

# Land surface Verification Toolkit (LVT)

## ***LVT 7.3 Users' Guide***

Version 1.8, 31 Mar 2021

# Table of Contents

1. Introduction	2
1.1. What's New	2
2. Background	7
2.1. LVT	7
2.2. Summary of key features	7
3. Preliminary Information	9
4. Obtaining the Source Code	10
4.1. Important Note Regarding File Systems	10
4.2. Public Release Source Code	10
4.3. master branch	11
4.4. Source files	11
4.5. Documentation	12
5. Building the Executable	13
5.1. Development Tools	13
5.2. Required Software Libraries	14
5.3. Optional Software Libraries	14
5.4. Build Instructions	16
6. Running the Executable	20
7. LVT config File	21
7.1. Overall driver options	21
7.2. Domain specification	28
7.3. Analysis options specification	29
7.4. Data stream sources	40
7.5. OptUE processing options	70
7.6. DA diagnostics analysis	72
7.7. DA observation analysis	73
7.8. 557 post	73
7.9. USAFSI post	75
8. Configuration of metrics	77
9. Model Output Specifications	81
Appendix A: How to verify a “non-LIS” dataset?	87
Appendix B: Description of output files from LVT	88
B.1. METADATA files	88
B.2. Stats summary file	89
B.3. ASCII Time Series files	89
B.4. Domain time Series files (Binary/NETCDF)	90
B.5. Final domain files (Binary/NETCDF)	91
References	92

<b>Revision</b>	<b>Summary of Changes</b>	<b>Date</b>
1.8	LISF Public 7.3.1 release	Mar 31, 2021
1.6	LISF Public 7.3.0 release	Dec 21, 2020
1.5	LVT 557WW 7.3.0 release	Aug 24, 2020
1.4	LVT 557WW 7.3 release candidate 4	Aug 28, 2019
1.3	LVT 7.2 public release	May 6, 2015
1.2	Update lvt.config documentation	May 12, 2015
1.1	LVT 7.1 AFWA rp2	May 11, 2015
1.0	LVT 7.1 public release	April 24, 2015

# Chapter 1. Introduction

This is the User's Guide for Land surface Verification Toolkit (LVT;[[kumar\\_etal\\_gmd\\_2012](#)]). This document describes how to download and install LVT codes and instructions on building an executable.

This document consists of 9 sections, described as follows:

1. **Introduction**: the section you are currently reading.
2. **Background**: general information about the LVT.
3. **Preliminary Information**: general information, steps, instructions, and definitions used throughout the rest of this document.
4. **Obtaining the Source Code**: the steps needed to download the source code.
5. **Building the Executable**: the steps needed to build the LVT executable.
6. **Running the Executable**: the steps needed to run the LVT executable.
7. **LVT config File**: describes the user-configurable options.
8. **Configuration of metrics**: describes the specification of various metrics in LVT.
9. **Model Output Specifications**: describes the specification of the model output from LIS.

## 1.1. What's New

### 1.1.1. Version 7.3

1. Includes new runmodes
  - 557post runmode
  - USAFSIpost runmode
2. Includes additional metrics
  - Anomaly Rank Correlation metric
  - Clayton Skill Score (CSS) metric
  - Conditional Entropy metric
  - Detection Failure Ratio (DFR) metric
  - Event Frequency (EF) metric
  - Forecast Frequency (FF) metric
  - Heidke Skill Score (HSS) metric
  - Information Entropy metric
  - Joint Entropy metric
  - K-means metric
  - Mutual Information metric

- Peirce Skill Score (PSS) metric
- Reliability metric
- Resilience metric
- Relative Entropy metric
- Standard Precipitation Index (SPI) metric
- Triple collocation metric
- Tian False Alarm Bias (TFB) metric
- Tian Hit Bias (THB) metric
- Tian Miss Bias (TMB) metric
- Trends metric
- Vulnerability metric

### 3. Supports additional data streams

- TRMM 3B42V7 data stream
- Atmospheric Land Exchange Inverse (ALEXI) model Evaporative Stress Index (ESI) data stream
- APHRODITE (Asian Precipitation - Highly-Resolved Observational Data Integration Towards Evaluation) daily gridded precipitation data stream
- NASA Airborne Snow Observatory (ASO) snow water equivalent (SWE) data stream
- CHIRPSv2 data stream
- ECMWF forcing data stream
- ERA5 data stream
- FLUXCOM data stream
- FLUXNET2015\_NC data stream
- GDAS forcing data stream
- GLASS albedo data stream
- GLASS LAI data stream
- High Asia Reanalysis (HAR) precipitation data stream
- Indian Meteorological Department (IMD) unified gauge-based analysis of daily precipitation data stream
- IMERG data stream
- JULES2Ddata data stream
- Land Parameter Retrieval Model (LPRM) vegetation optical depth data stream
- MERRA2 asm data stream
- MOD10A1V6 fractional snow cover data stream
- MODIS SPoRT LAI data stream

- MODSCAG fractional snow cover data stream
- OCO-2 Solar Induced Fluorescence (SIF) data stream
- OzFlux data stream
- Soil Moisture Active Passive (SMAP) L3 Tb data stream
- Soil Moisture Active Passive (SMAP) vegetation optical depth data stream
- Soil Moisture Active Passive (SMAP) vegetation water content data stream
- University of Arizona (UA) SWE/Snow Depth data stream
- UCLA SWE reanalysis data stream
- jasmin data stream

### **1.1.2. Version 7.2**

#### **1. Supports new data streams**

- datastreams/CMORPH
- datastreams/Daymet
- datastreams/FLUXNETmte
- datastreams/GCOMW\_AMSR2L3snd
- datastreams/GIMMS\_MODIS\_NDVI
- datastreams/GLDAS1
- datastreams/GOES\_LST
- datastreams/GOME2\_SIF
- datastreams/JULESdata
- datastreams/LIS6out
- datastreams/LISDAdiag
- datastreams/LVTpercentile
- datastreams/MERRA-Land
- datastreams/SCANGMAO
- datastreams/simGRACE
- datastreams/SMAPsm
- datastreams/SMAPTb
- datastreams/SMOS\_CATDS\_L3sm
- datastreams/SMOS\_NESDIS
- datastreams/SNODEP
- datastreams/USCRNsm
- datastreams/USDM
- datastreams/USGS\_streamflow\_gridded

### 1.1.3. Version 7.1

1. Supports Global Change Observation Mission - Water (GCOMW) observations
2. Supports Advanced Scatterometer (ASCAT) data
3. Supports Soil Moisture and Ocean Salinity (SMOS) observations
4. Supports Soil Moisture and Ocean Salinity (SMOS) L1 Tb observations
5. Supports MODIS LST data
6. Supports Great Lakes Hydro data
7. Supports time lagged computations

### 1.1.4. Version 7.0

Note that LVT has been renumbered to version 7.0.

1. Supports NLDAS-2 SAC datasets, including the post-processed SAC soil moisture on Noah levels
2. Supports the capability to compute metrics on a basin/region scale rather than on a pixel-by-pixel basis.
3. Supports the capability to process USGS ground water well data
4. Supports the capability to process Plate Boundary Observatory data (snow depth and soil moisture)
5. Supports the North American Soil Moisture Database (NASMD) data
6. Supports the ALEXI data
7. Supports the metric to compute percentiles for each ensemble member separately
8. Supports GRACE observations (these are the “processed” GRACE observations generated by LDT)
9. Supports the UW ET data
10. Supports the options for restart for SRI, SPI, SSWI and percentile calculations
11. Supports metric called SGWI - standardized ground water index — which is similar to SSWI, but for TWS variable
12. Support for a number of ratio variables are added:
  - SWE/P (snow water equivalent over precip)
  - ET/P (evapotranspiration over precip)
  - Qs/P (surface runoff over precip)
  - Qsb/P (subsurface runoff over precip)
13. Supports the river flow variate metric
14. Supports metrics for computing min, max and sum
15. Support to read LIS DA processed observations

### **1.1.5. Version 1.2**

1. Updates related to OptUE processing

### **1.1.6. Version 1.1**

1. Added the plugin for processing processed LIS DA observations
2. Removed the FEWSNET SOS processing metric
3. Added plugins for NLDAS2, Natural streamflow
4. Updated to provide backwards support for LIS 6.1.

### **1.1.7. Version 1.0**

1. This is the initial version developed for evaluating output from LIS version 6.0 or higher.



# Chapter 2. Background

Verification and evaluation are essential processes in the development and application of simulation models. The Land surface Verification Toolkit (LVT) is an integrated framework designed specifically for evaluating land surface model (LSM) outputs. The system was originally designed as a post processor to the NASA Land Information System (LIS), which is an integrated framework to conduct multi-model land surface model simulations and data assimilation integrations. LVT also includes the capabilities to convert any land surface-specific dataset to a “LIS output format/style”, thus enabling cross-comparisons of a broad set of land surface datasets (in-situ, remotely sensed, and reanalysis products).

## 2.1. LVT

LVT provides a formal system for LSM output evaluation and verification. The capabilities of LVT also provides a tool to systematically evaluate and benchmark LSM performance and the impact of computational enhancements such as data assimilation. LVT includes a range of both deterministic and probabilistic verification measures, with similarity-based and object-based methods in development.

LVT is designed as an object oriented framework, with a number of points of flexibility known as “plugins”. Specific implementations are added to the framework through the plugin-interfaces. LVT uses the plugin-based architecture to support the processing of different types of observational data sets, ranging from in-situ, satellite and remotely sensed and reanalysis products.

In addition to providing methods for model output verification, LVT also provides capabilities to analyze the outputs from LIS data assimilation (LIS-DA) and the LIS optimization and uncertainty estimation (LIS-OPTUE) subsystems.

## 2.2. Summary of key features

The key capabilities of LVT can be summarized as follows:

- Capability to convert a given dataset to “LIS-style format”
- A text-based, configurable input interface
- Supports a broad range of in-situ, remotely-sensed and reanalysis data products. For e.g.:
  - Surface fluxes - Ameriflux, ARM, CEOP, AMMA
  - Soil moisture - SCAN, SMOSREX, AMSR-E retrievals
  - Snow - COOP, GSOD, SNODAS, SNODEP, CMC, FMI, GlobSnow, SNOTEL
  - LST - ISCCP
  - Radiation - SURFRAD
- A number of deterministic and probabilistic verification measures. E.g. RMSE, Bias, Correlations, POD, FAR, etc.
- Supports the computation of land surface model diagnostics and closure checks. E.g.: Energy, water balance checks, seasonal and average diurnal cycles

- Options of time series extraction of individual grid points and area averages
- Options of temporal averaging. E.g. : Comparisons at hourly, daily, monthly scales
- Options for data masking. E.g. Apply an external mask to the comparisons, apply thresholds on the comparisons
- Supports the analysis of outputs from LIS-OPTUE and LIS-DA subsystems. For e.g: Analysis of normalized innovations from LIS-DA

# Chapter 3. Preliminary Information

This section provides some preliminary information to make reading this guide easier.

Commands are written with a fixed-width font. E.g.:

```
% cd /path/to/LVT
```

```
% ls
```

“... compiler flags, then run `gmake`.”

## NOTE

The `%` symbol represents the command-line prompt. You do **not** type that when entering any of the commands.

File names are written in italics. E.g.:

```
/path/to/LVT/src
```

# Chapter 4. Obtaining the Source Code

This section describes how to obtain the source code needed to build the LVT executable.

Beginning with Land Information System Framework (LISF) public release 7.3, the LVT source code is available as open source under the Apache License, version 2.0. Please see [Apache's web-site](#) for a copy of the license.

From LVT public release 7.1rp1 through 7.2, the LVT source code is available as open source under the NASA Open Source Agreement (NOSA). Please see [LISF's web-site](#) for a copy of the NOSA.

Due to the history of LVT's development, versions of the LVT source code prior to 7.1rp1 may not be freely distributed. Older source code is available only to U.S. government agencies or entities with a U.S. government grant/contract. [LISF's web-site](#) explains how qualified persons may request a copy of older source code.

**NOTE** All users are encouraged to go to [LISF's web-site](#) to fill in the Registration Form and join the mailing list.

## 4.1. Important Note Regarding File Systems

LVT is developed on Linux/Unix platforms. Its build process expects a case sensitive file system. Please make sure that you unpack and/or `git clone` the LVT code into a directory within a case sensitive file system. In particular, if you are using LVT within a Linux-based virtual machine hosted on a Windows or Macintosh system, do not compile/run LVT from within a shared folder. Move the LVT source code into a directory within the virtual machine.

## 4.2. Public Release Source Code

The LVT public release 7.3 source code is available both on [LISF's web-site](#) under the "Source" menu and on GitHub under the NASA-LIS organization at <https://github.com/NASA-LIS/LISF> under the "Releases" link.

After downloading the LISF tar-file:

### Step 1

Create a directory to unpack the tar-file into. Let's call it *TOPLEVELDIR*.

### Step 2

Place the tar-file in this directory.

```
% mv LISF_public_release_7.3.1.tar.gz TOPLEVELDIR
```

### Step 3

Go into this directory.

```
% cd TOPLEVELDIR
```

### Step 4

Unzip and untar the tar-file.

```
% gzip -dc LISF_public_release_7.3.1.tar.gz | tar xf -
```

## 4.3. master branch

The LVT source code is maintained in a git repository hosted on GitHub. If you wish to work with the latest development code (in the master branch), then you must use the `git` client to obtain the LVT source code. If you need any help regarding `git` or GitHub, please go to <https://github.com>.

### Step 1

Create a directory to clone the code into. Let's call it *TOPLEVELDIR*.

### Step 2

Go into this directory.

```
% cd TOPLEVELDIR
```

### Step 3

Clone the master branch.

```
% git clone https://github.com/NASA-LIS/LISF
```

## 4.4. Source files

Unpacking or cloning the LISF source code (according to the instructions in Section [Obtaining the Source Code](#)) will create a directory named *LISF*. The LVT specific source code is in *LISF/lvt*.

**NOTE** The directory containing the LVT source code, *LISF/lvt*, will be referred to as *\$WORKING* throughout the rest of this document.

## 4.5. Documentation

Processed documentation may be found on [LISF's web-site](#) under the “Docs” menu.

# Chapter 5. Building the Executable

This section describes how to build the source code and create LVT's executable: named LVT.

Please see Section [Important Note Regarding File Systems](#) for information regarding using a case sensitive file system for compiling/running LVT.

## 5.1. Development Tools

This code has been compiled and run on Linux PC (Intel/AMD based) systems and Cray systems. These instructions expect that you are using such a system. In particular you need:

### 5.1.1. Linux

#### Compilers

- Intel Fortran Compiler versions 18, 19, or 20 with corresponding Intel C Compiler along with GNU's Compiler Collection version 9.2.0
- or GNU's Compiler Collection version 4.9.2 or 7.3, both gfortran and gcc.

#### Tools

- GNU's make, gmake, version 3.77 or 3.81
- Perl, version 5.10
- Python, version 2.7 or 3.8

#### IMPORTANT

Support for Python 2.7 is now deprecated. Future releases will depend on Python 3.

### 5.1.2. Cray/Linux

#### Compilers

- Intel Fortran Compiler version 18 or 19 with corresponding Intel C Compiler, along with GNU's Compiler Collection version 7.3.0

#### Tools

- GNU's make, gmake, version 3.77 or 3.81
- Perl, version 5.10
- Python, version 2.7 or 3.8

#### IMPORTANT

Support for Python 2.7 is now deprecated. Future releases will depend on Python 3.

## 5.2. Required Software Libraries

In order to build the LVT executable, the following libraries must be installed on your system:

### 5.2.1. Earth System Modeling Framework (ESMF) version 7.1.0r (or higher)

(<https://earthsystemmodeling.org/>)

### 5.2.2. OpenJPEG version 2.3.0 (or higher)

(<http://www.openjpeg.org/>)

### 5.2.3. ecCodes version 2.7.0 (or higher)

(<https://confluence.ecmwf.int/display/ECC>)

## 5.3. Optional Software Libraries

The following libraries are not required to compile LVT. They are used to extend the functionality of LVT.

### 5.3.1. NetCDF either version 3.6.3 or version 4.5.0 (or higher)

(<http://www.unidata.ucar.edu/software/netcdf>)

Please read the on-line documentation for details on installing NetCDF.

#### Additional notes for NetCDF 4:

You must also choose whether to compile with compression enabled. Compiling with compression enabled requires HDF 5 and zlib libraries. To enable compression, add `--enable-netcdf-4` to the `configure` options. To disable compression, add `--disable-netcdf-4` to the `configure` options.

An example of installing NetCDF 4 without compression:

```
% ./configure --prefix=$HOME/local/netcdf-4.5.0 --disable-netcdf-4
% gmake
% gmake install
```

An example of installing NetCDF 4 with compression:

```
% CPPFLAGS=-I$HOME/local/hdf5/1.10.1/include \
> LDFLAGS=-L$HOME/local/hdf5/1.10.1/lib \
> ./configure --prefix=$HOME/local/netcdf/4.5.0 --enable-netcdf-4
% gmake
% gmake install
```



You must also download the *netcdf-fortran-4.4.4.tar.gz* file. First install the NetCDF C library, then install the NetCDF Fortran library. Again, please read the on-line documentation for more details.

An example of installing the NetCDF 4 Fortran library:

```
% LD_LIBRARY_PATH=$HOME/local/netcdf/4.5.0/lib:$LD_LIBRARY_PATH \  
> CPPFLAGS=-I$HOME/local/netcdf/4.5.0/include \  
> LDFLAGS=-L$HOME/local/netcdf/4.5.0/lib \  
> ./configure --prefix=$HOME/local/netcdf/4.5.0  
% gmake  
% gmake install
```

### 5.3.2. HDF

You may choose either HDF version 4, HDF version 5, or both.

HDF is used to support a number of remote sensing datasets.

If you wish to use MODIS snow cover area observations or NASA AMSR-E soil moisture observations, then you need HDF 4 support.

If you wish to use ANSA snow cover fraction observations, then you need HDF 5 support.

If you wish to use PMW snow observations, then you need both HDF 4 and HDF 5 support.

#### HDF 4

If you choose to have HDF version 4 support, please download the HDF source for version 4.2.13 (or higher) from <https://portal.hdfgroup.org/display/support/Download+HDF4> and compile the source to generate the HDF library. Make sure that you configure the build process to include the Fortran interfaces by adding the `--enable-fortran` option to the `configure` command.

Note that HDF4 contains its own embedded version of NetCDF. You must disable this support by adding the `--disable-netcdf` option to the `configure` command.

#### IMPORTANT

When compiling LVT with HDF 4 support, you must also download and compile HDF-EOS2 version 2.19v1.00 or higher from <http://hdfeos.org/software/library.php>.

#### HDF 5

If you choose to have HDF version 5 support, please download the HDF source for version 1.10.1 (or higher) from <http://www.hdfgroup.org/HDF5/> and compile the source to generate the HDF library. Make sure that you configure the build process to include the Fortran interfaces by adding the `--enable-fortran` option to the `configure` command.

### 5.3.3. GDAL version 2.4.1 (or higher)

(<https://gdal.org>)

## IMPORTANT

When compiling LVT with GDAL support, you must also download and compile FortranGIS version 2.4 (or higher) from <http://fortrangis.sourceforge.net>.

### 5.3.4. Notes

To install these libraries, follow the instructions provided at the various URL listed above. These optional libraries have their own dependencies, which should be documented in their respective documentation.

Please note that your system may have several different compilers installed. You must verify that you are building these libraries with the correct compiler. You should review the output from the `configure`, `make`, etc. commands. If the wrong compiler is being used, you may have to correct your `$PATH` environment variable, or set the `$CC` and `$FC` environment variables, or pass additional settings to the `configure` scripts. Please consult the installation instructions provided at the various URL listed above for each library.

Note that due to the mix of programming languages (Fortran and C) used by LVT, you may run into linking errors when building the LVT executable. This is often due to (1) the Fortran compiler and the C compiler using different cases (upper case vs. lower case) for external names, and (2) the Fortran compiler and C compiler using a different number of underscores for external names.

## 5.4. Build Instructions

### Step 1

Perform the steps described in Section [Obtaining the Source Code](#) to obtain the source code.

### Step 2

Goto the `$WORKING` directory. This directory contains two scripts for building the LVT executable: `configure` and `compile`.

### Step 3

Set the `LVT_ARCH` environment variable based on the system you are using. The following commands are written using Bash shell syntax.

*For a Linux system with the Intel Fortran compiler*

```
% export LVT_ARCH=linux_ifc
```

*For a Linux system with the GNU Fortran compiler*

```
% export LVT_ARCH=linux_gfortran
```

It is suggested that you place this command in your `.profile` (or equivalent) startup file.

## Step 4

Run the *configure* script first by typing:

```
% ./configure
```

This script will prompt the user with a series of questions regarding support to compile into LVT, requiring the user to specify the locations of the required and optional libraries via several LVT specific environment variables. The following environment variables are used by LVT.

Variable	Description	Usage
LVT_FC	Fortran 90	required
LVT_CC	C compiler	required
LVT_MODESMF	path to ESMF modules files	required
LVT_LIBESMF	path to ESMF library files	required
LVT_OPENJPEG	path to openJPEG library	required
LVT_ECCODES	path to ecCodes library	required
LVT_NETCDF	path to NETCDF library	optional
LVT_HDF4	path to HDF4 library	optional
LVT_HDF5	path to HDF5 library	optional
LVT_HDFEOS	path to HDFEOS2 library	optional
LVT_GDAL	path to GDAL library	optional
LVT_FORTRANGIS	path to FortranGIS library	optional (required by GDAL)

Note that the **CC** variable must be set to a C compiler, not a C++ compiler. A C++ compiler may mangle internal names in a manner that is not consistent with the Fortran compiler. This will cause errors during linking.

It is suggested that you add these definitions to your *.profile* (or equivalent) startup file.

You may encounter errors either when trying to compile LVT or when trying to run LVT because the compiler or operating system cannot find these libraries. To fix this, you must add these libraries to your **\$LD\_LIBRARY\_PATH** environment variable. For example, say that you are using ESMF, ecCodes, NetCDF, and HDF5. Then you must execute the following command (written using Bash shell syntax):

```
% export
LD_LIBRARY_PATH=$LVT_HDF5/lib:$LVT_LIBESMF:$LVT_NETCDF/lib:$LVT_ECCODES/lib:$LD_LIBRAR
Y_PATH
```

It is also suggested that you add this command to your *.profile* (or equivalent) startup file.

## Example

An example execution of the configure script is shown below:

```
% ./configure
-----
Setting up configuration for LVT
Optimization level (-3=strict checks with warnings, -2=strict checks, -1=debug,
0,1,2,3, default=2):
Assume little/big_endian data format (1-little, 2-big, default=2):
Use GRIBAPI/ECCODES? (1-gribapi, 2-eccodes, default=2):
Use NETCDF? (1=yes, 0=no, default=1):
NETCDF version (3 or 4, default=4):
NETCDF use shuffle filter? (1=yes, 0=no, default = 1):
NETCDF use deflate filter? (1=yes, 0=no, default = 1):
NETCDF use deflate level? (1 to 9=yes, 0=no, default = 9):
Use HDF4? (1=yes, 0=no, default=1):
Use HDF5? (1=yes, 0=no, default=1):
Use HDFEOS? (1=yes, 0=no, default=1):
Enable AFWA-specific grib configuration settings? (1=yes, 0=no, default=0):
Enable GeoTIFF support? (1=yes, 0=no, default=1):
Use MATLAB support? (1=yes, 0=no, default=0):
-----
configure.lvt file generated successfully
-----
Settings are written to configure.lvt in the make directory.
If you wish to change settings, please edit that file.

To compile, run the compile script.
-----
```

At each prompt, select the desired value. If you desire the default value, then you may simply press the Enter key.

Most of the configure options are be self-explanatory. Here are a few specific notes:

- for **Optimization level (-3=strict checks with warnings, -2=strict checks, -1=debug, 0,1,2,3, default=2):**

### IMPORTANT

There is an open issue regarding compiling LVT with the GNU compilers version 7.3 (and higher). One of the modules takes a long time to compile, and the compilation consumes a lot of memory, often to the point of aborting.

When using the GNU compilers, you should select **0**.

- for **Assume little/big\_endian data format (1-little, 2-big, default=2):**, select the default value of 2. By default, LVT reads and writes binary data in the big endian format. Only select the value of 1, if you have reformatted all required binary data into the little endian format.

- for `Use GRIBAPI/ECCODES? (1-gribapi, 2-eccodes, default=2)`:, select the default value of 2. Technically, GRIB support is not required by LVT; however, most of the commonly used met forcing data are in GRIB, making GRIB support a practical requirement. ecCodes is ECMWF's replacement to their GRIB-API library. GRIB-API is supported only for historical reasons; thus, please use ecCodes.

### IMPORTANT

GRIB-API support is now deprecated. Future releases will support only ecCodes.

- for `Enable GeoTIFF support? (1-yes, 0-no, default=1)`:, GeoTIFF means the GeoTIFF support provided by the GDAL library.

Note that due to an issue involving multiple definitions within the NetCDF 3 and HDF 4 libraries, you cannot compile LVT with support for both NetCDF 3 and HDF 4 together.

Note that if you compiled NetCDF 4 without compression, then when specifying `NETCDF version (3 or 4, default=4)`:, select 3. Then you must manually append `-lnetcdf` to the `LDFLAGS` variable in the `make/configure.lvt` file.

## Step 5

Compile the LVT source code by running the `compile` script.

```
% ./compile
```

This script will compile the libraries provided with LVT, the dependency generator and then the LVT source code. The executable `LVT` will be placed in the `$WORKING` directory upon successful completion of the `compile` script.

## Step 6

Finally, copy the `LVT` executable into your running directory, `$RUNNING`. (See Section [Running the Executable](#).)

# Chapter 6. Running the Executable

This section describes how to run the LVT executable.

First you should create a directory to run LVT in. It is suggested that you run LVT in a directory that is separate from your source code. This running directory shall be referred to as *\$RUNNING*. Next, copy the LVT executable into your running directory

```
% cp $WORKING/LVT $RUNNING
```

The single-process version of *LVT* is executed by the following command issued in the *\$RUNNING* directory.

```
% ./LVT <configfile>
```

where *<configfile>* represents the file containing the run time configuration options for LVT. Currently LVT only supports a serial mode.

Some systems require that you submit your job into a batch queue. Please consult with your system administrator for instructions on how to do this.

Note that before running LVT, you must set your environment to have an unlimited stack size. For the Bash shell, run

```
% ulimit -s unlimited
```

To customize your run, you must specify a LVT runtime configuration file. See Section [LVT config File](#) for more information.

# Chapter 7. LVT config File

This section describes the options in the *lvt.config* file.

## 7.1. Overall driver options

**LVT running mode:** specifies the running mode to be used. Acceptable values are:

Value	Description
“Data intercomparison”	standard analysis mode where a particular data is compared against another
“Benchmarking”	A benchmarking output is generated based on the input training datasets
“DA statistics processing”	data assimilation diagnostics analysis
“DA observation processing”	data assimilation observation analysis
“OPTUE output processing”	parameter estimation/uncertainty output analysis
“RTM output processing”	radiative transfer model output analysis
“557 post”	LIS postprocessing mode for 557WW operations
“USAFSI post”	special postprocessing mode for USAFSI

*Example lvt.config entry*

```
LVT running mode: "Data intercomparison"
```

**Map projection of the LVT analysis:** specifies the map projection used in the LVT analysis. Acceptable values are:

Value	Description
latlon	Lat/Lon projection with SW to NE data ordering
mercator	Mercator projection with SW to NE data ordering
lambert	Lambert conformal projection with SW to NE data ordering
gaussian	Gaussian domain
polar	Polar stereographic projection with SW to NE data ordering
UTM	UTM domain

*Example lvt.config entry*

```
Map projection of the LVT analysis:      "latlon"
```

**LVT output format:** specifies the format of the LVT output. Acceptable values are:

Value	Description
binary	Write output in binary format
grib1	Write output in Grib format (not supported yet)
netcdf	Write output in NETCDF format

See Appendix [Description of output files from LVT](#) for more details about the structure of the LVT output files.

*Example lvt.config entry*

```
LVT output format:                      "netcdf"
```

**LVT output methodology:** specifies the output methodology used in LVT. The LVT output is written as a 1-D array containing only land points or as a 2-D array containing both land and water points. 1-d tile space includes the subgrid tiles and ensembles. 1-d grid space includes a vectorized, land-only grid-averaged set of values. Acceptable values are:

Value	Description
"1d tilespace"	LVT output in a 1-D tile domain
"2d gridspace"	LVT output in a 2-D grid domain
"1d gridspace"	LVT output in a 1-D grid domain

*Example lvt.config entry*

```
LVT output methodology:                 "2d gridspace"
```

**Analysis data sources:** specifies the two sources of data to be used in an LVT comparison. The user must always choose two sources specified in adjacent columns. The second column entry will be taken as the reference data and the first column will be used as the data being evaluated (against the reference data). If the comparison LIS output against a non-LIS data, it is recommended to specify the first source as "LIS output" and then the other data as the second data source. Acceptable values for each column are:

Value	Description
"none"	template
"LIS output"	output from another LIS run
"LIS DAOBS"	processed observations from a LIS DA run



<b>Value</b>	<b>Description</b>
“ISCCP LST”	ISCCP skin temperature observations
“MODIS LST”	MODIS (Terra/Aqua) land surface temperature observations
“SCAN”	SCAN soil moisture station observations
“NASMD”	North American Soil Moisture Database soil moisture station observations
“ISMN”	ISMN soil moisture station observations
“SURFRAD”	SURFRAD observations
“SNOTEL”	SNOTEL snow water equivalent observations
“LSWG Tb”	Tb brightness temperature observations at the LSWG sites
“FMI SWE”	Finnish Meteorological Institute (FMI) snow course data
“CMC”	Canadian Meteorological Center (CMC) snow depth analysis
“SNODAS”	NOHRSC SNow Data Assimilation (SNODAS) product
“AMSR-E NASA soil moisture”	NASA (NSIDC) retrieval of AMSR-E soil moisture
“AMSR-E LPRM soil moisture”	LPRM (VU) retrieval of AMSR-E soil moisture
“AMMA”	AMMA station observations
“Ameriflux”	Ameriflux station observations
“ARM”	ARM station observations
“SMOSREX”	SMOSREX station observations
“AGRMET”	AGRMET land surface analysis
“Globsnow”	GlobSnow SWE analysis
“SNODEP metobs”	WMO snow depth station observations
“MOD10A1”	MOD10A1 fractional snow cover data from MODIS
“MOD10A1V6”	MOD10A1.006 Normalized Difference from MODIS
“ANSA snowdepth”	ANSA snow depth retrievals
“ANSA SWE”	ANSA SWE retrievals
“CPC precipitation”	CPC unified precipitation product
“GPM IMERG”	IMERG precipitation data product
“USGS streamflow”	USGS streamflow observations

<b>Value</b>	<b>Description</b>
“Naturalized streamflow”	Naturalized streamflow estimates
“FLUXNET MTE”	Gridded FLUXNET MTE data from MPI
“MOD16A2”	MOD16A2 ET products from MODIS
“UW ET”	University of Washington ET products from MODIS
“ALEXI”	Atmosphere Land Exchange Inverse (ALEXI) model ET estimates from USDA
“USDA ARS soil moisture”	soil moisture measurements from USDA ARS watersheds
“GHCN”	Global Historical Climatology Network data
“ALEXI ESI”	Atmosphere Land Exchange Inverse (ALEXI) model Evaporative Stress Index (ESI) estimates
“NLDAS2”	North American Land Data Assimilation System Phase-2 data
“GRACE”	processed GRACE data used in a LIS-DA instance
“PBO H2O”	plate boundary observatory data
“USGS ground water well data”	USGS ground water well data
“SMOS L2 soil moisture”	SMOS level 2 soil moisture
“SMOS L1 TB”	SMOS level 1 brightness temperature
“GCOMW AMSR2 L3 soil moisture”	GCOMW AMSR2 level 3 soil moisture
“SMOPS soil moisture”	Soil Moisture Operational Product System data
“ESA CCI soil moisture”	ESA CCI soil moisture
“GIMMS NDVI”	GIMMS NDVI data
“GIMMS AVHRR NDVI”	GIMMS AVHRR NDVI data
“GIMMS MODIS NDVI”	GIMMS MODIS NDVI data
“GLDAS2”	NASA Global Land Data Assimilation System version 2 data
“MERRA2”	MERRA version 2 data
“ERA interim land”	ERA Interim Land data
“ERA5”	ERA5 data (currently only precipitation and near surface air temperature variables are supported)
“SSEB”	USGS SSEB operational ET and ET anomaly data
“GRDC”	Global Runoff Data Center data
“GLERL hydro data”	Great Lakes hydrology data

Value	Description
“GL6 JULES data”	GL6 JULES data
“GLEAM”	GLEAM data
“FLUXNET2015 dataset”	Measurements from FLUXNET2015 dataset
“USCRN soil moisture”	USCRN soil moisture data
“SMAP soil moisture”	SMAP soil moisture data
“SMAP vegetation water content”	SMAP vegetation water content
“SMAP vegetation optical depth”	SMAP vegetation optical depth
“LPRM vegetation optical depth”	LPRM vegetation optical depth
“WG SWRC”	specifies what?
“UA SNOW”	University of Arizona SWE/Snow depth data
“SMAP L3 Tb”	SMAP L3 brightness temperature
“OzFlux”	OzFlux ground measurements
“JASMIN soil moisture”	JASMIN soil moisture analysis

Example *lvt.config* entry

```
Analysis data sources:    "LIS output" "SURFRAD"
```

**Experiment code:** LIS-6 experiment code number.

**NOTE** | Only needed when using LVT on LIS 6 output.

Example *lvt.config* entry

```
Experiment code:          111
```

**Start mode:** specifies if the LVT analysis is to be restarted from a previous (unfinished) analysis. Note that if “restart” option is selected, then the starting time (below) must be changed appropriately. Acceptable values are:

Value	Description
coldstart	Fresh analysis
restart	Restart from a previous analysis

Example *lvt.config* entry

```
Start mode:              "coldstart"
```

**LVT output restart files:** a flag to specify if LVT restart files should be written out. The default is to turn off writing these restart files. To enable writing of LVT restart files, change this option to “1”.

*Example lvt.config entry*

```
LVT output restart files:          0
```

**LVT restart output interval:** specifies the frequency at which the restart files must be written during a LVT analysis. The time interval is specified with a number followed by a 2 character suffix that indicates the units. For example, a restart interval of 1 hour can be specified as “1hr”, “60mn”, or “3600ss”. Acceptable values for the 2 character suffix are:

Value	Description
ss	second
mn	minute
hr	hour
da	day
mo	month
yr	year

*Example lvt.config entry*

```
LVT restart output interval:      "1mo"
```

**LVT restart filename:** specifies the name of the LVT restart file

*Example lvt.config entry*

```
LVT restart filename:             "none"
```

The start time of the evaluation period is specified in the following format:

Variable	Value	Description
<b>Starting year:</b>	integer 2001 – present	specifying starting year
<b>Starting month:</b>	integer 1 – 12	specifying starting month
<b>Starting day:</b>	integer 1 – 31	specifying starting day
<b>Starting hour:</b>	integer 0 – 23	specifying starting hour
<b>Starting minute:</b>	integer 0 – 59	specifying starting minute
<b>Starting second:</b>	integer 0 – 59	specifying starting second

*Example lvt.config entry*

```
Starting year:          2007
Starting month:        11
Starting day:          1
Starting hour:         0
Starting minute:       0
Starting second:       0
```

The end time of the evaluation period is specified in the following format:

Variable	Value	Description
Ending year:	integer 2001 – present	specifying ending year
Ending month:	integer 1 – 12	specifying ending month
Ending day:	integer 1 – 31	specifying ending day
Ending hour:	integer 0 – 23	specifying ending hour
Ending minute:	integer 0 – 59	specifying ending minute
Ending second:	integer 0 – 59	specifying ending second

*Example lvt.config entry*

```
Ending year:          2008
Ending month:         5
Ending day:           31
Ending hour:          0
Ending minute:        0
Ending second:        0
```

**LVT clock timestep:** specifies the timestep to be used in the LVT analysis. It is recommended to use a timestep consistent with the output frequency of the first data stream.

*Example lvt.config entry*

```
LVT clock timestep:    "1da"
```

**Undefined value:** specifies the undefined value. The default is set to -9999.

*Example lvt.config entry*

```
Undefined value:      -9999
```

**LVT diagnostic file:** specifies the name of run time diagnostic file. Acceptable values are any 40 character string.

*Example lvt.config entry*

```
LVT diagnostic file:          lvtlog
```

## 7.2. Domain specification

LVT expects three sets of domain specification. (1) the domain over which the LVT analysis needs to be carried out (2) the domain in which LIS simulation was carried out (LIS run domain). Section [Overall driver options](#) lists the projections that LIS supports.

### 7.2.1. LVT run domain

This section describes how to specify the run domain over which LVT will perform its analysis. See the LDT Users' Guide for more details about setting these values.

*Example lvt.config entry*

```
Run domain lower left lat:    30.125
Run domain lower left lon:   -124.875
Run domain upper right lat:   50.125
Run domain upper right lon:  -69.875
Run domain resolution (dx):   0.25
Run domain resolution (dy):   0.25
```

**Input domain and mask data file:** specifies the name of the parameter input file to be used for the definition of domain and parameters in LVT. This file should be in NetCDF format and must include details about the map projection (as global attributes) and must include a landmask field. LVT uses the landmask specified in this file to derive the landmask for the LVT run domain. The extents of the run domain should be inclusive of the landmask specified in this file (but it doesn't have to match exactly).

*Example lvt.config entry*

```
Input domain and mask data file:    ./lis_input.d01.nc
```

**LVT datastream attributes table::** specifies the variables being analyzed and their attributes. The list of variables are to be specified in a table form with each line representing the variable specification from datastream 1 and datastream 2. In the example below, latent heat flux variable from datastream 1 is compared against the sensible heat flux from datastream 2. In addition, the root zone soil moisture from datastream 1 is compared against the root zone soil moisture from datastream 2. Each variable specification consists of 7 columns representing the short name, starting index of the level of the variable (0 indicates that variable is not selected), ending index of the level of the variable, units, direction type, time-averaging option (0 = instantaneous, 1 = time-averaged) and total number of vertical levels.

**NOTE**

This line must end with two colons after the word table (e.g., “table::”), and the final line of the table is a line with only two colons (e.g., “::”). The table information is contained on the lines in-between these beginning and end lines. For example, if the soil moisture variable has 4 levels and the user chooses levels 2 to 3, then they can be entered as (for a datastream.

SoilMoist 2 3 m3/m3 - 1 4

*Example lvt.config entry*

```
LVT datastream attributes table::
Qle      1 1 W/m2  DN  1  1 Qh      1 1 W/m2  DN  1  1
RootMoist 1 1 m3/m3  -   1  1 RootMoist 1 1 m3/m3  -   1  1
::
```

**LVT surface soil layer thickness:** specifies the thickness (in m) of the surface soil layer to be considered in the LVT analysis.

*Example lvt.config entry*

```
LVT surface soil layer thickness:      0.1
```

**LVT root zone soil layer thickness:** specifies the thickness (in m) of the root zone soil layer to be considered in the LVT analysis.

*Example lvt.config entry*

```
LVT root zone soil layer thickness:    1.0
```

## 7.3. Analysis options specification

This section of the config file specifies the type of analysis to be conducted during the verification/evaluation. Note that some options are only available in certain running modes.

**Apply external mask:** Specifies whether to apply an external mask in limiting the analysis to a selected set of data points. Note that undefined value is considered to be the value used for omitting grid points. All values other than “undefined values” (e.g. -9999.0) are considered as valid. Acceptable values are:

Value	Description
0	Do not apply external mask
1	Apply external, temporally varying mask
2	Apply fixed mask
3	Apply temporal monthly mask

Example lvt.config entry

```
Apply external mask: 0
```

**Temporal (monthly) mask flags:** specifies the temporal (monthly) mask flags (specified as 0/1 for each calendar month). This option is only read when the **Apply external mask:** option is set to 3.

Example lvt.config entry

```
Temporal (monthly) mask flags: 0 0 0 0 0 1 1 1 0 0 0 0
```

**External mask directory:** Specifies the name of the data mask file/directory. If the mask varies temporally, then this option specifies the top-level directory containing data mask. Note that the mask files are expected to be in binary, sequential access format.

Example lvt.config entry

```
External mask directory: "none"
```

**Compute ensemble metrics:** specifies whether to compute ensemble-based metrics or not. If this option is turned on, then all the traditional (non-ensemble) metrics will be turned off. Acceptable values are:

Value	Description
0	Do not compute
1	compute ensemble metrics

Example lvt.config entry

```
Compute ensemble metrics: 0
```

**Compute information theory metrics:** specifies whether to compute information theory-based metrics or not. If this option is turned on, then the ensemble and traditional metrics will be turned off. Acceptable values are:

Value	Description
0	Do not compute
1	compute information theory metrics

Example lvt.config entry

```
Compute information theory metrics: 0
```

**Metrics attributes file:** specifies the attributes of the metrics that are used in the LVT analysis. Section [Configuration of metrics](#) describes the format of the metrics attributes file.



*Example lvt.config entry*

```
Metrics attributes file:          ./METRICS.TBL
```

**Observation count threshold:** Specifies the number of observations to be used as the minimum threshold for computing statistics. Grid points with observation count less than this value will be ignored. Acceptable values are 0 or higher

*Example lvt.config entry*

```
Observation count threshold:      50
```

**Metrics computation frequency:** Specifies the temporal averaging interval of the LIS output and observation data.

The ending time and the **Metrics computation frequency:** must be consistent. For example, for a **Metrics computation frequency:** of 1da, you must have an ending time of:

**NOTE**

```
Ending hour: 0  
Ending minute: 0  
Ending second: 0
```

*Example lvt.config entry*

```
Metrics computation frequency:    "1da"
```

**Temporal lag in metrics computations:** Specifies the temporal lag in metric computations. The values can be positive or negative (e.g. "+1da" or "-1da")

*Example lvt.config entry*

```
Temporal lag in metrics computations: 0
```

**Spatial averaging mode:** Specifies the spatial averaging mode to be used. Acceptable values are:

Value	Description
pixel-by-pixel	each pixel is treated separately
region-based	stats are computed for on a region by region basis

*Example lvt.config entry*

```
Spatial averaging mode:    "pixel-by-pixel"
```

**Regional mask file for spatial averaging:** Specifies the name of the regional mask file to be used for determining the (sub) regions, if region-based spatial averaging mode is selected.

This file must be in big-endian, sequential access format and must correspond exactly to the LVT run domain.

*Example lvt.config entry*

```
Regional mask file for spatial averaging: none
```

**Metrics output directory:** Specifies the top-level directory where the output from the analysis is to be written.

*Example lvt.config entry*

```
Metrics output directory:          ./STATS
```

**Metrics output frequency:** Specifies the frequency (in seconds) of the analysis output.

**NOTE**

The **Metrics output frequency:** is simply a setting for specifying the frequency of LVT outputs. If the **Metrics output frequency:** is different from (greater than) the time averaging interval, no additional averaging will be performed between the time averaging intervals.

*Example lvt.config entry*

```
Metrics output frequency:          "1da"
```

**Apply temporal smoothing to obs:** specifies whether to temporal smoothing to the observations. If enabled, the code will compute an average value across the specified time window, instead of only using the value that corresponds to the current time. Acceptable values are:

Value	Description
1	apply temporal smoothing to obs
0	do not apply temporal smoothing to obs

*Example lvt.config entry*

```
Apply temporal smoothing to obs: 0
```

**Obs temporal smoothing window half length:** specifies the observation temporal smoothing window half length. The smoothing window is then defined as (current time +/- half length).

*Example lvt.config entry*

```
Obs temporal smoothing window half length: "2da"
```

**Obs temporal smoothing window interval:** specifies the observation temporal smoothing window interval. This will be used as the increment length across the smoothing window. For e.g., if the window half length is specified as 2 days the smoothing window will be of 5 days. If the smoothing window interval is 1 day, then number of points in the smoothing window will be 5 (-2 da, -1da, current day, +1da, +2da).

*Example lvt.config entry*

```
Obs temporal smoothing window interval: "1da"
```

**Time series location file:** specifies the name of the file which lists the locations and regions in the domain where ASCII time series data are to be derived.

The locations can be specified in five different formats: (1) using the lat/lon values; (2) using the column/row indices; (3) using the tile indices; (4) specifying lat/lon values to draw a polygon around a region; and (5) using a categorical from which to define subregions.

Note that LVT has been updated so the format of the time series locations file uses a minimum fraction of the domain before the temporal calculations will occur. Previously, the time series locations file used a minimum number of observations. A value for this “min frac” of 0.1 (for example) implies that at least 10 percent of the total number of points in the domain location must be available for the temporal calculations to occur.

A sample file with location style 1 is shown below:

```
#Number of locations
2
#Location style (1-lat/lon, 2-col/row, 3-tile, 4-polygon, 5-map)
1
#Name (then, next line), SW-lat, SW-lon, NE-lat, NE-lon, min frac
WEST_US
40.0 -130.0 50.0 -110.0 0.0
HIGH_PLAINS_US
43.0 -110.0 49.0 -100.0 0.0
```

If the location style is 2, the user specifies the column and row indices for the bounding boxes, instead of the corner lat/lon values. A sample file with location style 2 is shown below:

```
#Number of locations
2
#Location style (1-lat/lon, 2-col/row, 3-tile, 4-polygon, 5-map)
2
#Name (then, next line), SW-col, SW-row, NE-col, NE-row, min frac
WEST_US
1 1 20 30 0.0
EAST_US
21 1 40 30 0.0
```

If the location style is 3, the user specifies the tile indices for specifying the bounds (starting tile index and ending tile index). A sample file with location style 3 is shown below:

```
#Number of locations
2
#Location style (1-lat/lon, 2-col/row, 3-tile, 4-polygon, 5-map)
3
#Name (then, next line), Start index, End index, min frac
WEST_US
 1 20 0.0
EAST_US
21 40 0.0
```

If the location style is 4, the user explicitly specifies the lat/lons of each grid point to be used to specify a region in the shape of a polygon. Users should be careful with this location style option, as they cannot specify a minimum fraction of the domain that must have valid observations. A sample file with location style 4 is shown below:

```
#Number of locations
2
#Location style (1-lat/lon, 2-col/row, 3-tile, 4-polygon, 5-map)
4
#Number of points followed by lat/lon of each point
REGION1
3
34.4 -103.2
33.4 -100.2
32.1 -99.3
REGION2
2
40.2 -103.3
42.2 -104.2
```

If the location style is 5, the user explicitly specifies a categorical map from which to define subregions. In the map, the categories must be in numerically increasing order from 1. The map must be a binary direct-access file, with point (1,1) in the southwest corner of the domain. A sample file with location style 5 is shown below:

```

#Number of stations
3
#Location style (1-lat/lon, 2-col/row, 3-tile, 4-polygon, 5-map)
5
#Name (then, next line), min frac
NEWENGLAND
0.0
MIDATLANTIC
0.0
SOUTHATLANTIC
0.0
#categorical map
../huc02_conus_0.125dg.1gd4r

```

Please see the sample TS\_LOCATIONS.TXT file for an example in location format (1).

*Example lvt.config entry*

```
Time series location file:      ../TS_LOCATIONS.TXT
```

**Variable-based stratification:** Specifies if the errors are to be stratified using one of the model output variables. The errors will be stratified into two levels (1) where the values of the stratification variable falls above the specified threshold and (2) where the values of the stratification variable falls below the specified threshold. Acceptable values are:

Value	Description
0	Do not stratify
1	Stratify errors

*Example lvt.config entry*

```
Variable-based stratification:  0
```

**Stratification variable:** Specifies the name of the variable to be used in the stratification

*Example lvt.config entry*

```
Stratification variable:      SWdown_f
```

**Stratification threshold:** Specifies the minimum value to be used as the stratification threshold.

*Example lvt.config entry*

```
Stratification threshold:     1.0
```

**Confidence interval (%):** Specifies the confidence interval threshold (in percentage) of the

computed statistics

Example *lvt.config* entry

```
Confidence interval (%):          95
```

**External data-based stratification:** Specifies if the errors are to be stratified based on an external (static) dataset (e.g. landcover, elevation,etc.) The associated attributes file specifies the stratification details. Acceptable values are:

Value	Description
0	Do not stratify
1	Stratify errors

Example *lvt.config* entry

```
External data-based stratification:  0
```

**Stratification attributes file:** Specifies the name of the file which lists the details of the stratification. The format of the time series location file is as follows:

```
#Number of stratification data sources
3
#Stratification data files
srtm_elev1km.1gd4r
srtm_slope1km.1gd4r
srtm_aspect1km.1gd4r
#stratification variable name
ELEV
SLOPE
ASPECT
#Max (row 1) min (row2) values for each category
7000 1.0 6
500 0.0 0
#number of bins
12 12 12
```

Example *lvt.config* entry

```
Stratification attributes file:      ./strat_attribs.txt
```

**Seasonal cycle interval type:** Specifies the interval type for average seasonal cycle computations (when enabled in the METRICS.TBL file) Acceptable values are:

Value	Description
monthly	monthly seasonal cycles

Value	Description
3 monthly	3-monthly seasonal cycles (DJF,MAM,JJA,SON)
6 monthly	6-monthly seasonal cycles
yearly	yearly seasonal cycles

*Example lvt.config entry*

```
Seasonal cycle interval type:          "monthly"
```

**Seasonal cycle minimum count threshold:** Specifies minimum number of points to be used in computing the average seasonal cycle computations.

*Example lvt.config entry*

```
Seasonal cycle minimum count threshold:    10
```

**Average diurnal cycle minimum count threshold:** Specifies minimum number of points to be used in computing the average diurnal cycle computations.

*Example lvt.config entry*

```
Average diurnal cycle minimum count threshold:    10
```

**Averaging window for computing mean values in anomaly calculations:** Specifies the time window to use for computing mean values to be used in anomaly calculations. This option only applies to the Anomaly metrics (Anomaly, Anomaly R, Anomaly RMSE). Acceptable values are:

Value	Description
monthly	use monthly means
yearly	use annual means

*Example lvt.config entry*

```
Averaging window for computing mean values in anomaly calculations: monthly
```

**Compute only the climatology for percentiles:** If set to 1, then LVT will only calculate the climatology when calculating percentiles. After the percentiles have been calculated, then LVT can be run in restart mode using these climatology files without having to calculate the climatology every time. If set to any value other than 1, LVT will first calculate percentiles climatology, and then calculate the percentiles.

*Example lvt.config entry*

```
Compute only the climatology for percentiles:    0
```

**Scale model data prior to computing percentiles:** specifies whether to scale model data prior to computing percentiles. The standard normal deviate based scaling is employed. Acceptable values are:

Value	Description
0	do not scale
1	scale

**Percentile scaling mean (input data) filename:** specifies the name of the file containing the input data mean.

**Percentile scaling standard deviation (input data) filename:** specifies the name of the file containing the input data standard deviation.

**Percentile scaling mean (scaled data) filename:** specifies the name of the file containing the scaled data mean.

**Percentile scaling standard deviation (scaled data) filename:** specifies the name of the file containing the scaled data standard deviation

**Percentile scaling start time for scaling:** specifies the time to start scaling.

**Percentile scaling end time for scaling:** specifies the time to end scaling.

*Example lvt.config entry*

```
Scale model data prior to computing percentiles: 0
Percentile scaling mean (input data) filename:
Percentile scaling standard deviation (input data) filename:
Percentile scaling mean (scaled data) filename:
Percentile scaling standard deviation (scaled data) filename:
Percentile scaling start time for scaling:
Percentile scaling end time for scaling:
```

**Compute only the climatology for SGWI:** If set to 1, then LVT will only calculate the climatology for SGWI.

*Example lvt.config entry*

```
Compute only the climatology for SGWI: 0
```

**Compute only the climatology for SPI:** If set to 1, then LVT will only calculate the climatology for SPI.

*Example lvt.config entry*

```
Compute only the climatology for SPI: 0
```



**Compute only the climatology for SRI:** If set to 1, then LVT will only calculate the climatology for SRI.

*Example lvt.config entry*

```
Compute only the climatology for SRI: 0
```

**SPI timescale of computation:** Set to 1,3,6,...for SPI (or SRI, SSWI) agregations.

**Compute only the climatology for SSWI:** If set to 1, then LVT will only calculate the climatology for SSWI.

*Example lvt.config entry*

```
Compute only the climatology for SSWI: 0
```

**Starting month if a shifted year definition is used in temporal averaging:** The starting month (integer from 1 to 12) if doing a yearly average or outputting the stats in yearly intervals. Setting this value to 10, for example, represents the start of a hydrologic water year.

*Example lvt.config entry*

```
Starting month if a shifted year definition is used in temporal averaging: 10
```

**Time specification option for MinTime metric:** The “MinTime” metric computes the time at which the minimum value occurs within the stats writing time window. This particular option specifies the nature of the time value saved. Acceptable values are:

Value	Description
doy	day of year
mo	month
da	day
hr	hour
mn	minute
ss	second

If doy is chosen, for example, then the day of the year at which the minimum value occurs will be saved.

*Example lvt.config entry*

```
Time specification option for Mintime metric: doy
```

**Time specification option for MaxTime metric:** The “Maxtime” metric computes the time at which the maximum value occurs within the stats writing time window. This particular option specifies

the nature of the time value saved. Acceptable values are:

Value	Description
doy	day of year
mo	month
da	day
hr	hour
mn	minute
ss	second

If doy is chosen, for example, then the day of the year at which the maximum value occurs will be saved.

*Example lvt.config entry*

```
Time specification option for MaxTime metric: doy
```

**Calculate reference ET without energy balance:** allows for the reference evapotranspiration to be calculated without using surface energy flux variables that are generated by the land-surface model. The calculation follows the methodology in Hobbins, M. et al. 2016.

Value	Description
0	use default Reference ET calculation
1	use Reference ET without LSM output

This option is only valid for Template LSM runs.

*Example lvt.config entry*

```
Calculate reference ET without energy balance: 1
```

## 7.4. Data stream sources

This section of the config file specifies the details of the sources of data streams.

### 7.4.1. LIS output

**LIS output analysis data class:** specifies the type of land surface data that is being analyzed. Acceptable values are:

Value	Description
LSM	LSM variables
Routing	Routing variables

Value	Description
RTM	Radiative transfer model variables
Irrigation	Irrigation model output

**LIS output number of surface model types:** specifies the number of surface model types used in the LIS simulation

**LIS output surface model types:** specifies the surface model types used in the LIS simulation, in consecutive columns. Acceptable values are:

Value	Description
LSM	land surface model types
Lake	Lake model types

**LIS output model name:** specifies the name of the model used that generated the output. Acceptable values are:

Value	Description
“CLSM”	Catchment land surface model
“CLM”	CLM 2.0
“CLSM F2.5”	Catchment Fortuna 2.5
“HYSSIB”	HySSiB
“MOS”	Mosaic
“Noah.2.7.1”	Noah 2.7.1
“Noah.3.2”	Noah 3.2
“Noah.3.3”	Noah 3.3
“SiB2”	SiB2
“TEMPLATE”	template (no model)
“VIC411”	VIC 4.1.1
“WRSI”	GeoWRSI 2.0

**LIS output domain and parameter file:** specifies the name of the parameter input file used in the LIS simulation This file is generated by the Land Data Toolkit (LDT).

**LIS output directory:** specifies the LIS output directory

**LIS output naming style:** specifies the style of the LIS output. Acceptable values are:

Value	Description
“5 level hierarchy”	5 levels of hierarchy
“3 level hierarchy”	3 levels of hierarchy

Value	Description
“2 level hierarchy”	2 levels of hierarchy
“WMO convention”	WMO convention for weather codes (written by LIS)
“WMO convention (AFW OPS)”	WMO convention for weather codes (for AFW production use)

**LIS output methodology:** specifies the output methodology used for generating the output. The LIS output is written as a 1-D array containing only land points or as a 2-D array containing both land and water points. 1-d tile space includes the subgrid tiles and ensembles. 1-d grid space includes a vectorized, land-only grid-averaged set of values. Acceptable values are:

Value	Description
“1d tilespace”	LIS output in a 1-D tile domain
“2d gridspace”	LIS output in a 2-D grid domain
“1d gridspace”	LIS output in a 1-D grid domain

**LIS output format:** specifies the format of the LIS output. Acceptable values are:

Value	Description
binary	Write output in binary format
grib1	Write output in Grib format (not supported yet)
netcdf	Write output in NETCDF format

**LIS output interval:** specifies the frequency at which the LIS output was written. The time interval is specified with a number followed by a 2 character suffix that indicates the units. For example, an output interval of 1 hour can be specified as “1hr”, “60mn”, or “3600ss”. Acceptable values for the 2 character suffix are:

Value	Description
ss	second
mn	minute
hr	hour
da	day
mo	month
yr	year

**LIS output attributes file:** specifies the model output attribute file used for generating the LIS output

**LIS output maximum number of surface type tiles per grid:** defines the maximum surface type tiles per grid (this can be as many as the total number of vegetation/landcover types) used in the LIS

simulation.

**LIS output minimum cutoff percentage (surface type tiles):** defines the smallest percentage (among the surface type distributions within a grid cell) for which to create a tile, used in the LIS simulation. The percentage value is expressed as a fraction.

**LIS output maximum number of soil texture tiles per grid:** defines the maximum soil texture type tiles per grid (this can be as many as the total number of soil texture types) used in the LIS simulation.

**LIS output minimum cutoff percentage (soil texture tiles):** defines the smallest percentage (among the soil texture distributions within a grid cell) for which to create a tile, used in the LIS simulation. The percentage value is expressed as a fraction.

**LIS output maximum number of soil fraction tiles per grid:** defines the maximum soil fraction tiles per grid used in the LIS simulation.

**LIS output minimum cutoff percentage (soil fraction tiles):** defines the smallest percentage (among the soil fraction distributions within a grid cell) for which to create a tile, used in the LIS simulation. The percentage value is expressed as a fraction.

**LIS output maximum number of elevation bands per grid:** defines the maximum elevation bands per grid used in the LIS simulation.

**LIS output minimum cutoff percentage (elevation bands):** defines the smallest percentage (among the elevation distributions within a grid cell) for which to create a tile, used in the LIS simulation. The percentage value is expressed as a fraction.

**LIS output maximum number of slope bands per grid:** defines the maximum slope bands per grid used in the LIS simulation.

**LIS output minimum cutoff percentage (slope bands):** defines the smallest percentage (among the slope distributions within a grid cell) for which to create a tile, used in the LIS simulation. The percentage value is expressed as a fraction.

**LIS output maximum number of aspect bands per grid:** defines the maximum aspect bands per grid used in the LIS simulation.

**LIS output minimum cutoff percentage (aspect bands):** defines the smallest percentage (among the aspect distributions within a grid cell) for which to create a tile, used in the LIS simulation. The percentage value is expressed as a fraction.

**LIS output nest index:** specifies the nest index of the domain.

**LIS output elevation data source:** specifies the name of the topographical elevation data source used in the LIS run (This information is used to replicate the domain creation that was done in the LIS run).

**LIS output slope data source:** specifies the name of the topographical slope data source used in the LIS run (This information is used to replicate the domain creation that was done in the LIS run).

**LIS output aspect data source:** specifies the name of the topographical aspect data source used in the LIS run (This information is used to replicate the domain creation that was done in the LIS run).

**LIS output soil texture data source:** specifies the name of the soil texture data source used in the LIS run (This information is used to replicate the domain creation that was done in the LIS run).

**LIS output soil fraction data source:** specifies the name of the soil fraction data source used in the LIS run (This information is used to replicate the domain creation that was done in the LIS run).

**LIS output number of soil moisture layers:** specifies the number of soil moisture layers used in the LIS output.

**LIS output number of soil temperature layers:** specifies the number of soil temperature layers used in the LIS output.

**LIS output soil moisture layer thickness:** specifies the thicknesses of soil moisture layers in the LIS output.

**LIS output soil temperature layer thickness:** specifies the thicknesses of soil temperature layers in the LIS output.

### Example lvt.config entry

```
LIS output analysis data class:          "LSM"
LIS output number of surface model types: 1
LIS output surface model types:         "LSM"
LIS output model name:                   "Noah.3.3"
LIS output domain and parameter file:    "lis_input.d01.nc"
LIS output directory:                    ./CLSM/OUTPUT
LIS output naming style:                  "3 level hierarchy"
LIS output methodology:                   "2d gridspace"
LIS output format:                        "netcdf"
LIS output interval:                      "1da"
LIS output attributes file:               ../Data/Noah33_CONUS/NOAH33_OUTPUT_LIST.TBL
LIS output maximum number of surface type tiles per grid: 1
LIS output minimum cutoff percentage (surface type tiles): 0.10
LIS output maximum number of soil texture tiles per grid: 1
LIS output minimum cutoff percentage (soil texture tiles): 0.10
LIS output maximum number of soil fraction tiles per grid: 1
LIS output minimum cutoff percentage (soil fraction tiles): 0.10
LIS output maximum number of elevation bands per grid: 1
LIS output minimum cutoff percentage (elevation bands): 0.10
LIS output maximum number of slope bands per grid: 1
LIS output minimum cutoff percentage (slope bands): 0.10
LIS output maximum number of aspect bands per grid: 1
LIS output minimum cutoff percentage (aspect bands): 0.10
LIS output number of ensembles per tile: 1
LIS output nest index:                    1
LIS output elevation data source: none
LIS output slope data source:             none
LIS output aspect data source:            none
LIS output soil texture data source: none
LIS output soil fraction data source: none
LIS output number of soil moisture layers: 4
LIS output number of soil temperature layers: 4
LIS output soil moisture layer thickness: 0.1 0.3 0.6 1.0
LIS output soil temperature layer thickness: 0.1 0.3 0.6 1.0
```

## 7.4.2. AGRMET data

**AGRMET data directory:** specifies the location of the AGRMET data.

**AGRMET data security class name:** specifies the security level classification of the data (U for unclassified)

**AGRMET data distribution class name:** specifies the distribution classification of AGRMET data

**AGRMET data category name:** specifies category name for AGRMET data.

**AGRMET data area of data:** specifies geographical extent of AGRMET data

*Example lvt.config entry*

```
AGRMET data directory:          ./AGRMET_data
AGRMET data security class name: U
AGRMET data distribution class name: C
AGRMET data category name:     ANLYS
AGRMET data area of data:      GLOBAL
```

### 7.4.3. ALEXI data

**ALEXI data directory:** specifies the location of the ALEXI ET data.

**ALEXI data resolution (in km):** specifies the resolution of the ALEXI ET data in km. Acceptable values are 4 or 10.

*Example lvt.config entry*

```
ALEXI data directory:          ./ALEXI
ALEXI data resolution (in km): 4
```

### 7.4.4. Ameriflux station observations

**Ameriflux observation directory:** specifies the location of the Ameriflux datasets. Under this directory, the Ameriflux data is expected to be organized by the station names and then under each station name directory, the Level3 Ameriflux files are expected to be staged.

**Ameriflux data level:** specifies the level of Ameriflux data (Note that only level 3 data is currently supported)

**Ameriflux station list file:** specifies the file that lists the location of the Ameriflux stations. The format of the station list is as follows:

```
#nstns
76
#stnname; location name; lat; lon; SWC1 depth; SWC2 depth; TS1 depth; TS2 depth
ARM_SGP_Burn; USARb; 35.5497; -98.0402; 10; 30; 5; 15
ARM_SGP_Control; USARc; 35.5465; -98.0401; 10; 30; 5; 15
ARM_SGP_Main; USARM; 36.6058; -97.4888; 5; 25; 5; 15
Atqasuk; USAtq; 70.4696; -157.4089; -1; -1; 0; 5
Audubon_Grasslands; USAud; 31.5907; -110.5092; 10; 20; 2; 4
Austin_Cary; USSP1; 29.7381; -82.2188; -1; -1; 0; 5
Barrow; USBrw; 71.3225; -156.6259; -1; -1; 0; 5
Bartlett_Experimental_Forest; USBar; 44.0645; -71.2881; 10; -1; 5; -1
Blodgett_Forest; USBlo; 38.8953; -120.6328; 10; 20; 5; 10
Bondville; USBo1; 40.0062; -88.2904; 5; 20; 2; 4
□
```



*Example lvt.config entry*

```
Ameriflux observation directory:    ./AmeriFlux
Ameriflux data level:              Level3
Ameriflux station list file:       ./AmeriFlux/Ameriflux_stns.txt
```

### 7.4.5. ANSA snow depth data

**ANSA snow depth observation directory:** specifies the location of the ANSA snow depth observation data (Note that this plugin handles the raw observations that go into the generation of the AFWA ANSA snow depth product)

*Example lvt.config entry*

```
ANSA snow depth observation directory: ./SNODEP
ANSA snow depth lower left lat:      -89.875
ANSA snow depth lower left lon:     -179.875
ANSA snow depth upper right lat:    89.875
ANSA snow depth upper right on:     179.875
ANSA snow depth resolution (dx):    0.25
ANSA snow depth resolution (dy):    0.25
```

### 7.4.6. ANSA SWE data

**ANSA SWE observation directory:** specifies the location of the ANSA snow water equivalent observation data (Note that this plugin handles the raw observations that go into the generation of the AFWA ANSA snow swe product)

*Example lvt.config entry*

```
ANSA SWE observation directory: ./ANSA_SWE
ANSA SWE lower left lat:       -89.875
ANSA SWE lower left lon:      -179.875
ANSA SWE upper right lat:     89.875
ANSA SWE upper right on:     179.875
ANSA SWE resolution (dx):     0.25
ANSA SWE resolution (dy):     0.25
```

### 7.4.7. ARM station observations

**ARM observation directory:** specifies the location of the ARM datasets

**ARM site identifier name:** specifies the text identifier (e.g. sgp, twp, nsa, etc.)

**ARM station list file:** specifies the file that lists the location of the ARM stations. The format of the station list is as follows:

```

#stns
22
#stnname; lat; lon
E1; 38.202; -99.316
E2; 38.306; -97.301
E3; 38.201; -95.597
E4; 37.953; -98.329
E5; 38.114; -97.513
E6; 37.842; -97.020
E7; 37.383; -96.180
E8; 37.333; -99.309
E9; 37.133; -97.266
□

```

**ARM use BAEBBR data:** specifies if to use the BAEBBR data or not

**ARM use EBBR data:** specifies if to use the EBBR data or not

**ARM use ECOR flux data:** specifies if to use the ECOR data or not

**ARM use SWATS data:** specifies if to use the SWATS data or not

**ARM use SMOS data:** specifies if to use the SMOS data or not

*Example lvt.config entry*

```

ARM observation directory:      ./ARM_SGP
ARM site identifier name:      sgp
ARM station list file:         ./ARM_SGP/sgp_stns.txt
ARM use BAEBBR data:          1
ARM use EBBR data:            1
ARM use ECOR flux data:       1
ARM use SWATS data:           1
ARM use SMOS data:            1

```

#### 7.4.8. USDA ARS soil moisture observations

**ARS soil moisture observation directory:** specifies the location of the ARS soil moisture observation data

**ARS soil moisture station list file:** specifies the file that lists the location of the ARS stations. The format of the file is as follows: ARS station name, station lat, station lon

```
#nstns
4
#stns
wg 31.73 -110.05
lr 31.50 -83.550
lw 34.95 -97.983
rc 43.07 -116.75
```

*Example lvt.config entry*

```
ARS soil moisture observation directory: ./ARS_Watersheds
ARS soil moisture station list file:     ./ARS_Watersheds/stnlist.dat
```

### 7.4.9. SNODAS snow analysis data

**SNODAS observation directory:** specifies the location of the SNODAS data

*Example lvt.config entry*

```
SNODAS observation directory:     ./SNODAS
```

### 7.4.10. CPC precipitation data

**CPC PCP observation directory:** specifies the location of the CPC unified precipitation data

**CPC PCP domain type (CONUS or GLOBAL):** specifies the version of the CPC unified precipitation data

**CPC PCP use real time data:** specifies whether to use the real time version of the CPC unified precipitation data

*Example lvt.config entry*

```
CPC PCP observation directory:     ../CPC_CONUS
CPC PCP domain type (CONUS or GLOBAL):  CONUS
CPC PCP use real time data:         0
```

### 7.4.11. GPM IMERG precipitation data

**IMERG data directory:** specifies the location of the GPM IMERG precipitation product

**IMERG version:** specifies the version of the GPM IMERG precipitation product. Most current version is V06B.

**IMERG product:** specifies the GPM IMERG precipitation product. Current options are: early, late, or final.

*Example lvt.config entry*

```
IMERG data directory:      ../IMERG
IMERG version:             V06B
IMERG product:            final
```

### 7.4.12. ESA CCI soil moisture data

**ESA CCI soil moisture data directory:** specifies the location of the ESA CCI soil moisture data

**ESA CCI soil moisture data version:** specifies the version of the ESA CCI data (1 or 2)

*Example lvt.config entry*

```
ESA CCI soil moisture data directory: ../ECV_sm_v2.0/
ESA CCI soil moisture data version:   2
```

### 7.4.13. Gridded FLUXNET MTE data

**FLUXNET data directory:** specifies the location of the gridded FLUXNET MTE data

*Example lvt.config entry*

```
FLUXNETmte data directory:      ../FLUXNET
```

### 7.4.14. Global Change Observation Mission - Water (GCOMW)

**GCOMW AMSR2 L3 soil moisture observation directory:** specifies the location of the GCOMW AMSR2 L3 soil moisture observations.

*Example lvt.config entry*

```
GCOMW AMSR2 L3 soil moisture observation directory:
```

### 7.4.15. Global Historic Climatology Network (GHCN) data

**GHCN observation directory:** specifies the location of the GHCN data

**GHCN station file:** specifies the file that lists the GHCN stations in the following format (station id, latitude, longitude, elevation)

USC00020170	33.63920	-109.3278	2792.000
USC00020678	35.23000	-111.8214	2179.900
USC00020750	36.67780	-110.5411	2220.800
USC00021001	36.21470	-112.0620	2438.400
USC00023009	35.16110	-111.7311	2171.700
USC00023828	34.74330	-111.4139	2279.900
USC00025412	34.11420	-109.8589	2237.200
□			

*Example lvt.config entry*

```
GHCN observation directory:      ./GHCN
GHCN station file:              ./GHCN/ghcnd-qc-stations.txt
```

### 7.4.16. GlobSnow data

**GlobSnow data directory:** specifies the location of the GlobSnow data.

*Example lvt.config entry*

```
GlobSnow data directory:        ./GlobSnow
```

### 7.4.17. ISMN soil moisture observations

**ISMN observation directory:** specifies the location of the ISMN (International Soil Moisture Network) soil moisture observation data

*Example lvt.config entry*

```
ISMN observation directory:      ./ISMN
```

### 7.4.18. LPRM AMSR-E soil moisture retrievals

**LPRM AMSR-E soil moisture observation directory:** specifies the location of the LPRM AMSR-E soil moisture retrievals.

**LPRM AMSR-E data version:** specifies the version of LPRM soil moisture data to be used. Acceptable values are:

Value	Description
GES-DISC	NASA GES-DISC version
V5	version 5 data (from VU)

**LPRM AMSR-E data channel:** specifies the frequency channel of the soil moisture retrieval. (This option is only needed for the GES-DISC data version.) Acceptable values are:

Value	Description
X-band	X-band retrieval
C-band	C-band retrieval

**LPRM AMSR-E use raw data:** specifies whether to use the LPRM AMSR-E raw data. (This option is only used for V5 data.) Acceptable values are:

Value	Description
0	Do not use
1	Use

*Example lvt.config entry*

```
LPRM AMSR-E soil moisture observation directory: ./LPRM-AMSRE
LPRM AMSR-E soil moisture data version: GES-DISC
LPRM AMSR-E soil moisture data channel: X-band
LPRM AMSR-E use raw data: 0
```

#### 7.4.19. MOD16A2 data

**MOD16A2 data directory:** specifies the location of the MOD16A2 - MODIS based ET data.

*Example lvt.config entry*

```
MOD16A2 data directory: ./MOD16
```

#### 7.4.20. MODIS LST

**MODIS LST data directory:** specifies the location of the MODIS LST data.

*Example lvt.config entry*

```
MODIS LST data directory:
```

#### 7.4.21. North American Soil Moisture Database (NASMD)

**NASMD observation directory:** specifies the location of the NASMD observation directory.

**NASMD coord file:** specifies the name of the NASMD coordinates file.

**NASMD number of stations:** specifies the number of NASMD stations.

Example lvt.config entry

```
NASMD observation directory: ../TAMU_NASMD
NASMD coord file:          ../TAMU_NASMD/nasmd_stations.txt
NASMD number of stations:  1289
```

## 7.4.22. Naturalized monthly streamflow data

**Naturalized streamflow observation directory:** specifies the location of the naturalized monthly streamflow observation data

**Naturalized streamflow station list file:** lists the locations of the streamflow stations in the following format (total number of stations followed by each usgs station id, latitude, longitude, data coverage begin year, data coverage end year).

```
#nstns
23
#name, lat, lon
ala 31.55 -87.51 1950 1993
apa 29.95 -85.02 1950 1993
del 39.69 -75.69 1948 1987
ftp 48.04 -106.36 1950 2009
gar 47.39 -101.39 1950 2009
gre 40.91 -109.42 1905 2006
□
```

Example lvt.config entry

```
Naturalized streamflow observation directory: ./Naturalized_StreamFlow
Naturalized streamflow station list file:
./Naturalized_StreamFlow/Naturalized_stnlist.txt
```

## 7.4.23. NLDAS2

**NLDAS2 data directory:** specifies the location of the NLDAS-2 data directory.

**NLDAS2 hourly or monthly:** specifies if the NLDAS-2 files to be read in are hourly or are monthly-averaged.

**NLDAS2 land surface model:** specifies the NLDAS-2 forcing or land-surface model dataset. Acceptable values are:

Value	Description
FORCINGA	NLDAS-2 primary forcing
FORCINGB	NLDAS-2 secondary forcing
NOAH	NLDAS-2 Noah LSM

Value	Description
MOS	NLDAS-2 Mosaic LSM
VIC	NLDAS-2 VIC LSM
SAC	NLDAS-2 SAC LSM (hourly only)
SACSM	NLDAS-2 SAC LSM soil moisture variables (hourly only)
NOAHST	NLDAS-2 Noah LSM streamflow variables (hourly only)
MOSST	NLDAS-2 Mosaic LSM streamflow variables (hourly only)
VICST	NLDAS-2 VIC LSM streamflow variables (hourly only)
SACST	NLDAS-2 SAC LSM streamflow variables (hourly only)

**NLDAS2 analysis data class:** specifies the type of NLDAS-2 data that is being analyzed. Acceptable values are:

Value	Description
LSM	LSM variables
Routing	Routing variables

**NLDAS2 soil moisture volumetric:** specifies if LVT should convert the NLDAS-2 soil moisture values in amounts (kg m<sup>-2</sup>) to volumetric (m<sup>3</sup> m<sup>-3</sup>), when **LVT running mode:** is set to “Observation processing”. If **NLDAS2 soil moisture volumetric:** is set to “.true.”, then the values will be converted.

**NLDAS2 VIC soil depth1 file:** specifies the name of the VIC soil depth 1 binary file.

**NLDAS2 VIC soil depth2 file:** specifies the name of the VIC soil depth 2 binary file.

**NLDAS2 VIC soil depth3 file:** specifies the name of the VIC soil depth 3 binary file.

*Example lvt.config entry*

```
NLDAS2 data directory:      ./NLDAS2.VIC
NLDAS2 hourly or monthly:  hourly
NLDAS2 land surface model:  VIC
NLDAS2 analysis data class: LSM
NLDAS2 soil moisture volumetric: .true.
NLDAS2 VIC soil depth1 file: ./vic_depth1.1gd4r
NLDAS2 VIC soil depth2 file: ./vic_depth2.1gd4r
NLDAS2 VIC soil depth3 file: ./vic_depth3.1gd4r
```



## 7.4.24. Soil Moisture Operational Processing System (SMOPS)

**SMOPS soil moisture observation directory:** specifies the location of the SMOPS soil moisture observation directory.

**SMOPS soil moisture use ASCAT data:** specifies whether to use the ASCAT data. Acceptable values are:

Value	Description
0	do not use
1	use

**SMOPS soil moisture use WindSat data:** specifies whether to use the WindSat data. Acceptable values are:

Value	Description
0	do not use
1	use

**SMOPS soil moisture use SMOS data:** specifies whether to use the SMOS data. Acceptable values are:

Value	Description
0	do not use
1	use

*Example lvt.config entry*

```
SMOPS soil moisture observation directory: ./SMOPS
SMOPS soil moisture use ASCAT data:      1
SMOPS soil moisture use WindSat data:    0
SMOPS soil moisture use SMOS data:      0
```

## 7.4.25. SMOS L1 TB

**SMOS L1 TB observation directory:** specifies the location of the SMOS L1 TB observation directory.

*Example lvt.config entry*

```
SMOS L1 TB observation directory:
```

## 7.4.26. SMOS L2 soil moisture

**SMOS L2 soil moisture observation directory:** specifies the location of the SMOS L2 soil moisture observation directory.

*Example lvt.config entry*

```
SMOS L2 soil moisture observation directory:
```

### 7.4.27. SNODEP metobs snow depth data

**SNODEP metobs directory:** specifies the location of the SNODEP observation data (Note that this plugin handles the raw observations that go into the generation of the AFWA SNODEP product)

*Example lvt.config entry*

```
SNODEP metobs directory:      ./SNODEP
```

### 7.4.28. SNOTEL SWE observations

**SNOTEL observation directory:** specifies the location of the SNOTEL SWE observation data

**SNOTEL coord file:** specifies the file that lists the location of the SNOTEL stations. The format of the station list is as follows:

```
AZ      BAKER BUTTE      11R06S      308      34.450      -111.400
AZ      BAKER BUTTE SMT 11R07S      1140     34.450      -111.367
AZ      BALDY        09S01S      310      33.967      -109.500
AZ      BEAVER HEAD  09S06S      902      33.683      -109.200
[]
```

*Example lvt.config entry*

```
SNOTEL observation directory:  ./SNOTEL
SNOTEL coord file:           ./SNOTEL/SNOTEL_CONUS_list.txt
```

### 7.4.29. SURFRAD observations (radiation, wind speed, pressure)

**SURFRAD observation directory:** specifies the location of the SURFRAD radiation data

*Example lvt.config entry*

```
SURFRAD observation directory:  ./SURFRAD
```

### 7.4.30. SCAN soil moisture observations

**SCAN observation directory:** specifies the location of the SCAN soil moisture observation data (Note that the plugin handles the “reprocessed” SCAN data from NASA GMAO.)

**SCAN number of stations:** specifies the number of SCAN stations used in the analysis

**SCAN coord file:** specifies the file that lists the location of the SCAN stations. The format of the

metadata file is as follows: station id, station lat, station lon, elevation, SCAN state

```
2058 34.43 -87 633 AL
2030 34.85 -91.88 250 AR
2091 34.28 -91.35 197 AR
2026 31.73 -110.05 4500 AZ
2013 33.88 -83.43 770 GA
2027 31.5 -83.55 350 GA
[]
```

*Example lvt.config entry*

```
SCAN observation directory: ./SCAN
SCAN number of stations: 37
SCAN coord file: ./SCAN_coord.txt
```

### 7.4.31. USGS ground water

**USGS ground water (well data) observation directory:** specifies the location of the USGS ground water (well data) observation directory.

**USGS ground water (well data) coord file:** specifies the name of the USGS ground water (well data) coordinates file.

*Example lvt.config entry*

```
USGS ground water (well data) observation directory: ./USGS_GW_welldata
USGS ground water (well data) coord file: ./USGS_GW_welldata/Well_mdata.txt
```

### 7.4.32. USGS daily streamflow data

**USGS streamflow observation directory:** specifies the location of the USGS daily streamflow observation data

**USGS streamflow station list file:** lists the locations of the streamflow stations in the following format (total number of stations followed by each usgs station id, latitude, longitude).

```

#nstns
961
#name, lat, lon
01010000 46.70060 -69.71560
01010500 47.11310 -69.08810
01011000 47.06970 -69.07940
01013500 47.23750 -68.58280
01022500 44.60810 -67.93530
01030500 45.50110 -68.30580
01031500 45.17500 -69.31470
01038000 44.22280 -69.59390
01047000 44.86920 -69.95500
01052500 44.87750 -71.05750
01054200 44.39060 -70.97970
[]

```

*Example lvt.config entry*

```

USGS streamflow observation directory: ./USGS_StreamFlow
USGS streamflow station list file:    ./USGS_StreamFlow/USGS_stnlist_nldas.txt

```

### 7.4.33. University of Washington ET data

**UW ET data directory:** specifies the location of the UWET ET data.

*Example lvt.config entry*

```

UW ET data directory:                ./UW_ET_MODIS-SRB_Monthly

```

### 7.4.34. GIMMS AVHRR NDVI data

**GIMMS AVHRR NDVI data directory:** specifies the location of the GIMMS NDVI data.

*Example lvt.config entry*

```

GIMMS AVHRR NDVI data directory:    ./GIMMS-3g/

```

### 7.4.35. GIMMS MODIS NDVI data

**GIMMS MODIS NDVI data directory:** specifies the location of the GIMMS MODIS NDVI data.

*Example lvt.config entry*

```

GIMMS MODIS NDVI data directory:    ./GIMMS_MODIS_NDVI

```

### 7.4.36. GLDAS version 2 data

**GLDAS2 data directory:** specifies the location of the GLDAS2 data.

**GLDAS2 data model name:** specifies the name of the land surface model used in the GLDAS2 data.

*Example lvt.config entry*

```
GLDAS2 data directory:      ./GLDAS2/  
GLDAS2 data model name:    NOAH
```

### 7.4.37. MERRA version 2 data

**MERRA2 data directory:** specifies the location of the MERRA2 data.

*Example lvt.config entry*

```
MERRA2 data directory:      ./MERRA2/
```

### 7.4.38. ERA Interim Land data

**ERA interim land data directory:** specifies the location of the ERA Interim Land data.

*Example lvt.config entry*

```
ERA interim land data directory:      ./ERA_Interim_Land/
```

### 7.4.39. ERA5 data

**ERA5 data directory:** specifies the location of the ERA5 data.

**ERA5 forcing tile to grid mapping file:** specifies the location of the ERA5 forcing tile to grid mapping file.

*Example lvt.config entry*

```
ERA5 data directory:          ./ERA5/  
ERA5 forcing tile to grid mapping file:  ./ERA5/mapping.nc
```

### 7.4.40. SSEB operational data

**SSEBop data directory:** specifies the location of the SSEB data.

**SSEBop process anomaly data:** specifies whether to process the ET anomaly data or the raw ET data.

*Example lvt.config entry*

```
SSEBop data directory:      ./SSEB/ETA/  
SSEBop process anomaly data: 1
```

### 7.4.41. GRDC data

**GRDC data directory:** specifies the location of the GRDC data.

**GRDC station list file:** specifies the file that contains the listing of the GRDC stations

**GRDC frequency of data:** specifies the frequency of the GRDC data (monthly/daily)

**GRDC file version:** specifies the version of the GRDC data (1 or 2)

*Example lvt.config entry*

```
GRDC data directory:      ../GRDC_data  
GRDC station list file:  ../GRDC_data/GRDC_stations.txt  
GRDC frequency of data:  monthly  
GRDC file version:       1
```

### 7.4.42. LIS DAOBS output as the observation

**LIS DAOBS output directory:** specifies the LIS DAOBS output directory

*Example lvt.config entry*

```
LIS DAOBS output directory:      ./DAOBS
```

**LIS DAOBS use scaled obs:** specifies whether to use scaled obs. Acceptable values are:

Value	Description
0	Do not use scaled obs
1	Use scaled obs

*Example lvt.config entry*

```
LIS DAOBS use scaled obs: 0
```

### 7.4.43. ISCCP land surface temperature observations

**ISCCP Tskin data directory:** specifies the location of the ISCCP land surface temperature data

*Example lvt.config entry*

```
ISCCP Tskin data directory:      ./ISCCP
```

## 7.4.44. LSWG Tb observations

**LSWG Tb observation filename:** specifies the name of the LSWG filename containing Brightness Temperature (Tb) observations

**LSWG Tb satellite name:** specifies the name of satellite — same as what's used in CRTM

**LSWG Tb data format:** 0 for AMSR-E, 1-for AMSU

**LSWG Tb metadata file:** specifies the file that lists the metadata for LSWG Tb observations. The format of the metadata file is as follows:

```
#nstns, undef, starting time, ending time, timestep (mins)
1 -1 2006 07 01 10 00 2007 06 30 17 00 3600
#LIS channel data index in file
1 1
2 2
3 3
4 4
5 5
6 6
7 7
8 8
9 9
10 10
11 11
12 12
13 13
14 14
15 15
```

**LSWG Tb include cloud masking:** specifies if data is to be ignored in the presence of clouds (0-do not ignore, 1-ignore)

**LSWG Tb cloud mask file:** specifies the name of the cloud mask file

**LSWG Tb cloud mask column:** specifies what?

**LSWG Tb cloud mask threshold(%):** specifies the threshold below which clouds can be ignored (used only if cloud masking is enabled).

*Example lvt.config entry*

```
LSWG Tb observation filename:      ./_LSWG/C3VP.txt
LSWG Tb satellite name:           "N18_"
LSWG Tb data format:              1
LSWG Tb metadata file:            ./C3VP_mdata
LSWG Tb include cloud masking:    1
LSWG Tb cloud mask file:          ./cloud_mask.txt
LSWG Tb cloud mask column:        ??
LSWG Tb cloud mask threshold(%): 75
```

#### 7.4.45. FMI SWE observations

**FMISWE observation directory:** specifies the location of the FMI snow course data

*Example lvt.config entry*

```
FMISWE observation directory:      ./FMI_SWE
```

#### 7.4.46. CMC daily snow depth observations

**CMC snow depth observation directory:** specifies the location of the CMC snow depth observation data

*Example lvt.config entry*

```
CMC snow depth observation directory:  ./CMC_data
```

#### 7.4.47. NASA AMSR-E soil moisture retrievals

**NASA AMSR-E soil moisture observation directory:** specifies the location of the standard (NASA) AMSR-E soil moisture retrievals

*Example lvt.config entry*

```
NASA AMSR-E soil moisture observation directory: ./NASA_AMSRE
```

#### 7.4.48. AMMA station observations

**AMMA observation directory:** specifies the location of the AMMA in-situ observations.

**AMMA static txt file list:** specifies the file with the station file names in text format (.txt)

A sample static txt file list is shown below:



```
12
201006140332132535.csv
201006140337342536.csv
201006140347082537.csv
201006140348592538.csv
201006140351382539.csv
201006140358582540.csv
201006140400532541.csv
201006140402202542.csv
201006140640302543.csv
201006140641442544.csv
201006140642422545.csv
201006140643372546.csv
```

**AMMA static netcdf file list:** specifies the file with the station file names in netcdf format

A sample static txt file list is shown below:

```
10
ceh-aws_agoufou_20050414.nc
ceh-aws_bamba_20050426.nc
ceh-aws_banizoumbou_20051115.nc
ceh-aws_belifougou_20051111.nc
ceh-aws_bira_20051113.nc
ceh-aws_hedgerit_20050415.nc
ceh-aws_kelema_20050416.nc
ceh-aws_nalohou_20051111.nc
ceh-aws_pobe_20050220.nc
ceh-aws_wankama_20051117.nc
```

**AMMA soil moisture layer weights:** normalized weights to be applied for root zone computations of soil moisture

**AMMA soil temperature layer weights:** normalized weights to be applied for root zone computations of soil temperature

*Example lvt.config entry*

```
AMMA observation directory:      ./AMMA
AMMA static txt file list:      amma_static_txtfiles.txt
AMMA static netcdf file list:   amma_static_ncfiles.txt
AMMA soil moisture layer weights: 0.1875 0.1875 0.625 0.0 0.0
AMMA soil temperature layer weights: 0.1875 0.1875 0.625 0.0 0.0
```

#### 7.4.49. SMOSREX in-situ soil moisture observations

**SMOSREX observation filename:** specifies the name of the SMOSREX observation filename. Currently this plugin only handles a single observation location.

*Example lvt.config entry*

```
SMOSREX observation filename:      ./SMOSREX/Toulouse_SMOSREX.dat
```

### 7.4.50. MOD10A1 snow cover data

**MOD10A1 observation directory:** specifies the location of the 1km resolution MOD10A1 fractional snow cover data

*Example lvt.config entry*

```
MOD10A1 observation directory:    ./MOD10A1
```

### 7.4.51. MOD10A1V6 snow cover data

**MOD10A1V6 observation directory:** specifies the location of the 1km resolution MOD10A1V6 fractional snow cover data

*Example lvt.config entry*

```
MOD10A1V6 observation directory:  ./MOD10A1V6
```

### 7.4.52. GRACE (processed from LDT) data

**GRACE data directory:** specifies the location of the GRACE data (note that this plugin handles the “processed” GRACE data, where the GRACE TWS anomalies have been added to the LIS model TWS).

**GRACE configuration:** specifies the GRACE configuration. Acceptable values are:

Value	Description
default	GRACE
follow-on	GRACE follow-on
GRACE-2	GRACE 2

*Example lvt.config entry*

```
GRACE data directory:  ./GRACE  
GRACE configuration:  default
```

### 7.4.53. CEOP

**CEOP data directory:** specifies the location of the CEOP data.

**CEOP station list file:** specifies the location of the CEOP station list file.

**CEOP read surface meteorology data:** specifies whether to read the CEOP surface meteorology data.

Acceptable values are:

Value	Description
0	do not read
1	read

**CEOP read flux data:** specifies whether to read the CEOP flux data. Acceptable values are:

Value	Description
0	do not read
1	read

**CEOP read soil moisture and temperature data:** specifies whether to read the CEOP soil moisture and temperature data. Acceptable values are:

Value	Description
0	do not read
1	read

*Example lvt.config entry*

```
CEOP data directory:      ./CEOP
CEOP station list file:
CEOP read surface meteorology data: 1
CEOP read flux data:      1
CEOP read soil moisture and temperature data: 1
```

#### 7.4.54. COOP

**COOP data directory:** specifies the location of the COOP data.

**COOP coord file:** specifies the name of the COOP coordinates file.

**COOP metadata file:** specifies the name of the COOP metadata file.

*Example lvt.config entry*

```
COOP data directory:
COOP coord file:
COOP metadata file:
```

#### 7.4.55. Great Lakes Environmental Research Lab (GLERL)

**GLERL hydro data directory:** specifies the location of the GLERL hydro data.

**GLERL hydro lake locations file:** specifies the name of the GLERL hydro lake locations file.

*Example lvt.config entry*

```
GLERL hydro data directory:  
GLERL hydro lake locations file:
```

### 7.4.56. GSOD

**GSOD observation directory:** specifies the location of the GSOD observation data.

**GSOD coord file:** specifies the name of the GSOD coordinates file.

**GSOD metadata file:** specifies the name of the GSOD metadata file.

*Example lvt.config entry*

```
GSOD observation directory:  
GSOD coord file:  
GSOD metadata file:
```

### 7.4.57. Plate Boundary Observatory H2O (PBOH2O)

**PBOH2O observation directory:** specifies the location of the PBOH2O observation directory.

**PBOH2O station list file:** specifies the name of the PBOH2O station list file.

*Example lvt.config entry*

```
PBOH2O observation directory:  
PBOH2O station list file:
```

### 7.4.58. WG PBMR

**WG PBMR observation directory:** specifies the location of the WG PBMR observation directory.

**WG PBMR site index:** specifies the location of the WG PBMR site index.

*Example lvt.config entry*

```
WG PBMR observation directory:  
WG PBMR site index:
```

### 7.4.59. GLEAM

**GLEAM data directory:** specifies the location of the GLEAM data directory.

**GLEAM data version:** specifies the version of the GLEAM data. Acceptable values are:

Value	Description
3.2a	version 3.2a
3.2b	version 3.2b
3.1a	version 3.1a
3.1b	version 3.1b
3.1c	version 3.1c
3.0a	version 3.0a
3.0b	version 3.0b
3.0c	version 3.0c

*Example lvt.config entry*

```
GLEAM data directory:      ./GLEAM/
GLEAM data version:       "3.2a"
```

#### 7.4.60. FLUXNET2015

**FLUXNET2015 observation directory:** specifies the location of the FLUXNET2015 data

**FLUXNET2015 station list file:** specifies the list of stations in the FLUXNET2015 data.

*Example lvt.config entry*

```
FLUXNET2015 observation directory: ./FLUXNET2015_STN/
FLUXNET2015 station list file:     ./FLUXNET2015_STN/STN_list.dat
```

#### 7.4.61. USCRN soil moisture

**USCRN soil moisture observation directory:** specifies the location of the USCRN soil moisture data

**USCRN soil moisture station file:** specifies the list of stations in the USCRN data.

*Example lvt.config entry*

```
USCRN soil moisture observation directory:  ./USCRN
USCRN soil moisture station file:          ./USCRN/USCRN_stations.txt
```

#### 7.4.62. SMAP soil moisture

**SMAP soil moisture observation directory:** specifies the location of the SMAP soil moisture observation directory.

**SMAP soil moisture data designation:** specifies the designation of the SMAP data. Acceptable values are:

Value	Description
SPL3SMP	SMAP L3 Radiometer Global Daily 36 km EASE-Grid Soil Moisture
SPL3SMP_E	SMAP Enhanced L3 Radiometer Global Daily 9 km EASE-Grid Soil Moisture
SPL2SMP	SMAP L2 Radiometer Global Daily 36 km EASE-Grid Soil Moisture
SPL2SMP_E	SMAP Enhanced L2 Radiometer Global Daily 9 km EASE-Grid Soil Moisture

*Example lvt.config entry*

```
SMAP soil moisture observation directory: ../SPL3SMP.005
SMAP soil moisture data designation:     SPL3SMP
```

### 7.4.63. SMAP vegetation optical depth

**SMAP vegetation optical depth observation directory:** specifies the location of the SMAP vegetation optical depth (VOD) observation directory.

**SMAP vegetation optical depth data designation:** specifies the designation of the SMAP data. Acceptable values are:

Value	Description
SPL3SMP	SMAP L3 Radiometer Global Daily 36 km EASE-Grid VOD
SPL3SMP_E	SMAP Enhanced L3 Radiometer Global Daily 9 km EASE-Grid VOD

*Example lvt.config entry*

```
SMAP vegetation optical depth observation directory: ../SPL3SMP.005
SMAP vegetation optical depth data designation:     SPL3SMP
```

### 7.4.64. LPRM vegetation optical depth

**LPRM vegetation optical depth observation directory:** specifies the location of the LPRM vegetation optical depth (VOD) observation directory.

**LPRM vegetation optical depth data designation:** specifies the designation of the LPRM data. Acceptable values are:

Value	Description
X-band	X-band retrievals of VOD

Value	Description
C-band	C-band retrievals of VOD

*Example lvt.config entry*

```
LPRM vegetation optical depth observation directory: ../LPRM_VOD
LPRM vegetation optical depth data designation:      X-band
```

## 7.4.65. WG SWRC

**WG SWRC observation directory:** specifies the location of the WG SWRC observation directory.

**WG SWRC station list file:** specifies the name of the WG SWRC station list file.

*Example lvt.config entry*

```
WG SWRC observation directory:
WG SWRC station list file:
```

## 7.4.66. UA SNOW observations

**UA SNOW observation directory:** specifies the location of the University of Arizona SWE data.

*Example lvt.config entry*

```
UA SNOW observation directory:      ./UA_SNOW
```

## 7.4.67. OzFlux soil moisture

**OzFlux observation directory:** specifies the location of the OzFlux data directory.

**OzFlux station list file:** specifies the location of the OzFlux stations.

A sample file is shown below:

```
#Nstns
9
#names
CapeTribulation; -16.103219; 145.446922
CumberlandPlainMelaleuca; -33.613996; 150.726418
Longreach ; -23.5232657644973; 144.310415719645
Riggs; -36.656; 145.576
Robson; -17.11746943; 145.6301375
Tumbarumba; -35.6566; 148.1516
Whroo; -36.6732; 145.0294
WombatStateForest; -37.4222; 144.0944
Yanco; -34.9878; 146.2908
```

*Example lvt.config entry*

```
OzFlux observation directory: ../OzFlux
OzFlux station list file:    ../OzFlux/stnlist.dat
```

## 7.4.68. JASMIN soil moisture

**JASMIN soil moisture data directory:** specifies the location of the JASMIN soil moisture data directory.

*Example lvt.config entry*

```
JASMIN soil moisture data directory: ../JASMIN
```

## 7.5. OptUE processing options

This section of the config file specifies the details of the optimization and uncertainty estimation processing options, and the specialized options to analyze outputs from the Optimization/Uncertainty Estimation algorithms.

**LIS OptUE restart file:** the name of the file that specifies the parameter distributions LVT expects this information to be provided through the uncertainty estimation algorithm restart file. Note that this option needs to be specified only if ensemble cross correlation metric is enabled.

*Example lvt.config entry*

```
LIS OptUE restart file:          MCSIM.001.MCSIMrst
```

**LIS OptUE number of model parameters:** specifies the number of model parameters in the uncertainty estimation algorithm restart file. Note that this option needs to be specified only if ensemble cross correlation metric is enabled.

*Example lvt.config entry*

```
LIS OptUE number of model parameters: 4
```

**OptUE output data directory:** specifies the location of the root directory containing OptUE outputs.

*Example lvt.config entry*

```
OptUE output data directory:  '../OUTPUT'
```

**OptUE algorithm used:** specifies the index of the optimization/uncertainty estimation algorithm used. Acceptable values are:



Value	Description
“Genetic algorithm”	Genetic Algorithm
“Monte carlo sampling”	Monte carlo sampling
“Random walk markov chain monte carlo”	Random walk markov chain monte carlo
“Differential evolution monte carlo”	Differential evolution monte carlo

Example *lvt.config* entry

```
OptUE algorithm used:          2
```

**OptUE decision space attributes file:** lists the decision space attributes file used in the LIS optimization/uncertainty estimation integration.

Example *lvt.config* entry

```
OptUE decision space attributes file:  ./GARun/noah_sm_decspace.txt
```

**OptUE number of iterations:** Number of generations used in the optimization/uncertainty estimation algorithm.

Example *lvt.config* entry

```
OptUE number of iterations:      20
```

**OptUE compute time series:** specifies if a time series of OptUE run output data is to be generated (0-no, 1-yes)

Example *lvt.config* entry

```
OptUE compute time series:       1
```

**OptUE time series location file:** specifies the file which lists the locations in the domain where the time series data are to be extracted. The format of the time series location file is as follows:

```
#Number of locations
1
#Location style (1-lat/lon, 2-col/row, 3-tile)
2
#mask filename
none
#site name
Site1
244 236
```

Example lvt.config entry

```
OptUE time series location file:      ./STN_LOCATIONS.DAT
```

**OptUE total number of parameters:** specifies the total number of parameters.

Example lvt.config entry

```
OptUE total number of parameters:      2
```

**OptUE total number of selected parameters:** specifies the total number of selected parameters.

Example lvt.config entry

```
OptUE total number of selected parameters: 1
```

## 7.6. DA diagnostics analysis

This section of the config file specifies the specialized options to analyze the data assimilation diagnostics. These options are employed for runmode="DA statistics processing"

**Compute innovation distribution:** Specifies if innovation distribution analysis (computing mean and variance) is to be computed. Acceptable values are:

Value	Description
0	Do not compute
1	Compute

**Compute analysis gain:** specifies if analysis gain is to be computed. Acceptable values are:

Value	Description
0	Do not compute
1	Compute

**Number of state variables in the DA update:** specifies the number of state variables in the DA update.

Example lvt.config entry

```
Compute Innovation Distribution:      1
Compute analysis gain:                 0
Number of state variables in the DA update: 4
```

## 7.7. DA observation analysis

This runmode is used to conduct analysis of observations used in the DA assimilation instance. LIS DA subsystem generates processed (interpolated, QC'd) estimates of input observations. This runmode enables analysis of such data.

No specialized options are necessary, except specifying the “LIS output attributes file:” option to correspond to the DA output. For example, if the DA instance generates estimates of a single variable (say SWE) then specify the LIS output attributes file such that (only) SWE is present in the (LIS) output file. In other words, column number 2 should indicate 1 for SWE variable and 0 for every other variable. If multiple observation types are present in the DA output, then column number 2 should be appropriately modified.

## 7.8. 557 post

This runmode is used to postprocess LIS data and generate netCDF or GRIB2 output files for use by 557WW. Specific capabilities include:

- Calculate ensemble means and standard deviations (spreads).
- Optionally apply a Gaussian filter to the ensemble products (recommended).
- Postprocess over 3 or 24 hour periods.
- Interpolate US Navy GOFS (HYCOM) sea ice and sea surface temperature data to the LIS grid.
- Output in GRIB2 or netCDF.

Due to memory constraints, only a few (typically one) LIS variable can be processed by a single invocation of LVT in this mode. However, this provides an opportunity to divide a set of LIS variables into a number of independent jobs that can be run in parallel, thus reducing the overall runtime. For GRIB2 output, the files can then be combined using the standard Unix `cat` command. For netCDF, it is necessary to use the NCO tool `ncks` to merge the postprocessed fields together into a single file.

These capabilities are activated by setting LVT running mode to “557 post”

When using this runmode:

- The LVT output format must be set to either “netcdf” or “grib2”.
- The LVT output methodology should be “2d ensemble gridspace”.
- The first analysis data source must be “LIS output”.
- The process HYCOM data setting should only be turned on for a single LVT invocation — otherwise, multiple copies of the HYCOM data may be appended afterwards into a single GRIB2 or netCDF file.
- It is recommended that the LIS output data be in “1d tilespace” to reduce the time required to read and map the data to LVT’s internal structures.
- It is recommended that each LVT invocation have its own specified metrics output directory, to prevent file overwrites from different LVT invocations running in parallel.

It is recommended that scripts be used to customize *lvt.config* files for each LVT invocation, and then run LVT in individual batch modes. Example scripts are provided in *lvt/util/afwa/templates*, along with customized *MODEL\_OUTPUT\_LIST.TBL* files for use by 557WW. Sample scripts to automate merging of GRIB2 or netCDF output files are given in *lvt/utills/afwa*.

**Process HYCOM data:** Specifies if HYCOM (US Navy GOFS) data are processed in addition to LIS data. Acceptable values are:

Value	Description
0	Do not process
1	Process

*Example lvt.config entry*

```
Process HYCOM data: 1
```

**HYCOM data directory:** Specifies directory path to HYCOM (US Navy GOFS) data

*Example lvt.config entry*

```
HYCOM data directory: HYCOM
```

**Apply noise reduction filter:** Specifies whether to smooth the postprocessed fields before writing to file. Acceptable values are:

Value	Description
0	Do not smooth
1	Smooth

*Example lvt.config entry*

```
Apply noise reduction filter: 1
```

**Smoothing filter type:** Specifies which smoothing filter to use, provided smoothing is activated. Acceptable values are:

Value	Description
“box filter”	Box linear filter (or box blur)
“gaussian filter”	Gaussian filter (recommended)

*Example lvt.config entry*

```
Smoothing filter type: "gaussian filter"
```

## 7.9. USAFSI post

This runmode is used to postprocess USAFSI (snow and ice) analyses, which are originally produced in netCDF format. LVT will optionally convert the fields to GRIB2 format and output on the LDT grid (no interpolations). In addition, three options are provided for GRIB1 output: (1) interpolate to 0.25 deg lat/lon domain; (2) interpolate to Air Force Northern Hemisphere 16th mesh polar stereographic grid; and (3) interpolate to Air Force Southern Hemisphere 16th mesh polar stereographic grid. These GRIB1 products are intended as look-alike replacements for the legacy Air Force SNODEP product. These options are activated for runmode="USAFSI post"

When run in this mode, LVT will interpret the starting year, month, etc as the valid date/time to process, and will ignore the ending date and time information. (In other words, only a single date and time will be processed.)

LVT will also use the diagnostic file and undefined value entries.

Other entries that are required but not actually used in this runmode include the map projection, analysis sources, output format, output GRIB settings, the output methodology, start mode, clock timestep, metrics computation and output frequencies, option to apply temporal smoothing to obs, option to apply external mask, option for stratifying by external data and/or variable, and an observation count threshold. These other entries are required by the LVT config initialization routine, which is called by the LVT driver for all runmodes.

**USAFSI output GRIB2 native grid:** Specifies if USAFSI should be output in GRIB2 (no interpolation). Acceptable values are:

Value	Description
.true.	Output in GRIB2
.false.	Do not output in GRIB2

*Example lvt.config entry*

```
USAFSI output GRIB2 native grid: .true.
```

**USAFSI output GRIB1 global 0.25deg lat/lon:** Specifies if USAFSI should be interpolated to 0.25 deg lat/lon grid and output in GRIB1. Acceptable values are:

Value	Description
.true.	Interpolate and output in GRIB1
.false.	Do not interpolate or output

*Example lvt.config entry*

```
USAFSI output GRIB1 global 0.25deg lat/lon: .true.
```

**USAFSI output GRIB1 NH 16th mesh polar stereographic:** Specifies if USAFSI should be interpolated

to Northern Hemispheric 16th mesh polar stereographic grid and output in GRIB1. Acceptable values are:

Value	Description
.true.	Interpolate and output in GRIB1
.false.	Do not interpolate or output

*Example lvt.config entry*

```
USAFSI output GRIB1 NH 16th mesh polar stereographic: .true.
```

**USAFSI output GRIB1 SH 16th mesh polar stereographic:** Specifies if USAFSI should be interpolated to Southern Hemispheric 16th mesh polar stereographic grid and output in GRIB1. Acceptable values are:

Value	Description
.true.	Interpolate and output in GRIB1
.false.	Do not interpolate or output

*Example lvt.config entry*

```
USAFSI output GRIB1 SH 16th mesh polar stereographic: .true.
```

**USAFSI input netcdf directory:** Specifies path of directory containing USAFSI netcdf files to process.

*Example lvt.config entry*

```
USAFSI input netcdf directory: ./USAFSI
```

**USAFSI output grib directory:** Specifies path of directory to write GRIB2 and/or GRIB1 files.

*Example lvt.config entry*

```
USAFSI output grib directory: ./grib
```

# Chapter 8. Configuration of metrics

This section defines the specification of various metrics in LVT. This file is specified in a space delimited column format. Each row consists of the following entries:

**Name:** Name of the metric

**Use option:** determines whether to use this metric. When enabled, the metric will be computed through the duration of the evaluation and a final file will be written out. Acceptable values are:

Value	Description
0	do not use the metric
1	use the metric

**Time option:** specifies whether to compute the metric in time, at the specified stats output intervals. Acceptable values are:

Value	Description
0	do not compute the metric
1	compute the metric

**Temporal output:** determines whether to write (gridded) metric files at the specified stats output intervals. The “Time option” must also be enabled when this option is enabled. Acceptable values are:

Value	Description
0	do not write gridded files
1	write gridded time files

**Extract time series:** determines whether to extract (ASCII) time series files for the metric, at each sub-domain specified in the time series location file. Acceptable values are:

Value	Description
0	do not write ASCII files
1	write ASCII time files

**Threshold:** specifies the threshold values to be used in computing the metric. Note that this is used only for the categorical metrics. Two values must be specified. The first value is the minimum threshold and the second values is the maximum threshold.

**Compute average seasonal cycle:** determines whether to generate the average seasonal cycle of the metric (for each domain specified in the time series location file). Acceptable values are:

Value	Description
0	do not generate
1	generate seasonal cycle

**Compute average diurnal cycle:** determines whether to generate the average diurnal cycle of the metric (for each domain specified in the time series location file). Acceptable values are:

Value	Description
0	do not generate
1	generate diurnal cycle

#name	Use	Time	WriteTS	ExtractTS		
Threshold(min/max)	ASC	ADC	short_name			
Mean:	1	1	1	1	-9999.0	-9999.0
1 0 #MEAN						
Anomaly:	0	0	0	0	-9999.0	-9999.0
0 0 #Anomaly						
Min:	0	0	0	0	-9999.0	-9999.0
0 0 #MIN						
Max:	0	0	0	0	-9999.0	-9999.0
0 0 #MAX						
MinTime:	0	0	0	0	-9999.0	-9999.0
0 0 #MINTIME						
MaxTime:	0	0	0	0	-9999.0	-9999.0
0 0 #MAXTIME						
Sum:	0	0	0	0	-9999.0	-9999.0
0 0 #SUM						
Standard deviation:	0	0	0	0	-9999.0	-9999.0
0 0 #STDEV						
Variance:	0	0	0	0	-9999.0	-9999.0
0 0 #VARIANCE						
Bias:	0	0	0	0	-9999.0	-9999.0
0 0 #BIAS						
RMSE:	0	0	0	0	-9999.0	-9999.0
0 0 #RMSE						
ubRMSE:	0	0	0	0	-9999.0	-9999.0
0 0 #ubRMSE						
Anomaly RMSE:	0	0	0	0	-9999.0	-9999.0
0 0 #ARMSE						
Mean absolute error:	0	0	0	0	-9999.0	-9999.0
0 0 #MAE						
Anomaly correlation:	0	0	0	0	-9999.0	-9999.0
0 0 #ACORR						
Raw correlation:	0	0	0	0	-9999.0	-9999.0
0 0 #RCORR						
Rank correlation:	0	0	0	0	-9999.0	-9999.0
0 0 #RNKCORR						
Anomaly rank correlation:	0	0	0	0	-9999.0	-9999.0



0	0	#ARNKCORR					
Probability of detection (PODy):	0	0	0	0	0	0.1	0.3
0	0	#PODY					
Probability of detection (PODn):	0	0	0	0	0	0.1	0.3
0	0	#PODN					
False alarm ratio (FAR):	0	0	0	0	0	0.1	0.3
0	0	#FAR					
Detection failure ratio (DFR):	0	0	0	0	0	0.1	0.3
0	0	#DFR					
Probability of false detection (POFD):	0	0	0	0	0	0.1	0.3
0	0	#POFD					
Critical success index (CSI):	0	0	0	0	0	0.1	0.3
0	0	#CSI					
Accuracy measure (ACC):	0	0	0	0	0	0.1	0.3
0	0	#ACC					
Frequency bias (FBIAS):	0	0	0	0	0	0.1	0.3
0	0	#FBIAS					
Equitable threat score (ETS):	0	0	0	0	0	0.1	0.3
0	0	#ETS					
Event frequency (EF):	0	0	0	0	0	0.1	0.3
0	0	#EF					
Forecast frequency (FF):	0	0	0	0	0	0.1	0.3
0	0	#FF					
Heidke skill score (HSS):	0	0	0	0	0	0.1	0.3
0	0	#HSS					
Peirce skill score (PSS):	0	0	0	0	0	0.1	0.3
0	0	#PSS					
Clayton skill score (CSS):	0	0	0	0	0	0.1	0.3
0	0	#CSS					
Area metric:	0	0	0	0	0	-9999.0	-9999.0
0	0	#AREA					
Nash sutcliffe efficiency:	0	0	0	0	0	-9999.0	-9999.0
0	0	#NSE					
Metric entropy:	0	0	0	0	0	-9999.0	-9999.0
0	0	#Mentropy					
Information gain:	0	0	0	0	0	-9999.0	-9999.0
0	0	#Igain					
Fluctuation complexity:	0	0	0	0	0	-9999.0	-9999.0
0	0	#Fcomplexity					
Effective complexity:	0	0	0	0	0	-9999.0	-9999.0
0	0	#Ecomplexity					
Wavelet stat:	0	0	0	0	0	-9999.0	-9999.0
0	0	#Waveletstat					
Hausdorff norm:	0	0	0	0	0	-9999.0	-9999.0
0	0	#Hnorm					
Standard precipitation index:	0	0	0	0	0	-9999.0	-9999.0
0	0	#SPI					
Standard runoff index:	0	0	0	0	0	-9999.0	-9999.0
0	0	#SRI					
Standard dS index:	0	0	0	0	0	-9999.0	-9999.0
0	0	#SdSI					

Standardized soil water index:	0	0	0	0	-9999.0	-9999.0
0 0 #SSWI						
Standardized ground water index:	0	0	0	0	-9999.0	-9999.0
0 0 #SGWI						
Percentile:	0	0	0	0	-9999.0	-9999.0
0 0 #Percentile						
River flow variate:	0	0	0	0	-9999.0	-9999.0
0 0 #RFV						
Tendency:	0	0	0	0	-9999.0	-9999.0
0 0 #TENDENCY						
Tendency correlation:	0	0	0	0	-9999.0	-9999.0
0 0 #TENDENCYCORR						
Z score:	0	0	0	0	-9999.0	-9999.0
0 0 #ZSCORE						
Trend:	0	0	0	0	-9999.0	-9999.0
0 0 #TREND						
Triple collocation:	0	0	0	0	-9999.0	-9999.0
0 0 #TC						
Reliability:	0	0	0	0	-9999.0	-9999.0
0 0 #REL						
Resilience:	0	0	0	0	-9999.0	-9999.0
0 0 #RES						
Vulnerability:	0	0	0	0	-9999.0	-9999.0
0 0 #VUL						
Information entropy:	0	0	0	0	-9999.0	-9999.0
0 0 #IE						
Conditional entropy:	0	0	0	0	-9999.0	-9999.0
0 0 #CE						
Joint entropy:	0	0	0	0	-9999.0	-9999.0
0 0 #JE						
Relative entropy:	0	0	0	0	-9999.0	-9999.0
0 0 #RE						
Mutual information:	0	0	0	0	-9999.0	-9999.0
0 0 #MI						

# Chapter 9. Model Output Specifications

This section defines the specification of the model output from LIS. This file is specified in a space delimited column format. Each row consists of the following entries:

**Short Name** ALMA compliant short name of the variable.

**Use option** determines whether to write the variable. Acceptable values are:

Value	Description
0	do not write the variable
1	write the variable

**Units** the desired unit of the output variable.

**Sign Convention** direction in which the variable is considered to have positive values. Note that the land models in LIS employ the “traditional approach” where all variables are considered positive in their dominant direction. i.e. precipitation and radiation are positive towards the surface (downward), evaporation, sensible heat and runoff are positive away from the surface. Acceptable values are:

Value		Description
-		No sign
UP	DN	Up or Down (Used for fluxes, Precip)
IN	OUT	In or Out of the grid cell (Used for runoff, baseflow)
INC	DEC	Increase or Decrease (Used for change in storage terms)
S2L	L2S	Solid to Liquid and Liquid to Solid (for phase change terms)
S2V	V2S	Solid to Vapor and Vapor to Solid (for phase change terms)
E	N	Eastward and Northward (used for Wind components)

**Time Average option** determines how temporally process the variable. Acceptable values are:

Value	Description
0	Instantaneous output
1	Time averaged output
2	Instantaneous and Time averaged output
3	Accumulated output

**Min/Max option** determines whether to record minimum and maximum values for the variable. For a given grid-cell, the minimum and maximum values correspond to the minimum and maximum found for all subgrid tiles and ensembles contained in the grid-cell during that output interval. Acceptable values are:

Value	Description
0	Do not compute minimum and maximum values
1	Do compute minimum and maximum values

**Standard Deviation option** determines whether to record the standard deviation of the values for the variable. For a given grid-cell, the standard deviation is a measure of the spread of the subgrid tiles and ensembles contained within the grid-cell from the grid-cell's mean. Acceptable values are:

Value	Description
0	Do not compute standard deviation
1	Do compute standard deviation

**Number of vertical levels** The number of vertical levels corresponding to the variable.

**grib ID** The grib ID to be used for the variable if output is written in grib1 format.

**grib scale factor** The grib scale factor to be used for the variable if output is written in grib1 format.

**Use in LVT option** determines whether to include the variable in the LVT analysis. Acceptable values are:

Value	Description
0	Do not include the variable
1	Do include the variable
n	Enable the variable for analysis and include additional (n) layers

Note that this is a full list of output variables. Not all models support all these variables. You must check the source code to verify that the model you want to run supports the variables that you want to write.

```
#short_name select? units signconv timeavg? min/max? std? vert.levels grib_id
grib_scalefactor LVT? longname

#Energy balance components
Swnet:      0  W/m2   DN   1 0 0 1 111 10   0 # Net shortwave radiation (W/m2)
Lwnet:      0  W/m2   DN   1 0 0 1 112 10   0 # Net longwave radiation (W/m2)
Qle:        1  W/m2   UP   1 0 0 1 121 10   1 # Latent heat flux (W/m2)
Qh:         1  W/m2   UP   1 0 0 1 122 10   1 # Sensible heat flux (W/m2)
Qg:         1  W/m2   DN   1 0 0 1 155 10   1 # Ground heat flux (W/m2)
```

Qf:	0	W/m2	S2L	1	0	0	1	229	10	0	# Energy of fusion (W/m2)
Qv:	0	W/m2	S2V	1	0	0	1	198	10	0	# Energy of sublimation (W/m2)
Qa:	0	W/m2	DN	1	0	0	1	136	10	0	# Advective energy (W/m2)
Qtau:	0	N/m2	DN	1	0	0	1	172	10	0	# Momentum flux (N/m2)
DelSurfHeat:	0	J/m2	INC	1	0	0	1	137	10	0	# Change in surface heat storage (J/m2)
DelColdCont:	0	J/m2	INC	1	0	0	1	138	10	0	# Change in snow cold content (J/m2)
BR:	0	-	-	1	0	1	1	256	10	0	# Bowen ratio
EF:	0	-	-	1	0	1	1	256	10	0	# Evaporative fraction
 #Water balance components											
Snowf:	0	kg/m2s	DN	1	0	0	1	161	10000	0	# Snowfall rate (kg/m2s)
Rainf:	0	kg/m2s	DN	1	0	0	1	162	10000	0	# Rainfall rate (kg/m2s)
RainfConv:	0	kg/m2s	DN	1	0	0	1	63	10000	0	# Convective rainfall rate (kg/m2s)
TotalPrecip:	1	kg/m2s	DN	1	0	0	1	61	10000	0	# Total precipitation rate (kg/m2s)
Evap:	0	kg/m2s	UP	1	0	0	1	57	10000	0	# Total evapotranspiration (kg/m2s)
Qs:	0	kg/m2s	OUT	1	0	0	1	235	10000	0	# Surface runoff (kg/m2s)
Qrec:	0	kg/m2s	IN	1	0	0	1	163	10000	0	# Recharge (kg/m2s)
Qsb:	0	kg/m2s	OUT	1	0	0	1	234	10000	0	# Subsurface runoff (kg/m2s)
Qsm:	0	kg/m2s	S2L	1	0	0	1	99	10000	0	# Snowmelt (kg/m2s)
Qfz:	0	kg/m2s	L2S	1	0	0	1	130	10000	0	# Refreezing of water in the snowpack (kg/m2s)
Qst:	0	kg/m2s	-	1	0	0	1	131	10000	0	# Snow throughfall (kg/m2s)
DelSoilMoist:	0	kg/m2	INC	1	0	0	1	132	10000	0	# Change in soil moisture (kg/m2)
DelSWE:	0	kg/m2	INC	1	0	0	1	133	1000	0	# Change in snow water equivalent (kg/m2)
DelSurfStor:	0	kg/m2	INC	1	0	0	1	134	1000	0	# Change in surface water storage (kg/m2)
DelIntercept:	0	kg/m2	INC	1	0	0	1	135	1000	0	# Change in interception storage (kg/m2)
RHMin:	0	-	-	1	0	0	1	52	10	0	# Minimum 2-meter relative humidity (-)
Ch:	0	-	-	1	0	0	1	208	10	0	# Surface exchange coefficient for heat
Cm:	0	-	-	1	0	0	1	252	10	0	# Surface exchange coefficient for momentum
MixRatio:	0	kg/kg	-	1	0	0	1	53	10	0	# Mixing ratio
 #Surface state variables											
SnowT:	0	K	-	1	0	0	1	165	10	0	# Snow surface temperature (K)
VegT:	0	K	-	1	0	0	1	146	10	0	# Vegetation canopy temperature (K)
BareSoilT:	0	K	-	1	0	0	1	147	10	0	# Temperature of bare soil (K)
AvgSurfT:	0	K	-	1	0	0	1	148	10	0	# Average surface temperature (K)
RadT:	0	K	-	1	0	0	1	149	10	0	# Surface radiative temperature (K)
Albedo:	0	-	-	0	0	0	1	84	100	0	# Surface albedo (-)
SWE:	0	kg/m2	-	0	0	0	1	65	1000	0	# Snow Water Equivalent (kg/m2)
SWEVeg:	0	kg/m2	-	1	0	0	1	139	1000	0	# SWE intercepted by vegetation (kg/m2)
SurfStor:	0	kg/m2	-	1	0	0	1	150	1000	0	# Surface water storage (kg/m2)

### #Subsurface state variables

SoilMoist: 1 kg/m2 - 0 0 0 4 86 1000 0 # Average layer soil moisture (kg/m2)  
SoilTemp: 1 K - 0 0 0 4 85 1000 0 # Average layer soil temperature (K)  
SmLiqFrac: 0 - - 0 0 0 4 160 100 0 # Average layer fraction of liquid moisture (-)  
SmFrozFrac: 0 - - 0 0 0 4 140 100 0 # Average layer fraction of frozen moisture (-)  
SoilWet: 0 - - 0 0 0 1 144 100 0 # Total soil wetness (-)  
RelSMC: 0 m3/m3 - 0 0 0 4 141 1000 0 # Relative soil moisture  
RootTemp: 0 K - 0 0 0 1 142 1000 0 # Rootzone temperature (K)

### #Evaporation components

PotEvap: 0 kg/m2s UP 1 0 0 1 145 1 0 # Potential evapotranspiration (kg/m2s)  
ECanop: 0 kg/m2s UP 1 0 0 1 200 1 0 # Interception evaporation (kg/m2s)  
TVeg: 0 kg/m2s UP 1 0 0 1 210 1 0 # Vegetation transpiration (kg/m2s)  
ESoil: 0 kg/m2s UP 1 0 0 1 199 1 0 # Bare soil evaporation (kg/m2s)  
EWater: 0 kg/m2s UP 1 0 0 1 197 1 0 # Open water evaporation (kg/m2s)  
RootMoist: 0 kg/m2 - 0 0 0 1 171 1 0 # Root zone soil moisture (kg/m2)  
CanopInt: 0 kg/m2 - 0 0 0 1 223 1000 0 # Total canopy water storage (kg/m2)  
EvapSnow: 0 kg/m2s - 1 0 0 1 173 1000 0 # Snow evaporation (kg/m2s)  
SubSnow: 0 kg/m2s - 1 0 0 1 198 1000 0 # Snow sublimation (kg/m2s)  
SubSurf: 0 kg/m2s - 1 0 0 1 143 1000 0 # Sublimation of the snow free area (kg/m2s)  
ACond: 0 m/s - 1 0 0 1 179 100000 0 # Aerodynamic conductance  
CCond: 0 m/s - 1 0 0 1 181 100000 0 # Canopy conductance

### #Other hydrologic variables

WaterTableD: 0 m - 0 0 0 1 174 1 0 # Water table depth (m)  
TWS: 0 mm - 0 0 0 1 175 1 0 # Terrestrial water storage (mm)  
GWS: 0 mm - 1 0 0 1 176 1 0 # Ground Water Storage (mm)  
WT: 0 mm - 1 0 0 1 177 1 0 # Noah-MP WT variable (mm)

### #Cold season processes

Snowcover: 0 - - 0 0 0 1 238 100 0 # Snow covered fraction (-)  
SAlbedo: 0 - - 0 0 0 1 184 1000 0 # Albedo of the snow-covered area (-)  
SnowTProf: 0 K - 0 0 0 1 239 1000 0 # Temperature of the snow pack (K)  
SnowDepth: 0 m - 0 0 0 1 66 1000 0 # Snow depth (m)  
SLiqFrac: 0 - - 0 0 0 1 185 1000 0 # Fraction of SWE in the liquid phase

### #Variables to compared against remote sensed data

LWup: 0 W/m2 UP 1 0 0 1 212 1 0 # Longwave radiation up from the surface (W/m2)

### #Carbon variables

GPP: 0 kg/m2s2 DN 1 0 0 1 256 1 0 # Gross Primary Production

```

NPP:      0 kg/m2s2 DN  1 0 0 1 256 1  0 # Net Primary Production
NEE:      0 kg/m2s2 UP  1 0 0 1 256 1  0 # Net Ecosystem Exchange
AutoResp: 0 kg/m2s2 UP  1 0 0 1 256 1  0 # Autotrophic respiration
HeteroResp: 0 kg/m2s2 UP  1 0 0 1 256 1  0 # Heterotrophic respiration
LeafResp: 0 kg/m2s2 UP  1 0 0 1 256 1  0 # Leaf respiration
TotSoilCarb: 0 kg/m2 - 1 0 0 1 256 1  0 # Total soil carbon
TotLivBiom: 0 kg/m2 - 1 0 0 1 256 1  0 # Total living biomass

#Forcings
Wind_f:   1 m/s - 1 0 0 1 32 10  0 # Near surface wind (m/s)
Rainf_f:  0 kg/m2s DN  1 0 0 1 162 1000  0 # Average rainfall rate
Snowf_f:  0 kg/m2s DN  1 0 0 1 161 1000  0 # Average snowfall rate
Tair_f:   1 K - 1 0 0 1 11 10  0 # Near surface air temperature
Qair_f:   1 kg/kg - 1 0 0 1 51 1000  0 # Near surface specific humidity
Psurf_f:  1 Pa - 1 0 0 1 1 10  0 # Surface pressure
SWdown_f: 1 W/m2 DN  1 0 0 1 204 10  0 # Surface incident shortwave
radiation
LWdown_f: 1 W/m2 DN  1 0 0 1 205 10  0 # Surface incident longwave
radiation
PARDR_f:  0 W/m2 DN  1 0 0 1 256 10  0 # Surface incident PAR direct
PARDF_f:  0 W/m2 DN  1 0 0 1 256 10  0 # Surface incident PAR diffuse

#Additional forcings
DirectSW_f: 0 W/m2 - 1 0 0 1 166 10  0 # Surface direct incident shortwave
radiation
DiffuseSW_f: 0 W/m2 - 1 0 0 1 167 10  0 # Surface diffuse incident
shortwave radiation
NWind_f:  0 m/s N  1 0 0 1 34 10  0 # Northward wind
EWind_f:  0 m/s E  1 0 0 1 33 10  0 # Eastward wind
FHeight_f: 0 m - 1 0 0 1 256 10  0 # Height of forcing variables
CH_f:     0 - - 1 0 0 1 208 10  0 # Surface exchange coefficient for
heat
CM_f:     0 - - 1 0 0 1 252 10  0 # Surface exchange coefficient for
momentum
Emiss_f:  0 - - 1 0 0 1 256 10  0 # Surface emissivity
MixRatio_f: 0 kg/kg - 1 0 0 1 53 10  0 # Surface mixing ratio
CosZenith_f: 0 - - 1 0 0 1 256 10  0 # Cosine of zenith angle
Albedo_f:  0 - - 1 0 0 1 84 10  0 # Surface albedo

#Parameters
Landmask: 0 - - 0 0 0 1 81 1  0 # Land mask (0 - Water, 1 - Land)
Landcover: 0 - - 0 0 0 1 225 1  0 # Land cover
Soiltype:  0 - - 0 0 0 1 224 1  0 # Soil type
SandFrac:  0 - - 0 0 0 1 256 1  0 # Sand fraction
ClayFrac:  0 - - 0 0 0 1 256 1  0 # Clay fraction
SiltFrac:  0 - - 0 0 0 1 256 1  0 # Silt fraction
Porosity:  0 - - 0 0 0 1 240 1  0 # Porosity
Soilcolor: 0 - - 0 0 0 1 256 1  0 # Soil color
Elevation: 0 m - 0 0 0 1 196 10  0 # Elevation
Slope:     0 - - 0 0 0 1 222 10  0 # Slope
LAI:       0 - - 0 0 0 1 182 100  0 # LAI

```

```

SAI:          0 - - 0 0 0 1 256 100 0 # SAI
Snfralbedo:  0 - - 0 0 0 1 184 100 0 # Snow fraction albedo
Mxsnalbedo:  0 - - 0 0 0 1 159 100 0 # Maximum snow albedo
Greenness:   0 - - 0 0 0 1 87 100 0 # Greenness
Roughness:   0 m - 1 0 0 1 83 10 0 # Roughness
Tempbot:     0 - - 0 0 0 1 256 10 0 # Bottom soil temperature

#Routing
Streamflow:  0 m3/s - 0 0 0 1 256 10 0 # Streamflow

#Noah-MP
LeafMass:    0 g/m2 - 1 0 0 1 148 10 # leaf mass
RootMass:    0 g/m2 - 1 0 0 1 148 10 # stem mass
StemMass:    0 g/m2 - 1 0 0 1 148 10 # wood mass
WoodMass:    0 g/m2 - 1 0 0 1 148 10 # mass of wood including woody
roots [g/m2]
DeepSoilCarbon: 0 g/m2 - 1 0 0 1 148 10 # stable carbon in deep soil
[g/m2]
ShallowSoilCarbon: 0 g/m2 - 1 0 0 1 148 10 # short-lived carbon in
shallow soil [g/m2]

#LVT combination variables
EBAL:        0 - - 1 0 0 1 256 10 0 # Energy balance
WBAL:        0 - - 1 0 0 1 256 10 0 # Water balance
EVAPBAL:     0 - - 1 0 0 1 256 10 0 # Evaporation balance
SWE/P:       0 - - 1 0 0 1 256 10 0 # SWE over precipitation
ET/P:        0 - - 1 0 0 1 256 10 0 # Evapotranspiration over
precipitation
Qs/P:        0 - - 1 0 0 1 256 10 0 # Surface runoff over precipitation
Qsb/P:       0 - - 1 0 0 1 256 10 0 # Subsurface runoff over
precipitation

```



# Appendix A: How to verify a “non-LIS” dataset?

This section provides a description of how to convert a non-LIS data to a “LIS-style” so that verification and evaluation can be conducted using LVT.

First, an observation plugin for the dataset of interest must be developed within LVT. This plugin will handle the reading, processing and any spatial interpolation of the data and will connect the processed variables to the LVT core using the `LVT_logSingleVar` interface.

In the `lvt.config` file, specify the runmode to be “Observation processing”.

```
LIS Running mode: "Observation processing"
```

Though not used, the files `METRICS.TBL` and `TS_LOCATIONS.TXT` must be provided as a placeholder. The sample files provided along with the source code (under `src/configs`) can be used.

Finally, specify the `MODEL_OUTPUT_LIST_LVT.TBL` file such that the selection option for all the variables that should appear in the reprocessed files is enabled (Note that this is the second column in the file `MODEL_OUTPUT_LIST_LVT.TBL`. The last column which specifies “Use in LVT” option is ignored in this running mode.

The processed files in the “LIS-style” will be generated in the location specified by the following option:

```
Stats output directory: ./OUTPUT
```

# Appendix B: Description of output files from LVT

This section provides a description of various output files generated during an LVT analysis.

For the purposes of illustration, consider the following parameters for an LVT analysis

- Variables :  $Q_{le}$ ,  $Q_h$
- Metrics : *MEAN* and *RMSE*
- LSM : *Noah 3.2*
- location (from *TS\_LOCATIONS.TXT*) : *E20*
- Experiment name : *RUN*

## B.1. METADATA files

If the LVT output format is specified as binary, then a number of METADATA files will be output. The METADATA files contain the spatial domain, grid and map projection specifications and the list of variables and the order in which they appear. For the above example, a file named *MEAN\_NOAH32\_E20RUN\_METADATA.dat* will be created with entries such as the following:

```
DIMENSIONS
east-west          499
north-south        499

Missing value      -9999.000

GRID INFORMATION
MAP_PROJECTION: LAMBERT CONFORMAL
SOUTH_WEST_CORNER_LAT  34.42922
SOUTH_WEST_CORNER_LON -100.6136
TRUELAT1  36.70000
TRUELAT2  36.70000
STANDARD_LON -97.90000
DX  1.000000
DY  1.000000
VARIABLE: Qle          1
VARIABLE: COUNT_Qle          1
VARIABLE: OBS_Qle          1
VARIABLE: OBS_COUNT_Qle          1
VARIABLE: Qh          1
.....
.....
```

This file can be used to determine the order of variables written to the gridded output files.

For NETCDF output, the header of each file contains similar information.

## B.2. Stats summary file

The LVT analysis will write out a summary file, for each computed metric with the following name: *MEAN\_SUMMARY\_STATS.dat*. This file can be used to not only determine the domain averaged statistics, but also the order in which variables are written to files. For the above example the file *MEAN\_SUMMARY\_STATS.dat* will contain entries such as the following (The columns 1 to 4 represent the location name, average value for that location, confidence interval, number of points contributing to the average):

-----  
VAR: Q1e  
-----

ALL:	0.710E+02 +/-	0.492E+01	22
E1:	0.767E+02 +/-	-	1
E2:	0.826E+02 +/-	-	1
E3:	0.421E+02 +/-	-	1
E4:	0.659E+02 +/-	-	1

.....

.....

-----  
VAR: OBS\_Q1e  
-----

ALL:	0.906E+02 +/-	0.834E+02	11
E1:	-0.100E+05 +/-	-	0
E2:	0.670E+02 +/-	-	1
E3:	-0.100E+05 +/-	-	0
E4:	0.496E+02 +/-	-	1

.....

.....

-----  
VAR: Qh  
-----

ALL:	0.151E+02 +/-	0.564E+01	22
E1:	0.329E+02 +/-	-	1
E2:	-0.678E+01 +/-	-	1
E3:	0.164E+02 +/-	-	1
E4:	0.207E+02 +/-	-	1

.....

.....

## B.3. ASCII Time Series files

If the extract time series option is enabled for a particular metric in METRICS.TBL, then a corresponding time series file will be generated (similar files will be generated for each location in

the *TS\_LOCATIONS.TXT* file).

The time series files with the following names will be generated: *MEAN\_E4.dat*, *RMSE\_E4.dat*

The *MEAN\_E4.dat* file will have entries such as the following:

```
2007 05 02 01 00 qle qle_STD qle_min qle_max qle_ensSTD qle_CI obs_qle obs_qle_STD
obs_qle_min obs_qle_max obs_qle_ensSTD obs_qle_CI qh qh_STD qh_min qh_max qh_ensSTD
qh_CI obs_qh obs_qh_STD obs_qh_min obs_qh_max obs_qh_ensSTD obs_qh_CI

2007 05 02 02 00 qle qle_STD qle_min qle_max qle_ensSTD qle_CI obs_qle obs_qle_STD
obs_qle_min obs_qle_max obs_qle_ensSTD obs_qle_CI qh qh_STD qh_min qh_max qh_ensSTD
qh_CI obs_qh obs_qh_STD obs_qh_min obs_qh_max obs_qh_ensSTD obs_qh_CI

.....
.....
```

The columns represent Time (columns 1-5: year, month, day, hour, minute), mean value, spatial standard deviation, minimum, maximum, ensemble standard deviation, confidence interval of Qle from model (columns 6-11), and mean value, spatial standard deviation, minimum, maximum, ensemble standard deviation, confidence interval of Qle from observations (columns 12-17), and mean value, spatial standard deviation, minimum, maximum, ensemble standard deviation, confidence interval of Qh from model (columns 18-23), and mean value, spatial standard deviation, minimum, maximum, ensemble standard deviation, confidence interval of Qh from observations (columns 24-29).

If more variables are included in the analysis, then additional columns will be included for each variable (6 columns per variable).

For metrics such as RMSE (e.g. *RMSE\_E4.dat*, the file entries will be as follows (note that there are no columns for observation values):

```
2007 05 02 01 00 rmse_qle rmse_qle_STD rmse_qle_min rmse_qle_max
rmse_qle_ensSTD rmse_qle_CI
2007 05 02 02 00 rmse_qle rmse_qle_STD rmse_qle_min rmse_qle_max
rmse_qle_ensSTD rmse_qle_CI

.....
.....
```

## B.4. Domain time Series files (Binary/NETCDF)

LVT will output a gridded/tiled output file for each chosen metric, at each stats output interval, if the “write time series” option is enabled (in *METRICS.TBL*) for that metric. For the above example, files such as the following will be generated.

```
MEAN_TS.200705020000.d01.nc
MEAN_TS.200705030000.d01.nc
MEAN_TS.200705040000.d01.nc
```

*RMSE\_TS.200705020000.d01.nc*  
*RMSE\_TS.200705030000.d01.nc*  
*RMSE\_TS.200705040000.d01.nc*

where the timestamp indicates the end time of each analysis interval. For binary output file extension of “.gs4r” will be used instead of “.nc”.

## **B.5. Final domain files (Binary/NETCDF)**

LVT will output a gridded/tiled output file for each chosen metric. For the above example, two final files will be generated with filenames of *MEAN\_FINAL.200705100000.d01.nc* and *RMSE\_FINAL.200705100000.d01.nc*, where the timestamp indicates the end time of the LVT analysis. For binary output file extension of “.gs4r” will be used instead of “.nc”.

# References

- [kumar\_etal\_gmd\_2012] S.V. Kumar, C.D. Peters-Lidard, J.A. Santanello, K. Harrison, Y. Liu, and M. Shaw. Land surface Verification Toolkit (LVT) - a generalized framework for land surface model evaluation. *Geosci. Model Dev.*, pages 869–886, 2012.
- [protex] W. Sawyer and A. da Silva. ProTeX: A sample Fortran 90 source code documentation system. Technical report, NASA GMAO, 1997. DAO Office Note 97-11.