be read by `tc_pairs`. Used only if `TC_PAIRS_REFORMAT_DECK` is true or yes.

**Used by:** TcPairs

**Family:** [dir]

**Default:** `{OUTPUT_BASE}/track_data_atcf`

---

#### **TC\_PAIRS\_REFORMAT\_TYPE**

Specify which type of reformatting to perform on cyclone data. Currently only SBU extra tropical cyclone reformatting is available. Only used if `TC_PAIRS_REFORMAT_DECK` is true or yes.

Acceptable values: SBU

**Used by:** TcPairs

**Family:** [config]

**Default:** Varies

---

#### **TC\_PAIRS\_SKIP\_IF\_REFORMAT\_EXISTS**

Specify whether to overwrite the reformatted cyclone data or not. If set to true or yes and the reformatted file already exists for a given run, the reformatting code will not be run. Used only when `TC_PAIRS_REFORMAT_DECK` is set to true or yes.

Acceptable values: yes/no

**Used by:** TcPairs

**Family:** [config]

**Default:** no

---

#### **TC\_PAIRS\_SKIP\_IF\_OUTPUT\_EXISTS**

Specify whether to overwrite the output from the MET `tc_pairs` tool or not. If set to true or yes and the output file already exists for a given run, `tc_pairs` will not be run.

Acceptable values: yes/no

**Used by:** TcPairs

**Family:** [config]

**Default:** no

---

#### **TC\_PAIRS\_STORM\_ID**

The identifier of the storm(s) of interest.

**Used by:** TcPairs

**Family:** [config]

**Default:** Varies

---

**TC\_PAIRS\_STORM\_NAME**

The name(s) of the storm of interest.

*Used by:* TcPairs

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_AMODEL**

Specify the AMODEL for the MET tc\_stat tool.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_BASIN**

Specify the BASIN for the MET tc\_stat tool.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_BMODEL**

Specify the BMODEL for the MET tc\_stat tool.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_CMD\_LINE\_JOB**

Specify expression(s) that will be passed to the MET tc\_stat tool via the command line. Only specify if TC\_STAT\_RUN\_VIA=CLI. Please refer to the MET User's Guide chapter for tc-stat for the details on performing job summaries and job filters.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_COLUMN\_STR\_NAME**

Specify the string names of the columns for stratification with the MET tc\_stat tool.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_COLUMN\_STR\_VAL**

Specify the values for the columns set via the TC\_STAT\_COLUMN\_STR\_NAME option for use with the MET tc\_stat tool.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_COLUMN\_THRESH\_NAME**

Specify the string names of the columns for stratification by threshold with the MET tc\_stat tool.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_COLUMN\_THRESH\_VAL**

Specify the values used for thresholding the columns specified in the TC\_STAT\_COLUMN\_THRESH\_NAME option for use with the MET tc\_stat tool.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_CYCLONE**

Specify the CYCLONE of interest for use with the MET tc\_stat tool.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_DESC**

Specify the DESC option for use with the MET tc\_stat tool.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_INIT\_BEG**

Specify the beginning initialization time for stratification when using the MET tc\_stat tool.

Acceptable formats: YYYYMMDD\_HH, YYYYMMDD\_HHmss

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_INIT\_END**

Specify the ending initialization time for stratification when using the MET `tc_stat` tool.

Acceptable formats: `YYYYMMDD_HH`, `YYYYMMDD_HH:mm:ss`

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_INIT\_EXCLUDE**

Specify the initialization times to exclude when using the MET `tc_stat` tool, via a comma separated list e.g.:

`20141220_18`, `20141221_00`

Acceptable formats: `YYYYMMDD_HH`, `YYYYMMDD_HH:mm:ss`

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_INIT\_HOUR**

The beginning hour (HH) of the initialization time of interest.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_INIT\_INCLUDE**

Specify the initialization times to include when using the MET `tc_stat` tool, via a comma separated list e.g.:

`20141220_00`, `20141220_06`, `20141220_12`

Acceptable formats: `YYYYMMDD_HH`, `YYYYMMDD_HH:mm:ss`

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

**TC\_STAT\_INIT\_MASK**

This corresponds to the `INIT_MASK` keyword in the MET `tc_stat` config file. For more information, please refer to Chapter 20 in the MET User's Guide.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_INIT\_STR\_NAME**

This corresponds to the INIT\_STR\_NAME keyword in the MET tc\_stat config file. Please refer to Chapter 20 in the MET User's Guide for more details.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_INIT\_STR\_VAL**

This corresponds to the INIT\_STR\_VAL keyword in the MET tc\_stat config file. Please refer to Chapter 20 in the MET User's Guide for more information.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_INPUT\_DIR**

Specify the input directory where the MET tc\_stat tool will look for files.

*Used by:* TcStat

*Family:* [dir]

*Default:* Varies

---

#### **TC\_STAT\_JOBS\_LIST**

Specify expressions for the MET tc\_stat tool to execute.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_LANDFALL**

Specify whether only those points occurring near landfall should be retained when using the MET tc\_stat tool.

Acceptable values: True/False

*Used by:* TcStat

*Family:* [config]

*Default:* False

---

#### **TC\_STAT\_LANDFALL\_BEG**

Specify the beginning of the landfall window for use with the MET tc\_stat tool.

Acceptable formats: HH, HHmmss

*Used by:* TcStat

*Family:* [config]

*Default:* -24

---

#### **TC\_STAT\_LANDFALL\_END**

Specify the end of the landfall window for use with the MET tc\_stat tool.

Acceptable formats: HH, HHmmss

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_LEAD**

Specify the lead times to stratify by when using the MET tc\_stat tool.

Acceptable formats: HH, HHmmss

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_LEAD\_REQ**

Specify the LEAD\_REQ when using the MET tc\_stat tool.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_MATCH\_POINTS**

Specify whether only those points common to both the ADECK and BDECK tracks should be written out or not when using the MET tc\_stat tool.

Acceptable values: True/False

*Used by:* TcStat

*Family:* [config]

*Default:* false

---

#### **TC\_STAT\_OUTPUT\_DIR**

Specify the output directory where the MET tc\_stat tool will write files.

*Used by:* TcStat

*Family:* [dir]

*Default:* Varies

---

#### **TC\_STAT\_RUN\_VIA**

Specify the method for running the MET tc\_stat tool.

Acceptable values: CONFIG

If left blank (unset), tc\_stat will run via the command line.

*Used by:* TcStat

*Family:* [config]

*Default:* CONFIG

---

#### **TC\_STAT\_STORM\_ID**

Set the STORM\_ID(s) of interest with the MET tc\_stat tool.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_STORM\_NAME**

Set the STORM\_NAME for use with the MET tc\_stat tool.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_TRACK\_WATCH\_WARN**

Specify which watches and warnings to stratify over when using the MET tc\_stat tool.

Acceptable values: HUWARN, HUWATCH, TSWARN, TSWATCH, ALL

If left blank (unset), no stratification will be done.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_VALID\_BEG**

Specify a comma separated list of beginning valid times to stratify with when using the MET tc\_stat tool.

Acceptable formats: YYYYMMDD\_HH, YYYYMMDD\_HHmms

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_VALID\_END**

Specify a comma separated list of ending valid times to stratify with when using the MET tc\_stat tool.

Acceptable formats: YYYYMMDD\_HH, YYYYMMDD\_HHmss

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_VALID\_EXCLUDE**

Specify a comma separated list of valid times to exclude from the stratification with when using the MET tc\_stat tool.

Acceptable formats: YYYYMMDD\_HH, YYYYMMDD\_HHmss

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_VALID\_HOUR**

This corresponds to the VALID\_HOUR keyword in the MET tc\_stat config file. For more information, please refer to Chapter 20 of the MET User's Guide.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_VALID\_INCLUDE**

Specify a comma separated list of valid times to include in the stratification with when using the MET tc\_stat tool.

Acceptable formats: YYYYMMDD\_HH, YYYYMMDD\_HHmss

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

#### **TC\_STAT\_VALID\_MASK**

This corresponds to the VALID\_MASK in the MET tc\_stat config file. Please refer to Chapter 20 of the MET User's Guide for more information.

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

### **TC\_STAT\_WATER\_ONLY**

Specify whether to exclude points where the distance to land is  $\leq 0$ . If set to TRUE, once land is encountered the remainder of the forecast track is not used for the verification, even if the track moves back over water.

Acceptable values: true/false

*Used by:* TcStat

*Family:* [config]

*Default:* Varies

---

### **TIME\_METHOD**

Specify which time method to use with the MET pb2nc and point\_stat tools.

Acceptable values: BY\_VALID, BY\_INIT

*Used by:* PB2NC, PointStat

*Family:*

*Default:*

---

### **[deprecated] TIME\_SUMMARY\_BEG**

Please use PB2NC\_TIME\_SUMMARY\_BEG instead. Specify the starting time of the summary when using the MET pb2nc tool.

Acceptable formats: HHMMSS

*Used by:* PB2NC

*Family:* [config]

*Default:* 000000

---

### **[deprecated] TIME\_SUMMARY\_END**

Please use PB2NC\_TIME\_SUMMARY\_END instead. Specify the ending time of the summary when using the MET pb2nc tool.

Acceptable formats: HHMMSS

*Used by:* PB2NC

*Family:* [config]

*Default:* 235959

---

### **[deprecated] TIME\_SUMMARY\_FLAG**

Please use `PB2NC_TIME_SUMMARY_FLAG` instead. Specify whether to receive a time summary from the MET `pb2nc` tool or not.

Acceptable values: True/False

*Used by:* PB2NC

*Family:* [config]

*Default:* False

---

#### [deprecated] `TIME_SUMMARY_TYPES`

Please use `PB2NC_TIME_SUMMARY_TYPES` instead. Specify a comma separated list of time summary types to receive from the MET `pb2nc` tool.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### [deprecated] `TIME_SUMMARY_VAR_NAMES`

Please use `PB2NC_TIME_SUMMARY_VAR_NAMES` instead. Specify a comma separated list of time summary variable names to receive from the MET `pb2nc` tool.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### `TITLE`

Specify a title string for the TC Matched Pairs plotting tool.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

#### `TMP_DIR`

Specify the path to a temporary directory where the user has write permissions.

*Used by:* ExtractTiles, PB2NC, PointStat, SeriesByInit, SeriesByLead, TcStat

*Family:* [dir]

*Default:* Varies

---

#### [deprecated] `TOP_LEVEL_DIRS`

Please use `TC_PAIRS_READ_ALL_FILES`.

*Used by:* TcPairs

*Family:* [config]

*Default:* no

---

**[deprecated] TRACK\_DATA\_DIR**

Please use TC\_PAIRS\_[A/B/E]DECK\_INPUT\_DIR.

*Used by:* TcPairs

*Family:* [dir]

*Default:* Varies

---

**[deprecated] TRACK\_DATA\_MOD\_FORCE\_OVERWRITE**

Please use TC\_PAIRS\_SKIP\_IF\_REFORMAT\_EXISTS.

*Used by:* TcPairs

*Family:* [config]

*Default:* no

---

**[deprecated] TRACK\_DATA\_SUBDIR\_MOD**

No longer used.

*Used by:* TcPairs

*Family:* [dir]

*Default:* Varies

---

**[deprecated] TRACK\_TYPE**

Please use TC\_PAIRS\_REFORMAT\_DECK.

*Used by:* TcPairs

*Family:* [config]

*Default:* Varies

---

**TR**

Specify the path to the Linux “tr” executable.

*Used by:* PB2NC, PointStat

*Family:* [exe]

*Default:* /path/to

---

**[deprecated] TR\_EXE**

Please use TR.

*Used by:* PB2NC, PointStat

*Family:* [exe]

*Default:* /path/to

---

#### 4.6.21 U

#### 4.6.22 V

---

##### **VALID\_BEG**

Specify a begin time for valid times for use in the analysis. This is the starting date in the format set in the VALID\_TIME\_FMT. It is named accordingly to the value set for LOOP\_BY. However, in StatAnalysis, it is named accordingly to the value set for PLOT\_TIME.

See [4.3.1.2](#) for more information.

*Used by:* All

*Family:* [config]

*Default:* Varies

---

##### **VALID\_END**

Specify an end time for valid times for use in the analysis. This is the ending date in the format set in the VALID\_TIME\_FMT. It is named accordingly to the value set for LOOP\_BY.

See [4.3.1.2](#) for more information.

*Used by:* All

*Family:* [config]

*Default:* Varies

---

##### **VALID\_HOUR\_BEG**

Specify a beginning hour for valid times for use in the analysis.

Acceptable formats: HHMM

*Used by:* MakePlots, StatAnalysis

*Family:* [config]

*Default:* Varies

---

##### **VALID\_HOUR\_END**

Specify an end hour for valid times for use in the analysis.

Acceptable formats: HHMM

*Used by:* MakePlots, StatAnalysis

*Family:* [config]

*Default:* Varies

---

**VALID\_HOUR\_INCREMENT**

Specify a time increment for valid times for use in the analysis.

Acceptable formats: seconds

*Used by:* MakePlots, StatAnalysis

*Family:* [config]

*Default:* Varies

---

**VALID\_HOUR\_METHOD**

Specify the method for the treatment of valid hours. Valid options are LOOP or GROUP. LOOP will consider the valid hours individually, and GROUP will consider them valid hours as a whole.

Acceptable formats: LOOP or GROUP

*Used by:* MakePlots, StatAnalysis

*Family:* [config]

*Default:* Varies

---

**VALID\_INCREMENT**

Specify the time increment for valid times for use in the analysis.

See [4.3.1.2](#) for more information.

*Used by:* All

*Family:* [config]

*Default:* Varies

---

**VALID\_TIME\_FMT**

Specify a strftime formatting string for use with VALID\_BEG and VALID\_END.

See [4.3.1.2](#) for more information.

*Used by:* All

*Family:* [config]

*Default:* Varies

---

**VAR\_LIST**

Specify a comma separated list of variables to be used in the analysis.

*Used by:* PB2NC, SeriesByInit, SeriesByLead

*Family:* [config]

*Default:* Varies

---

**VAR<n>\_FOURIER\_DECOMP**

Specify if Fourier decomposition is to be considered (True) or not (False). If this is set to True, data stratification will be done for the Fourier decomposition of FCS\_VAR<n>\_NAME. This should have been previously run in grid\_stat\_wrapper. The default value is set to False.

*Used by:* MakePlots, StatAnalysis

*Family:* [config]

*Default:* False

---

**VAR<n>\_WAVE\_NUM\_LIST**

Specify a comma separated list of wave numbers pairings of the Fourier decomposition.

*Used by:* MakePlots, StatAnalysis

*Family:* [config]

*Default:*

---

**[deprecated] VERIFICATION\_GRID**

Please use REGRID\_DATA\_PLANE\_VERIF\_GRID. Specify the absolute path to a file containing information about the desired output grid from the MET regrid\_data\_plane tool.

*Used by:* RegridDataPlane

*Family:* [config]

*Default:* Varies

---

**VERIF\_CASE**

Specify a string identifying the verification case being performed. Valid options are grid2grid, grid2obs, and precip.

*Used by:* MakePlots, StatAnalysis

*Family:* [config]

*Default:* Varies

---

**VERIF\_GRID**

Specify a string describing the grid the verification was performed on. This is the name of the grid upon which the verification was done on, ex. G002.

*Used by:* MakePlots

*Family:* [config]

*Default:*

---

Specify a string describing the type of verification being performed. For VERIF\_CASE = grid2grid, valid options are anom, pres, and sfc. For VERIF\_CASE = grid2obs, valid options are conus\_sfc and upper\_air. For VERIF\_CASE = precip, any accumulation amount is valid, ex. A24.

---

**VERIF\_TYP**

*Used by:* MakePlots, StatAnalysis

*Family:* [config]

*Default:* Varies

---

#### [deprecated] VERTICAL\_LOCATION

Specify the vertical location desired when using the MET pb2nc tool.

*Used by:* PB2NC

*Family:* [config]

*Default:* Varies

---

#### 4.6.23 W

---

##### WGRIB2

Specify the path to the “wgrib2” executable.

*Used by:* PB2NC, PointStat

*Family:* [exe]

*Default:* /path/to

---

#### 4.6.24 X

---

##### XLAB

Specify the x-axis label when using the TC Matched Pairs plotting tool.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

##### XLIM

Specify the x-axis limit when using the TC Matched Pairs plotting tool.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

#### 4.6.25 Y

---

**YLAB**

Specify the y-axis label when using the TC Matched Pairs plotting tool.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

**YLIM**

Specify the y-axis limit when using the TC Matched Pairs plotting tool.

*Used by:* TCMPRPlotter

*Family:* [config]

*Default:* Varies

---

#### 4.6.26 Z

# References

- Alberson, S.D., 1998: Five-day Tropical cyclone track forecasts in the North Atlantic Basin. *Weather & Forecasting*, 13, 1005-1015.
- Bradley, A.A., S.S. Schwartz, and T. Hashino, 2008: Sampling Uncertainty and Confidence Intervals for the Brier Score and Brier Skill Score. *Weather and Forecasting*, 23, 992-1006.
- Brill, K. F., and F. Mesinger, 2009: Applying a general analytic method for assessing bias sensitivity to bias-adjusted threat and equitable threat scores. *Weather and Forecasting*, 24, 1748–1754.
- Brown, B.G., R. Bullock, J. Halley Gotway, D. Ahijevych, C. Davis, E. Gilleland, and L. Holland, 2007: Application of the MODE object-based verification tool for the evaluation of model precipitation fields. *AMS 22nd Conference on Weather Analysis and Forecasting and 18th Conference on Numerical Weather Prediction*, 25-29 June, Park City, Utah, American Meteorological Society (Boston), Available at <http://ams.confex.com/ams/pdfpapers/124856.pdf>.
- Bullock, R., T. Fowler, and B. Brown, 2016: Method for Object-Based Diagnostic Evaluation. NCAR Tech. Note NCAR/TN-532+STR, 66 pp.
- Candille, G., and O. Talagrand, 2008: Impact of observational error on the validation of ensemble prediction systems. *Q. J. R. Meteorol. Soc.* 134: 959–971.
- Casati, B., G. Ross, and D. Stephenson, 2004: A new intensity-scale approach for the verification of spatial precipitation forecasts. *Meteorol. Appl.* 11, 141-154.
- Davis, C.A., B.G. Brown, and R.G. Bullock, 2006a: Object-based verification of precipitation forecasts, Part I: Methodology and application to mesoscale rain areas. *Monthly Weather Review*, 134, 1772-1784.
- Davis, C.A., B.G. Brown, and R.G. Bullock, 2006b: Object-based verification of precipitation forecasts, Part II: Application to convective rain systems. *Monthly Weather Review*, 134, 1785-1795.
- Dawid, A.P., 1984: Statistical theory: The prequential approach. *J. Roy. Stat. Soc.* A147, 278-292.
- Ebert, E.E., 2008: Fuzzy verification of high-resolution gridded forecasts: a review and proposed framework. *Meteorological Applications*, 15, 51-64.
- Eckel, F. A., M.S. Allen, M. C. Sittel, 2012: Estimation of Ambiguity in Ensemble Forecasts. *Wea. Forecasting*, 27, 50-69. doi: <http://dx.doi.org/10.1175/WAF-D-11-00015.1>

- Efron, B. 2007: Correlation and large-scale significance testing. *Journal of the American Statistical Association*, 102(477), 93-103.
- Gilleland, E., 2010: Confidence intervals for forecast verification. *NCAR Technical Note* NCAR/TN-479+STR, 71pp.
- Gneiting, T., A. Westveld, A. Raferty, and T. Goldman, 2004: *Calibrated Probabilistic Forecasting Using Ensemble Model Output Statistics and Minimum CRPS Estimation*. Technical Report no. 449, Department of Statistics, University of Washington. [Available online at <http://www.stat.washington.edu/www/research/reports/> ]
- Hamill, T. M., 2001: Interpretation of rank histograms for verifying ensemble forecasts. *Mon. Wea. Rev.*, 129, 550-560.
- Hogan, R., E. O'Connor, and A. Illingworth, 2009: Verification of cloud-fraction forecasts. *Quart. Jour. Roy. Meteorol. Soc.*, 135, 1494-1511.
- Jolliffe, I.T., and D.B. Stephenson, 2012: *Forecast verification. A practitioner's guide in atmospheric science*. Wiley and Sons Ltd, 240 pp.
- Knaff, J.A., M. DeMaria, C.R. Sampson, and J.M. Gross, 2003: Statistical, Five-Day Tropical Cyclone Intensity Forecasts Derived from Climatology and Persistence." *Weather & Forecasting*," Vol. 18 Issue 2, p. 80-92.
- Mason, S. J., 2004: On Using "Climatology" as a Reference Strategy in the Brier and Ranked Probability Skill Scores. *Mon. Wea. Rev.*, 132, 1891-1895.
- Mittermaier, M., 2013: A strategy for verifying near-convection-resolving model forecasts at observing sites. *Wea. Forecasting*, 29, 185-204.
- Mood, A. M., F. A. Graybill and D. C. Boes, 1974: *Introduction to the Theory of Statistics*, McGraw-Hill, 299-338.
- Murphy, A.H., and R.L. Winkler, 1987: A general framework for forecast verification. *Monthly Weather Review*, 115, 1330-1338.
- Roberts, N.M., and H.W. Lean, 2008: Scale-selective verification of rainfall accumulations from high-resolution forecasts of convective events. *Monthly Weather Review*, 136, 78-97.
- Saetra O., H. Hersbach, J-R Bidlot, D. Richardson, 2004: Effects of observation errors on the statistics for ensemble spread and reliability. *Mon. Weather Rev.* 132: 1487-1501.
- Santos C. and A. Ghelli, 2012: Observational probability method to assess ensemble precipitation forecasts. *Q. J. R. Meteorol. Soc.* 138: 209-221.
- Stephenson, D.B., 2000: Use of the "Odds Ratio" for diagnosing forecast skill. *Weather and Forecasting*, 15, 221-232.
- Stephenson, D.B., B. Casati, C.A.T. Ferro, and C.A. Wilson, 2008: The extreme dependency score: A non-vanishing measure for forecasts of rare events. *Meteor. Appl.* 15, 41-50.

- Weniger, M., F. Kapp, and P. Friederichs, 2016: Spatial Verification Using Wavelet Transforms: A Review. *Quarterly Journal of the Royal Meteorological Society*, 143, 120-136.
- Wilks, D.S. 2010: Sampling distributions of the Brier score and Brier skill score under serial dependence. *Q.J.R. Meteorol. Soc.*, 136, 2109–2118. doi:10.1002/qj.709
- Wilks, D., 2011: *Statistical methods in the atmospheric sciences*. Elsevier, San Diego.

# List of Tables

# List of Figures