

# LVT tutorial

# Outline



Introduction



Obtaining LVT through repository

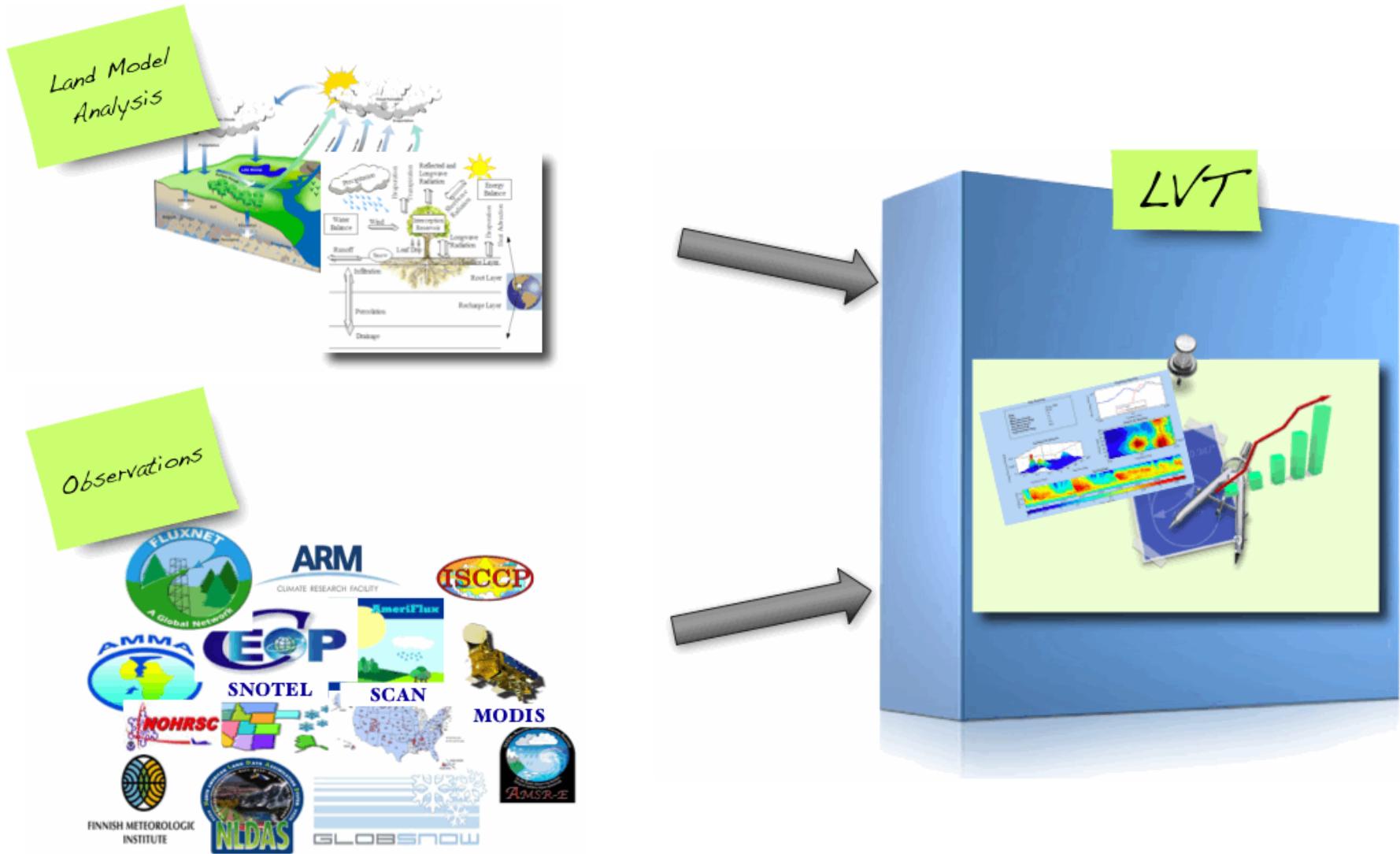


Building LVT



LVT – how to configure, run the verification system

# What is LVT?



# Features/Capabilities

- A range of evaluation metrics
- Land model diagnostics
- Spatial scale analysis
- Support for non-LIS data
- Extensible interfaces for incorporating new observational datasets, analysis metrics
- Processes datasets in the “native” formats (LVT handles the spatial and temporal transformations internally)

Metric class	Examples
Accuracy metrics	RMSE, Bias, correlation, ..
Ensemble metrics	Uncertainty importance
Information theory metrics	Entropy, Complexity
Data assimilation metrics	Mean, variance, of innovation distribution
Spatial similarity metrics	Hausdorff Norm
Scale decomposition metrics	Discrete wavelet transforms

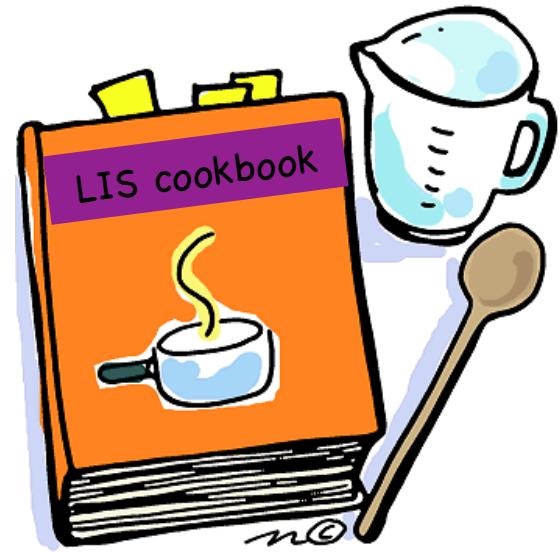
# Observational datasets supported in LVT

Model/Reanalysis products	
AGRMET (AFWA)	Water/energy fluxes, soil moisture, soil temperatures, snow, meteorology
NLDAS (NASA, NOAA)	Water/energy fluxes, soil moisture, soil temperatures, snow, meteorology
GLDAS (NASA)	Water/energy fluxes, soil moisture, soil temperatures, snow, meteorology
Canadian meteorological center's snow depth analysis	Snow depth
Snow data assimilation system (NOHRSC)	Snow depth, SWE
Satellite/remote sensing data	
ANSA	Snow depth, SWE
GlobSnow	SWE
ISCCP	Land surface temperature
MODIS	Snowcover, ET,
AMSR-E	Soil moisture, SWE

In-situ measurements	
AMMA	Water/energy fluxes, soil moisture, soil temperatures
ARM	Water/energy fluxes, soil moisture, soil temperatures, meteorology
Ameriflux	Water and energy fluxes
CEOP	Water/energy fluxes, soil moisture, soil temperatures, meteorology
COOP	Snow depth, precip, Land surface temperature
CPC unified	Precip
FLUXNET	Water and energy fluxes
FMI/SYKE	SWE
GSOD	Snow depth
ISMN	Soil moisture
SCAN	Soil moisture
WMO synoptic obs	Snow depth
NRCS SNOTEL	SWE
SURFRAD	Shortwave, longwave radiation
SWRC	Soil moisture
USGS water data	Streamflow

# Software Requirements

- Fortran 90/95 compiler (g95 will not work for LIS5.0)
  - preferred : intel, pgi, lahey, absoft
- C compiler
- Earth System Modeling Framework (ESMF)
  - 5.2.or
  - LVT supports GribI, NETCDF, HDF formats
    - GribI – mandatory
      - (grib\_api: [http://www.ecmwf.int/products/data/software/grib\\_api.html](http://www.ecmwf.int/products/data/software/grib_api.html))
    - NETCDF, HDF optional



# LVT Documentation

- User's guide

- Step-by-step instructions on how to build the LIS code

- Reference manual

- coming soon..

Land surface Verification Toolkit (LVT)  
User's Guide

January 12, 2012

Revision 1.0

History:

Revision	Summary of Changes	Date
1.0	Initial version for LIS 6.0	August, 2009



National Aeronautics and Space Administration  
Goddard Space Flight Center  
Greenbelt, Maryland 20771

# Getting LVT source

- 📌 Use the subversion repository (  
<https://flood.gsfc.nasa.gov>)
- 📌 Apply for an account  
([James.V.Geiger@nasa.gov](mailto:James.V.Geiger@nasa.gov))
- 📌 Check out the code (svn co  
<https://flood.gsfc.nasa.gov/svn/tools/lvt>  
LVT\_src)

# Compiling LVT

- ➊ Set the following environment variables
  - ➋ LVT\_SRC: location of the source
  - ➋ LVT\_ARCH (e.g. linux\_ifc)
  - ➋ LVT\_FC (e.g. mpif90)
  - ➋ LVT\_CC (e.g. mpicc)
  - ➋ LVT\_GRIBAPI (path to the grib api library)
  - ➋ LVT\_MODESMF (path to ESMF header files)
  - ➋ LVT\_LIBESMF (path to ESMF library files)
  - ➋ LVT\_NETCDF (path to the NETCDF library)
  - ➋ LVT\_HDF4 (path to the HDF4 library)
  - ➋ LVT\_HDF5 (path to the HDF5 library)
  - ➋ LVT\_HDFEOS (path to the HDFEOS library)

# Compiling LVT

- Run the configure script

```
% ./configure
-----
Setting up configuration for LVT version 1.0...
Optimization level (-1=g, 0,1,2,3): 0
Use NETCDF? (1=yes, 0=no): 1
Use HDF4? (1=yes, 0=no): 0
Use HDF5? (1=yes, 0=no): 0
Use HDFEOS? (1=yes, 0=no): 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.
-----
```

- Run the compile script

`./compile`