U.S. DEPARTMENT OF ENERGY | Office of Science

DOE/SC-ARM-TR-199

Description of the LASSO Alpha 2 Release

WI Gustafson
X Cheng
B Krishna
T Toto
AM Vogelmann
S Endo
Z Li
H Xiao

Revision 1
June 2018

ARM

DISCLAIMER

DOE/SC-ARM-TR-199

Description of the LASSO Alpha 2 Release

WI Gustafson Jr., Pacific Northwest National Laboratory
AM Vogelmann, Brookhaven National Laboratory
Principal Investigators

X Cheng, National University of Defense Technology, China
S Endo, Brookhaven National Laboratory
B Krishna, Oak Ridge National Laboratory
Z Li, University of California, Los Angeles, and Jet Propulsion Laboratory
T Toto, Brookhaven National Laboratory
H Xiao, Pacific Northwest National Laboratory
Co-Investigators

Revision 1
June 2018

LASSO web page: https/www.arm.gov/capabilities/modeling/lasso/
Alpha 2 documentation doi:10.2172/1376727
Alpha 2 data set doi:10.5439/1342961

Work supported by the U.S. Department of Energy,
Office of Science, Office of Biological and Environmental Research

Acknowledgements

The LASSO Pilot Project team consists of William I. Gustafson Jr. (Principal Investigator), Andrew M. Vogelmann (Co-Principal Investigator), Xiaoping Cheng, Satoshi Endo, Zhijin Li, Tami Toto, and Heng Xiao. The LASSO Bundle Browser has been developed by Bhargavi Krishna. Funding provided by the U.S. Department of Energy Office of Science Biological and Environmental Research via the Atmospheric Radiation Measurement user facility. We acknowledge the advice from external members for the LASSO Advisory Team: Maike Ahlgrimm (ECMWF), Chris Bretherton (University of Washington), Graham Feingold (NOAA ESRL), Chris Golaz (LLNL), David Turner (NOAA NSSL), Minghua Zhang (Stony Brook University), and Jim Mather (ARM Technical Director). We acknowledge numerous members of the ARM infrastructure team for their coordination, installation, maintenance, processing, and advice given on the data products used and for LASSO product management. Portions of the work were performed at 1) Pacific Northwest National Laboratory (PNNL)—Battelle Memorial Institute operates PNNL under contract DEAC05-76RL01830, 2) Brookhaven National Laboratory, and 3) the Jet Propulsion Laboratory and University of California, Los Angeles, via a subcontract through PNNL. Computation has been provided by 1) the ARM Data Center Computing Facility, 2) the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725; 3) the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported under Contract No. DE-AC02-05CH11231; and 4) PNNL Institutional Computing.

Acronyms and Abbreviations

ADC	ARM Data Center
AERI	atmospheric emitted radiance interferometer
AERIoe	optimally interpolated AERI
ARM	Atmospheric Radiation Measurement
ARSCL	active remote sensing of clouds
BNL	Brookhaven National Laboratory
CF	cloud fraction
Co-PI	co-principal investigator
DDH	Diagnostics in the Horizontal Domains
DOE	Department of Energy
DOI	digital object identifier
ECMWF	European Centre for Medium Range Weather Forecasting
EOSDIS	Earth Observing System Data and Information System
ESRL	Earth System Research Laboratory
ETS	equitable threat score
FASTER	FAst-physics System TEstbed & Research
FTP	file transfer protocol
GOES	Geostationary Operational Environmental Satellite
ID	identification number
IFS	Integrated Forecast System
LASSO	LES ARM Symbiotic Simulation and Observation
LCL	lifting condensation level
LES	large-eddy simulation
LFC	level of free connection
LLNL	Lawrence Livermore National Laboratory
LWP	liquid water path
MET	surface-meteorology instrument clusters
MSDA	Multiscale Data Assimilation
MWR	microwave radiometer
MWRRet	Microwave Radiometer Retrieval
NASA	National Aeronautics and Space Administration
NOAA	National Oceanographic and Atmospheric Administration
NSSL	National Severe Storms Laboratory
NWS	National Weather Service

OSSE	observation system simulation experiment
PBL	planet boundary layer
PI	principal investigator
PNNL	Pacific Northwest National Laboratory
QC	quality control
RMS	root-mean square
RRTMG	Rapid Radiation Transfer Model for Global Climate Models
SAM	System for Atmospheric Modeling
SGP	Southern Great Plains
SONDE	radiosonde
THREDDS	Thematic Real-time Environmental Distributed Data Services
TKE	turbulent kinetic energy
TSI	total sky imager
UTC	universal time code
VARANAL	variational analysis
WRF	Weather Research and Forecasting Model

Contents

Figures

Table

1.0 Introduction

1.1 The LASSO Project

The U. S. Department of Energy (DOE) Atmospheric Radiation Measurement (ARM) user facility began a pilot project in May 2015 to design a routine, high-resolution modeling capability to complement ARM's extensive suite of measurements. This modeling capability has been named the Large-Eddy Simulation (LES) ARM Symbiotic Simulation and Observation (LASSO) project. The initial focus of LASSO is on shallow convection at the ARM Southern Great Plains (SGP) atmospheric observatory.

The availability of LES simulations with concurrent observations will serve many purposes. LES helps bridge the scale gap between DOE ARM observations and models, and the use of routine LES adds value to observations. It provides a self-consistent representation of the atmosphere and a dynamical context for the observations. Further, it elucidates unobservable processes and properties. LASSO will generate a simulation library for researchers that enables statistical approaches beyond a single-case mentality. It will also provide tools necessary for modelers to reproduce the LES and conduct their own sensitivity experiments.

Many different uses are envisioned for the combined LASSO and observational library. For an observationalist, LASSO can help inform instrument remote-sensing retrievals, conduct observation system simulation experiments (OSSEs), and test implications of radar scan strategies or flight paths. For a theoretician, LASSO will help calculate estimates of fluxes and co-variability of values, and test relationships without having to run the model personally. For a modeler, LASSO will help one know ahead of time which days have good forcing, have co-registered observations at high-resolution scales, and have simulation inputs and corresponding outputs to test parameterizations. Further details on the overall LASSO project are available at https://www.arm.gov/capabilities/modeling/lasso.

1.2 LASSO Alpha Releases

This Alpha 2 release is the second and final early data release for the LASSO pilot phase. The Alpha 1 release occurred in July 2016 and was the first dissemination of simulations and analysis tools from LASSO, focusing on five shallow convection cases from spring-summer 2015 at the SGP site. More information on the Alpha 1 release is available at https://www.arm.gov/capabilities/modeling/lasso/releases/alpha1 and via the Description of the LASSO Alpha 1 Release (Gustafson et al. 2016). The Alpha 2 release, documented in this report and originally released in September 2017, focuses on 13 shallow convection cases during May to August 2016 at the SGP site. These 13 cases from 2016 have been expanded to include the 2015 case dates from Alpha 1 via a supplemental release to Alpha 2 in April 2018, described in Appendix D. Both Alpha releases are meant for evaluation purposes. The complete set of LASSO data products and tools can be accessed via https://www.arm.gov/capabilities/modeling/lasso/releases. This web page serves as a high-level interface to the data files, collectively called data bundles, which consist of LES input and outputs, ARM observations co-registered on the model grid, model diagnostics and skill scores, and quicklooks of various fields. An interface is provided through a web browsing tool, called the LASSO Bundle Browser, for users to find simulations of interest through examination of the LES performance relative to select ARM observations.

Alpha 2 expands upon Alpha 1 through the use of additional ARM observations when deriving model forcing data and when evaluating the LES results. Improvements have also been made in the LES to increase accuracy. A list of changes in Alpha 2 versus Alpha 1 is provided in Appendix A.

The Alpha 2 release contains a library of 544 simulations for 2016, plus 140 simulations for the 2015 supplement, and model-observation comparisons for different LES models, large-scale forcings, model configurations, and preliminary observational products that the LASSO team is using to assess workflow options for use in operations. Simulations from both the System for Atmospheric Modeling (SAM) and the Weather Research and Forecasting (WRF) models are available. They are combined with a range of forcings under consideration for use in LASSO combined with the sensitivity of model simulations and microphysics options. Many of the simulations are directly comparable, e.g., they use the same model configuration and only differ in the forcing or initial conditions used to drive the model. Other simulations differ in the particular choice of microphysics, domain size, or grid spacing. The release also serves as an initial introduction to the skill scores being designed for ranking the quality of the simulations. An initial Bundle Browser interface has been designed to help users find simulations of relevance for their needs, which is a tool based on the Cassandra NoSQL database methodology.

The simulations within this release represent the typical behavior to be expected from LES using best-practice configurations and are valid for use in various research applications. However, as made evident by the comparisons to observations, there is a range of simulation behavior so we encourage users to contact the LASSO team to ensure the details of the simulations are understood and that they are used appropriately for their applications. Note that some fields in the Alpha 1 model output have been deemed questionable and should not be used.

With the release of Alpha 2, Alpha 1 data bundles have been removed from the Bundle Browser, but the data are still accessible via the ARM principal investigator (PI) data area. See https://www.arm.gov/capabilities/modeling/lasso/releases/alpha1 for links to the prior version's documentation and data. A subset of the Alpha 1 simulations have been re-run using the code from Alpha 2 to fix bugs and enable hosting of the 2015 case dates within the Bundle Browser. Specifics on these re-run cases are described in Appendix D.

A key objective for the Alpha 2 release is to solicit feedback from the community. Efforts to improve the model configuration, forcings, and analysis tools are ongoing and will be based on user feedback and continued effort. We encourage users to explore the available simulations and tools and to share their experience and ideas for improvement with the LASSO team (lasso@arm.gov).

1.3 Highlighted Links

The Alpha 2 release consists of several components that are intended to be viewed as a single entity. The documentation is referenced as doi:10.2172/1376727, the data bundles are referenced as doi:10.5439/1342961, and the Bundle Browser is the primary web interface for accessing the data bundles. Links to quicklook plots and ordering of data bundles can be accessed from the Bundle Browser. Links to the different components are listed below.

Overview web page: https://www.arm.gov/capabilities/modeling/lasso

Alpha 2 web page: https://www.arm.gov/capabilities/modeling/lasso/releases/alpha2

Documentation: https://www.arm.gov/publications/tech_reports/doe-sc-arm-tr-199.pdf

Bundle Browser: https://adc.arm.gov/lassobrowser

1.4 Proper Acknowledgement when Using the Alpha 2 Release

At the time of the release, the Alpha 2 simulations and tools are not yet documented in a peer-reviewed article. We therefore request that users publishing with the LASSO products consider including appropriate LASSO team members as co-authors. Contact the LASSO PI (william.gustafson@pnnl.gov) for specifics regarding the different team member contributions.

The LASSO Alpha 2 documentation and data set have separate DOIs; therefore, they should be cited separately. The recommended citation for the documentation is:

> Gustafson, WI, AM Vogelmann, X Cheng, S Endo, B Krishna, Z Li, T Toto, and H Xiao, 2018: Description of the LASSO Alpha 2 Release. Ed. by R. Stafford, DOE Atmospheric Radiation Measurement user facility, DOE/SC-ARM-TR-199, doi:10.2172/1376727.

And, the recommended citation for the Alpha 2 data is:

> Atmospheric Radiation Measurement (ARM) user facility, 2017: LASSO Alpha 2 Data Bundles. <<dates of the data used, e.g., 10 Jun 2016 or various dates>>, 36° 36′ 18.0″ N, 97° 29′ 6.0″ W: Southern Great Plains Central Facility (C1). Compiled by WI Gustafson, AM Vogelmann, X Cheng, S Endo, B Krishna, Z Li, T Toto, and H Xiao. ARM Data Center: Oak Ridge, Tennessee, USA. Data set accessed <<insert date downloaded from web>> at http://dx.doi.org/10.5439/1342961.

At a minimum, ARM requests the following acknowledgement:

> LASSO data were used from the U.S. Department of Energy's Atmospheric Radiation Measurement (ARM) user facility.

2.0 Modeling Details

2.1 LES Model Configuration

Two models are being evaluated for ongoing use in LASSO. One is the SAM model (Khairoutdinov and Randall 2003) version 6.10.8 and the other is the WRF model version 3.8.1 (Skamarock et al. 2008) with additional components developed for the DOE FAst-physics System TEstbed & Research (FASTER) project (Endo et al. 2015). The two models have been configured as similarly as possible to enable comparing pairs of simulations for a given set of forcing and domain configurations. The WRF-FASTER modification to WRF includes LES-specific output, such as domain-averaged profiles that are time averaged between output times based on a specified sampling frequency, making it similar to how SAM handles output. Small changes have been made to the SAM model to make its handling of initial conditions more consistent to WRF-FASTER. Both models are run with doubly periodic boundary conditions.

Physics between the two models are generally similar. Many simulations in this release use Morrison microphysics (Morrison et al. 2005, 2009), as this is available in both SAM and WRF. An additional set of simulations have been done with WRF using the Thompson scheme (Thompson et al. 2004, 2008) to evaluate sensitivity to the microphysics choice. The resulting shallow clouds are similar between the two options; however, Morrison generates more cirrus than Thompson. The version of Morrison in WRF-FASTER is slightly older than in SAM, but the two behave similarly. The shortwave and longwave Rapid Radiation Transfer Model for Global Climate Models (RRTMG) radiation schemes are used for both models (Clough et al. 2005; Iacono et al. 2008; Mlawer et al. 1997). The subgrid-scale scheme for both models is based on the 1.5 order turbulent kinetic energy (TKE) approach (Deardorff 1980).

The domain configurations for the Alpha 2 release use 100-m grid spacing with the exception of a handful of sensitivity tests using finer and coarser resolutions. Most simulations have been run using a domain that is 14.4 km across, with some larger and smaller domains also available for comparison purposes. All simulations use 226 levels that extend from the surface to 14.7 km. Vertical grid spacing is 30 m up to 5 km and then stretches to 300 m near the model top. We recognize that, while these all-purpose configurations are sufficient for testing the model forcings and various scientific applications, higher resolutions or larger domains may be desirable for some applications. To assist those researchers, we make the input forcings, assessments of their quality, and model evaluation data available so that these simulations may serve as a starting point for their own simulations configured and tailored to their needs (see the Data Bundles Section).

Other details for the model setups can be found in the config directories associated with each model simulation. The model codes are mostly out of the box. Interested parties can contact the LASSO team (lasso@arm.gov) if they would like access to the specific code used for this release.

2.2 Large-scale Forcing, Surface Fluxes, and Initial Conditions

Initial profiles and surface forcings for the simulations come from multiple sources. Most simulations in the release use 12 UTC radiosonde soundings for the initial profiles and the observationally based surface

Table 1. Mapping of forcing-related variables between WRF and SAM input files.

	WRF File-name	WRF Variable Name	SAM File-name	SAM Variable Name	Variable Description
Surface Forcing	input_sfc_forcing.nc	Times	sfc	day	Timing info (date-time string; day of year)
		PRE_TSK		sst	Skin temperature (K)
		PRE_SH_FLX		H	Surface sensible heat flux (W m^{-2})
		PRE_LH_FLX		LE	Surface latent heat flux (W m^{-2})
				TAU	Surface momentum flux (m^2 s^{-2}) NOT PRESCRIBED
		PRE_ALBEDO			Surface albedo (unitless)
		PRE_TSK_TEND			Skin temperature tendency (K s^{-1})
		PRE_SH_FLX_TEND			Surface sensible heat flux tendency (W m^{-2} s^{-1})
		PRE_LH_FLX_TEND			Surface latent heat flux tendency (W m^{-2} s^{-1})
		PRE_ALBEDO_TEND			Surface albedo tendency (s^{-1})
Initial Sounding	wrfinput_d01	Times, PSFC	snd	day, pres0	Timing info (date-time string; day of year), surface pressure (WRF: Pa, SAM: hPa)
		(PH + PHB)/9.81		z	Height above ground level (m)
		P + PB		p	Pressure (WRF: Pa, SAM: hPa) NOT USED FOR SAM
		T + 300		tp	Potential temperature (K)
		QVAPOR		q	Water vapor mixing ratio (WRF: kg kg^{-1}, SAM: g kg^{-1})
		U, V		u, v	Horizontal wind components (m s^{-1})
Large-Scale Forcing	input_ls_forcing.nc	Times, no time-dependent pres.	lsf	day, pres0	Timing info (date-time string; day of year), surface pressure (hPa)
		Z_LS		z	Height above ground level (m)
		TH_ADV		tpls	Large-scale potential temperature tendency (K s^{-1}) (only potential temperature tendency due to large-scale horizontal advection is applied)
		QV_ADV		qls	Large-scale moisture tendency (WRF: kg kg^{-1} s^{-1}, SAM: g kg^{-1} s^{-1}) (only moisture tendency due to large-scale horizontal advection is applied)
		U_LS, V_LS		uls, vls	Large-scale horizontal winds to nudge toward (m s^{-1}) NOT USED
		W_LS		wls	Large-scale vertical velocity (m s^{-1}) (tendencies due to large-scale vertical advection are calculated dynamically at each grid point using this prescribed vertical velocity)
		Z_LS_TEND			Time tendencies of Z_LS (m s^{-1})
		TH_ADV_TEND			Time tendency of TH_ADV (K s^{-2})
		QV_ADV_TEND			Time tendency of QV_ADV (kg kg^{-1} s^{-2})
		U_LS_TEND, V_LS_TEND, W_LS_TEND			Time tendencies of wind components (m s^{-2})
		TH_RLX			Potential temperature profile for nudging (K) NOT USED
		QV_RLX			Water vapor mixing ratio profile for nudging (kg kg^{-1}) NOT USED
		TH_RLX_TEND			Time tendency of TH_RLX (K s^{-1}) NOT USED
		QV_RLX_TEND			Time tendency of QV_RLX (kg kg^{-1} s^{-1}) NOT USED

forcing from the ARM constrained variational analysis (VARANAL) product, discussed below. Others use surface forcing from the same data source as the large-scale forcing to identify sensitivity to the surface fluxes.

Each LASSO data bundle contains the forcing file used to generate the associated LES output. Since the Alpha release contains simulations from both the SAM and WRF models, the forcings are available to users in two different input formats depending on which data bundle is used to access the forcing data, either matching the input requirements for SAM or for the LASSO version of WRF. The forcing data consists of horizontal advective tendencies of potential temperature and moisture, large-scale vertical velocity, and relaxation profiles for potential temperature, water vapor, and wind components. The tendencies due to large-scale vertical advection are calculated dynamically at each grid point using the prescribed vertical velocity. Note that the relaxation profiles are provided, but nudging to these profiles is not done for LASSO. Surface fluxes for sensible and latent heat are also provided. Table 1 shows the different forcing-related variables and which files hold them within each data bundle. The WRF large-scale forcing files contain both the tendency of the large-scale state as well as the tendency of the tendency. The latter are used within WRF for accumulating the large-scale tendency that is applied to update the model state each time step.

Three different methodologies have been applied for deriving large-scale forcings to drive the LES models. The goal is to provide a vetted ensemble of LES runs for each case based on multiple forcings since the forcings are arguably one of the largest uncertainties for LES modeling. The three methodologies are also supplemented with different forcing region scales that may add additional model spread, as the best scale may vary from case to case. This is particularly important for days when large variations occur around SGP that get averaged into the overall forcing, which is represented as a single profile that varies on an hourly basis.

VARANAL

The first forcing method is the ARM constrained variational analysis, VARANAL (Xie et al. 2004), which is based on Zhang and Lin (1997) and Zhang et al. (2001). VARANAL for this release uses the National Oceanic and Atmospheric Administration (NOAA) Rapid Refresh (RAP) analyses as a background gridded field that is then optimally merged with ARM and other observations using a variational approach. The standard VARANAL represents conditions over a 300-km region. Packaged with VARANAL are observation-based estimates of surface sensible and latent heat fluxes over the forcing region, which are used to drive the bottom boundary conditions of the LES.

ECMWF

The second forcing method is a data set derived from European Centre for Medium-Range Weather Forecasts (ECMWF) forecasts. Two methods were tested. One is based on deriving forcing tendencies from the column nearest to SGP, which is the simplest approach that represents the single-column spatial extent of about 16 km. The simulations in this release use the second method in which large-scale forcing is based on the Diagnostics in the Horizontal Domains (DDH) system, which uses physical and dynamical tendencies directly from the ECMWF Integrated Forecast System (IFS) model to calculate closed budget terms. This method can more accurately close the moisture and energy budgets, plus it can be done over multiple spatial extents. The two methods produce similar results for the single-column scale. The Alpha 2 simulations use the DDH-based ECMWF forcing scale of 413 km, 114 km, and 16 km (a single IFS column). Note that the spatial scales listed here are average side dimensions based on the square root of the forcing domain area; the DDH domains are defined with longitude and latitude.

MSDA

The third forcing method is to derive the large-scale forcing from convection-permitting WRF simulations constrained using the multiscale data assimilation (MSDA) methodology developed by Zhijin Li (Li et al. 2015a, 2015b, 2016). MSDA is implemented using the community-based Gridpoint Statistical Interpolation (GSI) data assimilation software in conjunction with a scale separation algorithm to combine observations representing coarse and fine scales to accurately reflect the atmospheric state. It leverages the large-scale fields from existing reanalyses or forecasts produced by operational centers, but constrains small-scale fields by assimilating ARM observations, satellite measurements, and observations from other meteorological observing networks.

In the case of LASSO, the MSDA is carried out on a nested region over the central United States with the finest grid using 2-km grid spacing for the SGP region. The 2-km grid spacing offers the flexibility of generating large-scale forcing for selected area sizes. In the released cases, the areas for 75 km, 150 km, and 300 km are used for examining the sensitivity of large-scale forcing to the selected area sizes. In the 2015 cases from Alpha 1, the assimilated ARM observations include ARM radiosonde soundings from the Central Facility, and temperatures and moistures from the MET stations from across the site. Additional measurements included in the assimilation process include NOAA operational observations and satellite radiances. These measurements have been supplemented with additional hourly wind profiles for the 2016 cases in Alpha 2. Alpha 2 provides two versions of MSDA, one with the Alpha 1 set of input data and a second that adds wind profiles. The wind profiles come from the four SGP radar wind profilers. One is located at the Central Facility and the other three are spaced approximately 15 km away from the Central Facility. These profiles have been quality controlled to remove clutter effects and spurious values and are used within the MSDA.

3.0 Evaluation Data

Simulations are evaluated using ground-based ARM observations and retrievals of boundary-layer cloud and thermodynamic properties. This section describes the observations used thus far, and the next section describes the diagnostics and metrics. As discussed below, a meaningful comparison between observations and model output requires that they be co-registered on the same spatial and temporal grid and processed, as necessary, such that an apples-to-apples comparison may be made between a model-comparable observational quantity and an observation-comparable model quantity. Some of the methodology is based on Appendix B in Vogelmann et al. (2015). The Alpha 2 release provides 1-h averages for model-observation comparisons unless otherwise noted. Observations at native resolution are available by request, at this time.

In-Cloud Liquid Water Path (LWP)

Different from the Alpha 1 release, Alpha 2 in-cloud liquid water path (LWP) (g m^{-2}) is based on microwave radiometer (MWR)-only retrievals from the 2-channel MWR retrieval (MWRRet) (Turner et al. 2007) and retrievals from a new optimal estimation framework (AERIoe) (Turner and Löhnert 2014) that uses spectral infrared radiances measured by the atmospheric emitted radiance interferometer (AERI) and MWR radiances. The resulting LWP data have excellent sensitivity at low LWP ($1 < \mathrm{LWP} < 40$ g m^{-2}) and a full dynamic LWP range (up to 1000 g m^{-2}). Note that these retrievals have passed preliminary quality control (QC) but more rigorous QC is needed before they should be considered final. Additional work is ongoing to improve this product prior to releasing the operational LASSO product. Clear-sky screening is applied to the observations and simulations in a consistent manner so that only cloud values are used in the averaging (i.e., it is not a domain-averaged, all-sky value). Only retrievals from a single site at the SGP Central Facility are currently available, which will be expanded in the future. (See Appendix A for further details.)

1D Boundary-Layer Cloud Fraction (1D CF)

Two estimates of cloud fraction are used to approximate the measurement uncertainty of the shallow cloud fraction experienced at the surface. One is the hemispheric sky cover from the total sky imager (TSI), and the other is a column measurement of the cloud frequency of occurrence below 5 km derived from the ARM Active Remote Sensing of Clouds (ARSCL) value-added product (Clothiaux et al. 2000), which uses radar and lidar measurements to generate a vertically resolved cloud mask for the narrow column above the instruments. Cloud fraction is computed from the simulations in a manner following from the computation of in-cloud LWP (described in Appendix A). For the Alpha 2 release, the recommended one-dimensional (1D) CF measurement is from the TSI, as clouds above the boundary-layer clouds, e.g., cirrus, generally have a negligible impact on CF for the cases provided when boundary-layer cloud is present. (See Appendix A for further details.) CF from ARSCL will be improved in a future release by applying an ARSCL simulator to the simulations for a better apples-to-apples comparison.

2D Time-Height Cloud Mask

Two-dimensional (2D) cloud masks of the time-height location of cloud are used to assess the simulated cloud-base height, approximate vertical extent, and timing of cloud onset and decay. The observed cloud mask is generated from the ARSCL value-added data product, in which cloud fractional occurrence is determined for 15-minute windows and frequencies greater than zero are masked as being cloud. The lowest ARSCL cloud mask is at 160 m above the ground, so only clouds at or above this level are compared to the simulations. Simulated clouds below 160 m are not considered when comparing against ARSCL. The current cloud mask for simulations is obtained from grid cells with total hydrometeor mixing ratios greater than 10^{-7} kg kg^{-1}. The presence of liquid water cloud is relatively insensitive to this mixing ratio threshold value, but the presence of cirrus clouds is much more sensitive to the threshold.

The cloud masks available in the Alpha 2 release are useful for assessing the gross features of the simulated vertical distribution of cloud, but caution should be used to not over-interpret the results. This comparison assumes that the frozen turbulence assumption is valid, where the clouds sampled in the narrow column above the instruments is representative of the cloud field, which becomes more problematic for lower values of the 1D CF. Boundary-layer clouds may also be difficult to observe adequately when their radar reflectivity is low or when there is insect contamination (Lamer and Kollias 2015). A newly developed ARSCL simulator will be applied to simulations in the future.

Regional Lifting Condensation Level Height

The lifting condensation level height (LCL, m) is determined from continuous surface-air observations of relative humidity and temperature as the altitude where the surface-air moisture equals saturation following a dry-adiabatic ascent. Values are computed from surface meteorological observations and the same calculation is applied to the lowest air layer in the simulations, enabling a consistent observation-model comparison. In Alpha 1, LCL values were computed from the MET observations at the ARM Central Facility. In Alpha 2, this value is available (“LCL”) plus a domain-averaged LCL (“LCL_domian”) is computed from all ARM MET stations and the Mesonet stations within 60 km of the Central Facility. The regional variation of the latter is quantified as a standard deviation. The LCL heights of the individual values for the latter are offset according to site altitude to account for any height difference of the station relative to the SGP Central Facility.

Surface Temperature and Moisture

The SGP Central Facility MET station provides continuous measurements of surface air temperature, $T_{surface}$ (K), and moisture in the forms of mixing ratio, $QV_{surface}$ (g kg^{-1}), and relative humidity, $RH_{surface}$ (%), that are compared to values from the lowest model layer. In the future, other surface sites will be included in an areal average.

Boundary-Layer Thermodynamic Profiles

At sonde launch times, simulated thermodynamic profiles are compared to the lowest 5 km of the atmosphere observed by soundings and Raman lidar-based profiles of temperature and water vapor mixing ratio at the Central Facility. Raman lidar measurements provide high-frequency vertical profiles (~75 m vertical spacing every 10 minutes) of the boundary-layer water vapor mixing ratio, temperature, and relative humidity up to cloud base (see Appendix A for details). Water vapor profiles are provided from the Raman lidar above 0.1 km. Temperature profiles are a blend of the AERIoe profile retrievals and Raman lidar retrievals (see Appendix A). Future work will investigate inclusion of a continuous comparison of the boundary-layer structure.

Mid-Boundary-Layer Moisture and Temperature

The mid-boundary-layer moisture and temperature are an average over a 200-m-thick layer. The Raman lidar-based thermodynamic profiles described above are used for the averaging unless a sonde is available at that time, in which case rawindsonde data are used. In contrast to Alpha 1, where the layer is at a constant height for most case days, the values for Alpha 2 are from a layer whose height is set based on the observed cloud-base height for the case day. The top of the averaging layer is 200 m below the minimum observed cloud-base height between 12-24 UTC, meaning that the averaging layer is 200-400 m below the minimum cloud-base height. If the minimum cloud-base height is 450 m or lower, the top of the layer is set at 250 m. The values are averaged using a one-hour moving window to produce mid-boundary-layer-averaged water vapor mixing ratio, $QV_{boundary\_layer}$ (g kg^{-1}), temperature, $T_{boundary\_layer}$ (K), and relative humidity, $RH_{boundary\_layer}$ (%). These values are compared to the simulated values within the same height range.

Regional Boundary-Layer Cloud-Base Height

A regional-averaged boundary-layer cloud-based height is determined from five Doppler lidar estimates. Doppler lidars are located at the Central Facility and at four extended facilities surrounding the Central Facility about 45 km away. The Doppler lidar cloud-base heights are available at 10-minute intervals and are screened for clouds linked to boundary-layer fluxes through a comparison to the LCL at each site. Hourly averages are taken of values that pass the screening and a standard deviation characterizes the regional variability.

4.0 Diagnostics and Skill Scores

The evaluation data are used to assess model behavior using diagnostic plots and quantify model performance using skill scores. Diagnostics plots display comparable observed and simulated values that have been co-registered on the same spatial and temporal grid, and may be accompanied by basic statistical quantities (e.g., mean, root mean square [RMS], etc.). Skill scores are metrics that quantify the model performance seen in the diagnostic plots such that simulation quality can be numerically compared for different variables. Data files and quicklook plots of the diagnostics and skill scores are available within the data bundles, and a set of key skill scores are used in the LASSO Bundle Browser for users to find and select simulations based on their performance (see the LASSO Data Bundles and Tools Section).

4.1 Diagnostic Plots

Heat maps provide in a single plot an overview of model performance of a variable for all simulations within a case day. An example is given in Figure 1 where the magnitudes of the model-observation differences are provided as colored boxes for the diurnal cycle of each simulation. Heat map quicklooks (.png) are generated for: LCL, LCL_Domain; boundary-layer cloud-base height, CF(ARSCL); CF(TSI); LWP (linear y-axis); LWP (log y-axis); mid-boundary-layer Q_v, RH, T; and surface Q_v, RH, T. They are available for display via the "Heat Maps" button within the left column of the Bundle Browser and they are also stored within the data bundles, where they can be accessed using "heat_maps.html" located within the metrics directory of the *sgplassodiagconfobsmod#C1.m1.YYYYMMDD.tar* file of each data bundle, where YYYYMMDD is the case date and # is the simulation ID.

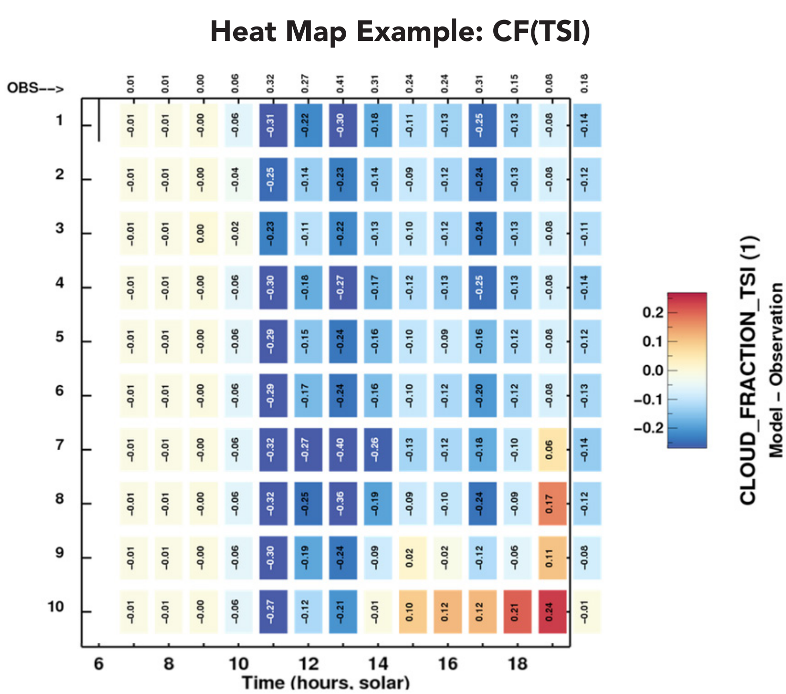


Figure 1. Heat map example of simulated cloud fraction compared to the TSI for the 27-Jun-2015 case. Simulation number is given by the left column and the local, solar time is given at the bottom. The top row gives the observational mean for each hour and the last value outside the box is the daily mean. The grid provides the corresponding differences of the simulated values (model-observation) that are color coded by the key given at the right.

Diagnostic Plot Examples

LASSO: 20150627, Simulation #10

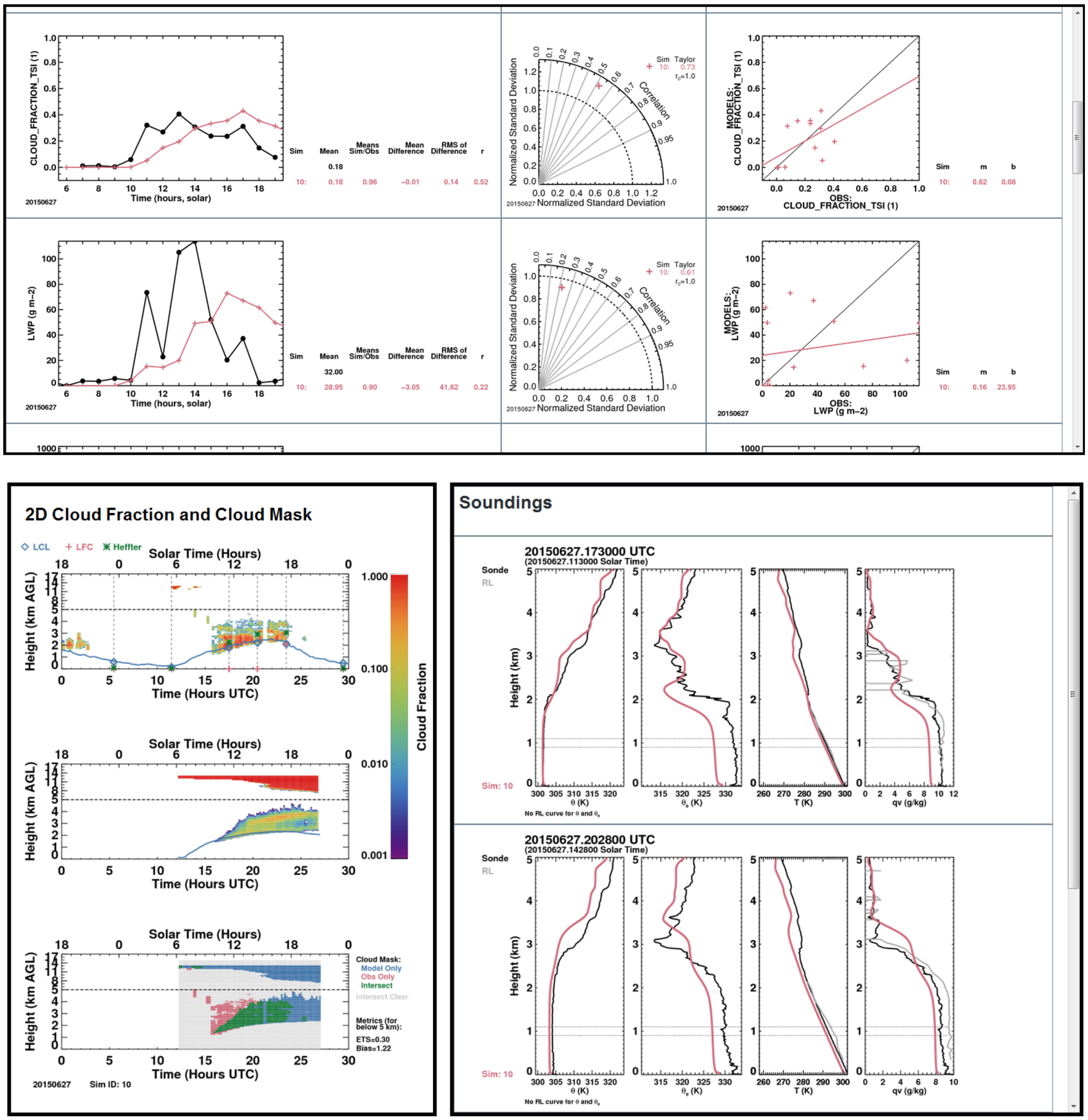


Figure 2. Example of diagnostic plots available for each simulation within the data bundles (see text for details). (Top panel) Quicklooks are available for various time series, Taylor diagram, and regression plots. (Bottom-left panel) Top plot is the 2D time-height cloud frequency from ARSCL, middle is from the simulation, and bottom is the 2D cloud masks from ARSCL and the simulation. (Bottom-right panel) Comparison of simulated profiles with those from sondes and the Raman lidar.

A series of other quicklook diagnostics plots are used to compare specific simulations to observations, which includes thermodynamic profiles at sounding times. Examples are given in Figure 2. They are available for display from within the Bundle Browser via the "Diagnostics" links in the browser's table. And, the diagnostic plots are stored in the *sgplassodiagconfobsmod#C1.m1.YYYYMMDD.tar* file for each simulation where they can be easily viewed using the "plots.html" file within the obs_model directory.

Top Panel

Contains plots of time series, Taylor diagrams, and regressions. Diagnostics plots are available for: LCL; LCL_Domain; boundary-layer cloud-base height; CF(ARSCL); CF(TSI); LWP (linear y-axis); LWP (log y-axis); mid-boundary-layer Q_v, RH, T; and surface Q_v, RH, T. The time series are accompanied by simple statistics: means of the observations and simulation, ratio of the means (Sim/Obs), mean difference (Sim-Obs), RMS difference, and correlation coefficient. Taylor diagrams graphically summarize how closely a pattern matches observations in terms of their correlation and normalized standard deviation (Taylor 2001). A perfect match of these terms in polar coordinates is at (1,1) (although, as discussed in the next section, the mean bias must also be considered). The regression plots provide the slope (m) and intercept (b).

Lower-Left Panel

Contains the 2D time-height cloud frequency of occurrence derived from ARSCL observations, the LES simulation, and 2D cloud masks from the ARSCL observations overlaid with those from the simulation. In the latter, green indicates where the simulation and observations both have cloud (model hit), red is where cloud is present only in the observations (model miss), and blue is where cloud is present only in the simulation (model false positive). Local solar time is indicated at the top and UTC at the bottom. A two-toned vertical scale is used where the vertical region below 5 km is expanded and above 5 km is reduced to show the full tropospheric cloud profiles without sacrificing details of the boundary-layer clouds; the partition between the regions is indicated by a dashed line. Simulated clouds lower than 160 m are not plotted since the ARSCL cloud mask begins at 160 m above the ground. Note that at this time the simulations have not been processed by an ARSCL simulator, which will be done in a future release. Also included is the observed LCL (blue line), and computations from the sonde profiles of the level of free convection (LFC, red +) and boundary-layer height from the Heffter (1980) method (green *).

Lower-Right Panel

Contains comparisons of the simulations to available sonde profiles at ~11:30 and ~17:30 solar time. The quantities compared are temperature, T (K), water vapor mixing ratio, qv (g kg^{-1}), potential temperature, θ (K), and equivalent potential temperature, θ_e (K). Also included are the Raman lidar profiles of temperature and water vapor mixing ratio above 300 m.

4.2 Skill Scores

Skill scores quantify model performance compared to observations. They monotonically increase with improved skill from 0 to 1, where 0 indicates no skill and 1 indicates perfect agreement in terms of the metric. LASSO draws as much as possible on skill scores commonly used within the community. The purpose for providing the skill scores is to help users find cases of suitable quality for their applications. As there may be a cluster of simulations with high skill scores for a given variable, there may be several suitable simulations that exceed a desired threshold. However, that number may dwindle as skill scores are considered for other variables important for the application of interest. Thus, the purpose of the skill scores is not to identify a "best" simulation, as what is best may depend on the application; rather, the purpose is to identify the "better" cases for consideration.

Time Series Skill Score

Model performance in terms of its time series is quantified using two skill scores, where one characterizes the agreement of the variation/shape of the time series and the other characterizes its mean. The Taylor skill score (equation 4 in Taylor [2001]), S_T, is used for the variation/shape of the distribution for a given variable, var (e.g., LWP), as

$$S_T(var) = \frac{4(1+R)}{\left[\left(\sigma_r + \frac{1}{\sigma_r}\right)^2 (1+R_0)\right]} \tag{1}$$

where σ_r is the normalized standard deviation given by model root-mean square (RMS) divided by the observed RMS, R is the correlation coefficient, and R_0 is the maximum correlation attainable, which is set to 1. Thus, if the correlation coefficient and normalized standard deviation are 1, the Taylor skill is 1. However, the Taylor skill alone cannot characterize the time series performance because it does not include information regarding the mean. To include this information, a skill score for the relative mean, S_{RM}, was developed as

$$S_{RM}(var) = \begin{cases} x \text{ for } x \le 1 \\ 1/x \text{ for } x > 1 \end{cases} \tag{2}$$

where x is the model mean divided by the observed mean. Through this formulation, the skill score has the range [0,1] and is symmetric around one. It is designed to quantify the relative difference from 1 and will yield the same value if the model underestimates or overestimates by the same factor. For example, two relative means that are different from observations by a factor of 2 on the low and high side, i.e., relative means of 0.5 and 2.0, would have the same skill score of 0.5 implying comparable performance relative to 1. Should users want to examine the relative means themselves, they are also included in the data bundles.

The Taylor and relative mean skill scores may be used in scatter plots to show how different simulations behave. See Figure 3 for an example. The closer the values are to the upper right-hand corner (1,1) the better the model performance in terms of the time series variable.

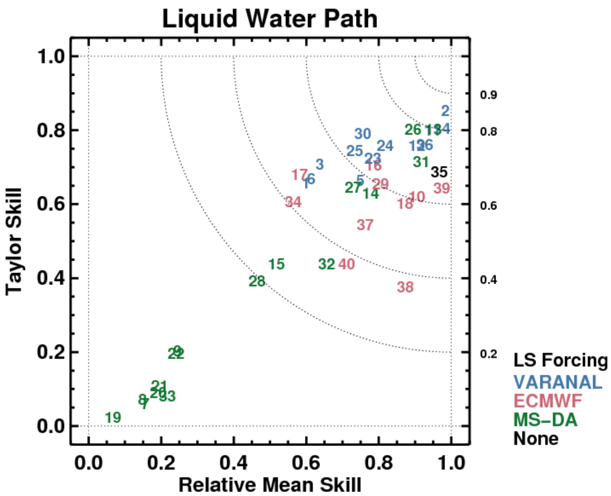


Figure 3. Scatter plot shows LWP Taylor skill score, S_T(LWP), and relative mean skill score, SRM(LWP), obtained from their time series comparison to observations. Each point represents a simulated day and the numbers indicate the simulation ID. Colors indicate the large-scale forcing used as indicated in the legend, where "none" is a simulation without large-scale forcing. The 40 simulations shown are from the 27-Jun-2015 case. Dashed curves are for constant values of net skill scores, S(LWP), notated at the right axis. The closer a point is to the upper-right-hand corner (1,1), the better the simulation performance for this metric.

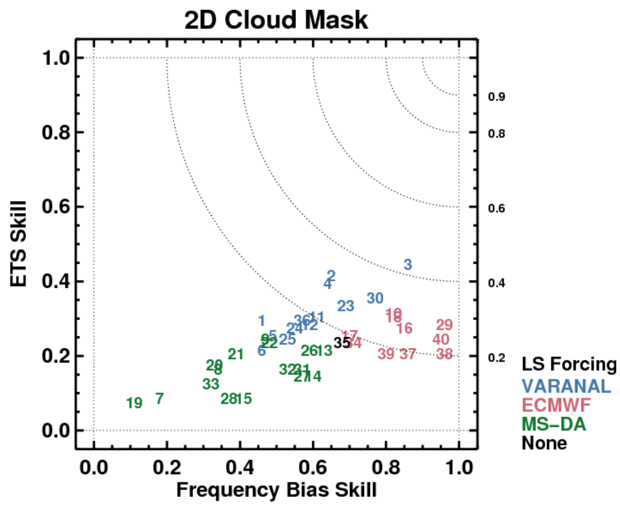


Figure 4. Scatter plot of 2D cloud mask skill scores shows S_{Bias} and S_{ETS}. S_{ETS} is often lower than other skill scores, but this happens uniformly such that the relative ordering of their values corresponds with relative visual agreement between simulations and observations in the 2D plots. When comparing skill scores for different variables, the relative ordering within each variable is most important rather than the absolute values between variables.

Finally, we found it useful to combine the Taylor and the relative mean skill scores into a single-variable net skill score, *S*, that quantifies how close a simulation is to (1,1) in Figure 3. To do so, we use

$$S(var) = \left(S_T(var) * S_{RM}(var)\right)^{\frac{1}{2}} \tag{3}$$

In this way, should the Taylor skill and relative mean skill both be 0.8, the combined single-variable net skill score would be returned as 0.8. This multiplication is preferable to, say, the square root of the sum of the squares, because it requires that both skill scores perform well for *S* to rank well; it does not allow a high score in one term to compensate for a much lower score in the other term. For example, if there were two sets of skill scores, (0.5, 0.5) and (1, 0.1), the former would score better using equation 3 while the latter would score better with the square root of the sum of the squares. Note that while high net skill scores can only be achieved by having both high Taylor and relative mean skills, medium-to-low values need not have the same level of consistency in their component skill scores since multiple combinations of high and low scores can yield the same net score.

2D Cloud Mask Skill Score

The 2D cloud mask in Figure 2 shows the ability of the model to simulate cloud-base height, vertical cloud extent, and the timing of cloud onset and decay. (As previously mentioned, application of an

ARSCL simulator is in progress that will enable a closer comparison of the simulated cloud fields to ARSCL observations.) The skill for the simulated 2D mask is quantified using the frequency bias and the equitable threat score (ETS), also called the Gilbert skill score (Mesinger and Black 1992; see "Forecast Verification Metrics" at https://hwt.nssl.noaa.gov/Spring_2012/),

$$ETS = \frac{(Hits - Hits_{random})}{(Hits + Misses + False\ alarms - Hits_{random})} \tag{4}$$

The ETS skill score is only applied to cloud below 5 km as boundary-layer clouds are the focus of this release. In this equation, *Hits* is the number of cloud pixels both correctly simulated and observed (green in bottom-left panel in Figure 2), *Misses* is the number of cloud pixels not simulated but observed (red), and *False alarms* is the number of cloud pixels simulated but not observed (blue). $Hits_{random}$ is the number of hits that might happen at random, given by,

$$Hits_{random} = \frac{(Hits + False\ alarms)(Hits + Misses)}{Total} \tag{5}$$

where *Total* is the total number of pixels below 5 km. ETS has the range of −1/3 to 1, where 0 indicates no skill. To have ETS conform to our other skills scores that are within the range [0,1], we truncate the low end of ETS at 0 and refer to it as the truncated ETS skill score, S_{ETS}.

The ETS-related frequency bias is the ratio of the frequency of simulated cloud pixels to the frequency of observed cloud pixels,

$$Bias = \frac{(Hits + False\ alarms)}{(Hits + Misses)} \tag{6}$$

The frequency bias is in terms of a ratio where a perfect score is 1; so, following the approach used to compute the skill for the relative mean, the skill score for the frequency bias, S_{Bias}, may be computed by replacing x in Eq 2 with Bias from Eq 6. The skill score is referred to as the frequency bias skill.

These two skill scores may be used in scatter plots to show how different simulations behave in terms of 2D cloud mask. See Figure 4 for an example. Generally speaking, simulations usually score better in terms of frequency bias skill than in ETS skill. While ETS skill can be a rather exacting skill score compared to its peers, it has the virtue of not giving errantly high values (i.e., a high value can be believed) and awarding values more consistent with visual inspections of plots.

Similar to the time series metrics, we find it useful to combine the ETS skill score with the frequency bias into a single-variable net skill score, *S*(*2D*), that quantifies how close a simulation is to (1,1) in Figure 4,

$$S(2D) = (S_{ETS} * S_{Bias})^{\frac{1}{2}} \tag{7}$$

Multivariable Net Skill Score

The advantage of having a net skill score for a variable (equations 3 and 7) is that it allows comparison of the net skill scores for two variables in scatter plots. An example is in Figure 5 for a comparison of the net time-series skill scores for LWP and CF(TSI). As for Figure 3, the closer the values are to the upper-right-hand corner, the better the model performance. The values for two net skill scores may be combined into a single multivariable net skill score similar to equation 3,

$$S(x,y) = \left(S(x) * S(y)\right)^{\frac{1}{2}} \tag{8}$$

where *S*(X) and *S*(Y) are the net skill scores for two variables. The multivariable net skill score for LWP and CF(TSI), *S*(LWP,CF(TSI)), is considered to be a special case that is referred to as the *1D cloud skill score*.

Following equation 8, it is also possible to combine a single-variable net skill score with a multivariable net skill score. For example, we combine the 1D cloud skill, *S*(LWP,CF(TSI)), with the 2D mask net skill, *S*(*2D*), to produce one skill score that characterizes model performance of the LWP and CF(TSI) time series *and* the 2D cloud mask,

$$S(S(2D), S(LWP, CF(TSI))) = (S(2D) * S(LWP, CF(TSI)))^{\frac{1}{2}} \tag{9}$$

This is a special case that is referred to as the *total cloud skill score*.

We note that while one could continue to combine more multivariable skill scores using equation 8 in series, one should be cautioned about diminishing returns. As noted earlier, a high net skill score can only be obtained by having two high input values, but the reason for a low skill score becomes more ambiguous since there are non-unique paths to the same low value. As more variables are added, the likelihood of a low score increases and, along with it, the ambiguity of comparing two low scores.

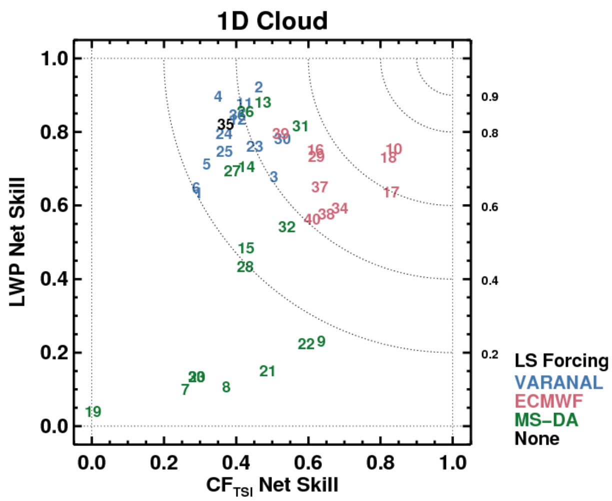


Figure 5. Scatter plot for two single-variable net-skill scores shows S(LWP) and S(CF[TSI]). In this way, simulation performance for two separate skill scores can be visualized. Should equation 8 use these two skill scores as input, it would result in a single multivariable net skill score for each pairing, S(LWP,CF[TSI]), that is referred to as the 1D cloud skill score.

5.0 LASSO Data Bundles and Tools

The overall concept for data organization within LASSO is the use of "data bundles." Each bundle is associated with a single simulation and contains the information necessary to repeat the simulation, the output from the simulation, subsetted output co-registered with observations, quicklook plots, and the skill scores and diagnostics for evaluating the simulation. Complementing the data bundles is a web tool called the Bundle Browser for quickly searching through simulations by querying skill score values and configuration details.

The browser additionally serves as an interface for users to view quicklook plots for simulations prior to downloading them, as well as summary statistics data and plots that conglomerate information across multiple simulations. A typical user uses the Bundle Browser to identify case dates and/or simulations of interest, as described in the next section. Links in the browser permit viewing of simulation-specific quicklook plots, providing a method for selecting data bundles for download from the ARM Data Center (ADC). Ordered data bundles are delivered in the form of two tar files per simulation, with the user able to request one or both of the tar files.

As an alternative to downloading the data bundles from ARM, users also have the option of requesting access to one of the ARM clusters supported by the ADC Computing Facility. The ADC maintains two clusters where ARM data can be staged to simplify analysis by users (Prakash et al. 2016). This is particularly useful for large data sets, such as LASSO, that can overwhelm some users' available computing resources. More information and a link to request access to the ARM clusters can be obtained from https://www.arm.gov/capabilities/computing-resources.

5.1 The LASSO Bundle Browser

The LASSO Bundle Browser provides an interactive web interface for users to find simulations of interest through examination of the LES performance relative to select ARM observations. It allows users to visualize the LASSO data bundle diagnostics and skill scores on the fly using plots and tables while also providing links to data. Values for plots and the data table are fetched dynamically from a Cassandra NoSQL database. The conditional query to retrieve data is formed based on the user-selected traits in the browser.

The LASSO Bundle Browser is available for use at https://adc.arm.gov/lassobrowser, and Figure 6 shows an example of the display. The left-hand side is occupied by selection options within (A) and (C) whose results are displayed to the right, plus case-specific overview plots in (B). The selection options and display features are described below referencing the labeling in Figure 6.

A. Expandable menus allow the user to select multiple dates and a single measurement type for any combination of traits that characterize the forcing and model configurations. "Select all" is an option within each trait category and the "Select All (Excludes Date)" button at the top of (A) selects all forcing and model configuration options while not altering the date selection(s). This essentially results in selecting all possible simulations for the selected date(s). Clicking the "Submit" button at the bottom of (A) applies the selected query and updates the plot in (D) for the Taylor diagrams, skill score plots, scatter plots, and time series plots. These plots are created using the D3.js and highcharts software libraries. The data table in (E) is also updated, which displays the skill scores and provides links for downloading the resulting simulations.

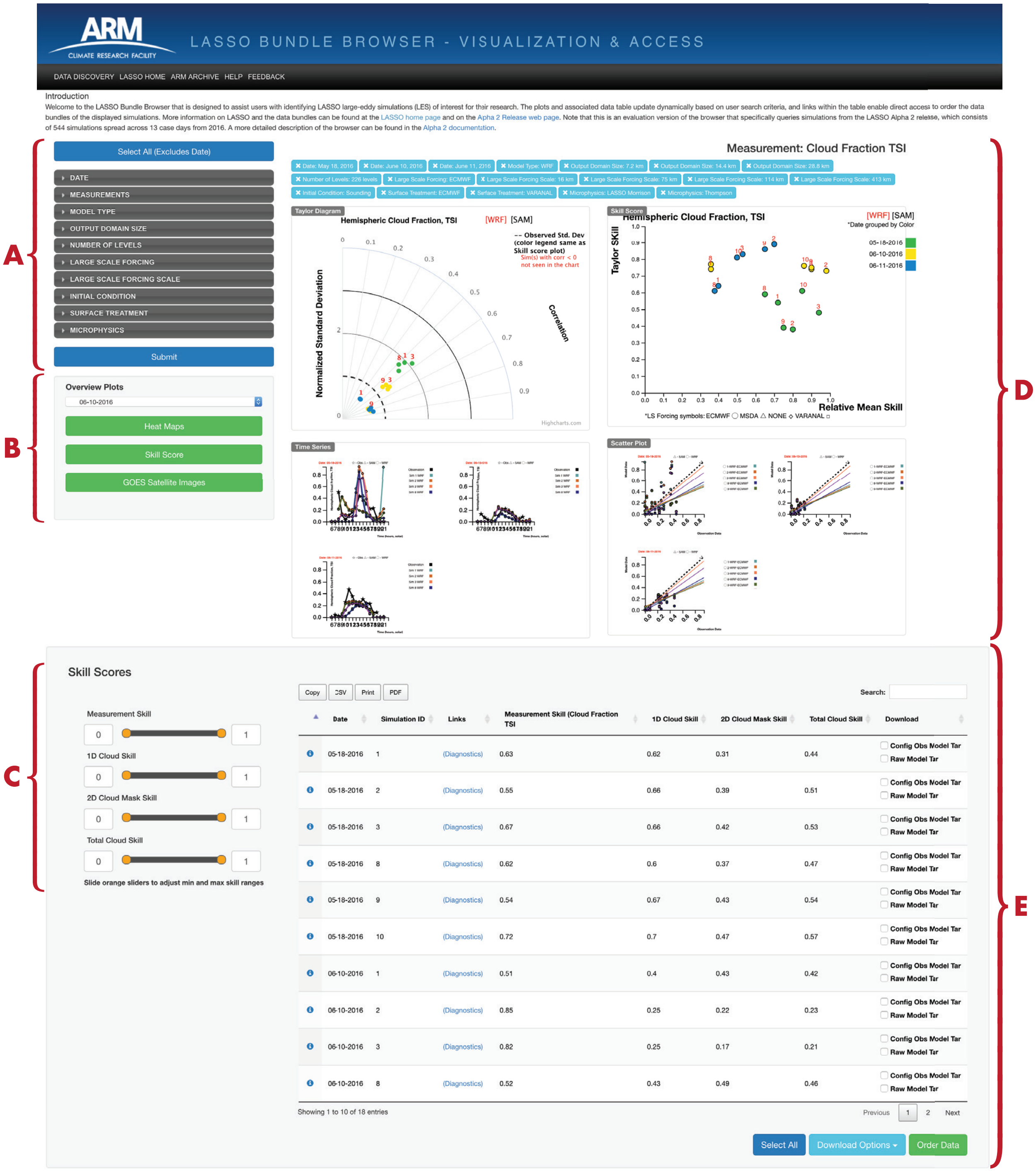


Figure 6. LASSO Bundle Browser interface at https://adc.arm.gov/lassobrowser. See text for descriptions.

B. Precomputed overview plots for all simulations and variables for a selected day are available for display as heat maps (Figure 1) and skill scores (e.g., Figures 3–5). These overview plots can assist users to locate the variables of interest for use in (A). This section also provides a link to Geostationary Operational Environmental Satellite (GOES) images for the selected date. This section has its own date dropdown menu to view images for a single date, which is independent of the multiple-date selection capability in (A).

C. Slide rulers allow choosing the range of net skill scores that are displayed in (D) and (E). Place the mouse over the label for a brief description of the variable. Moving the sliders or entering specific values in the boxes will update the table and plots to only include simulations within the selected score values.

D. Dynamically updated plots are shown in (D) for the selected measurement and date(s). The simulations included in the plots reflect the query selections from (A) and (C), and the selections from (A) are reflected as "breadcrumbs" (in blue) above the plots. The plot types include Taylor diagrams, skill score plots, scatter plots, and time series plots. The black line in the time series plot is the observations. The plots are interactive; mouse over the points to see the simulation ID and coordinate values. Click on a given plot to enlarge it and print.

E. Tabulated results are given for the net skill scores of the simulations selected via (A) and (C). The "i" circle to the left of the simulation identification number (ID) provides a short readme file containing a detailed summary of the simulation run configuration that includes information not available in (A). "Diagnostics" hyperlinks are provided in the column to the right of the simulation ID. These links return the precomputed diagnostic plots shown in Figure 2 for each simulation (e.g., 2D cloud-mask time series plots, etc.). Also within the table are arrows at the top of each column that order the table entries according to the column variable and the order may be reversed by a subsequent click. The "Search" box finds a given value within the table. Above the simulation ID are options to "print" or "copy" the results or download as a "CVS" or "PDF" file. Below the table are buttons for ordering data bundle tar files selected within the table.

There are two methods for downloading LASSO data bundles. The first is via section (E) of the Bundle Browser. The table of simulations in the browser includes a column with two checkboxes for each type of tar file in the bundle. Users can check which tars they want to download, or alternatively, they can use the "Select All" button to select all tar files in the table for download. A "Download Options" button permits selection of either receiving the data via either file transfer protocol (FTP) or Globus. After clicking the "Order Data" button, users are provided with a login pop-up to enter their credentials. Once entered, the order is created and the its ID will be displayed. Users then will receive an email with download instructions when the order is ready.

The second method for downloading the data bundles is via the Data Discovery application (https://www.adc.arm.gov/discovery/). Data Discovery is a timeline-based application designed for ordering data from a time range. To search for LASSO data, the model-related metadata traits are made available for narrowing one's search. When users search for LASSO data, every simulation will be listed as a datastream embracing the timeline structure of the Data Discovery application. Even though data are available for only 13 dates, the color on the timeline will show for the entire time range. Note that this indicates the data as model data and not the available dates. To narrow down to a single date, the user can use the "simulation date" search façade by selecting a single date. Data Discovery can be easier than the Bundle Browser for downloading LASSO data for a particular simulation from all days. However, multiple requests must be made if the desired dates are not consecutive within the available cases. It is generally quicker to use the Bundle Browser to download data for non-consecutive dates. Integration of LASSO datastreams (LASSO being a new kind of data for ARM) into Data Discovery is an ongoing effort and we would like to hear from users regarding how to improve the user experience.

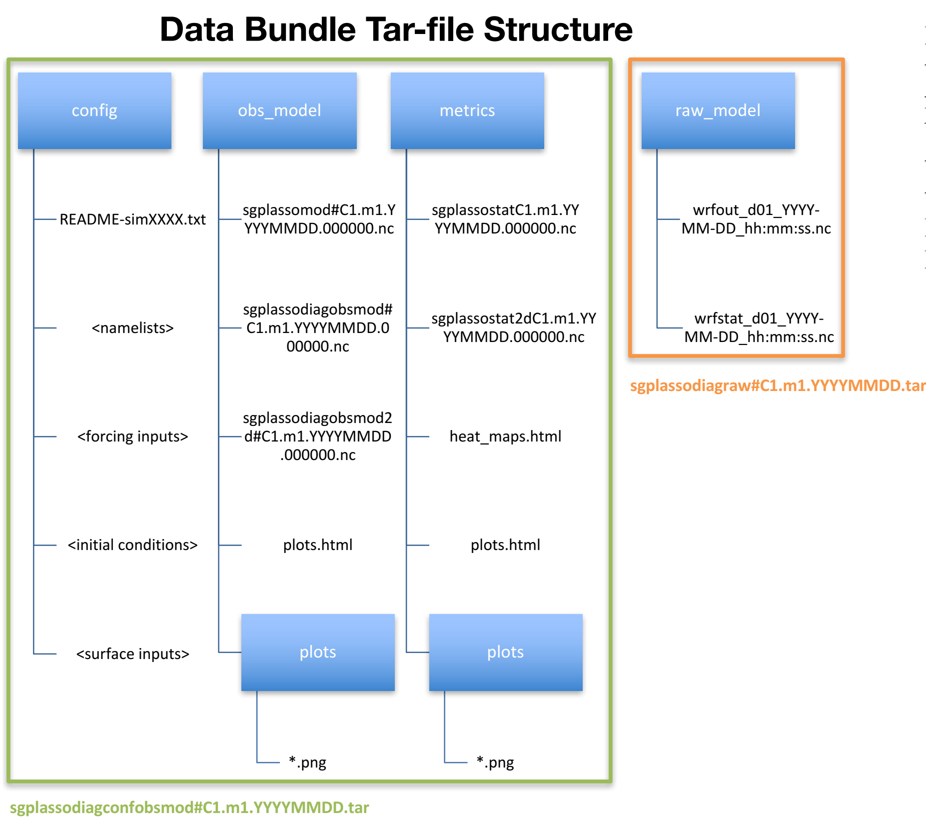


Figure 7. Schematic shows the contents of the two tar files associated with each data bundle. Users can choose to download one or both of these tar files when ordering LASSO data from the ARM Data Center.

5.2 Organization of Data Bundle Files

Once ordered, users download data bundles by simulation. Depending on whether one or both are requested, users receive up to two tar files per simulation. The first has the filename structured as *sgplassodiagconfobsmod#C1.m1.YYYYMMDD.tar*, and it contains three primary directories, as shown in Figure 7. The *config* directory contains model inputs and configuration files necessary for reproducing the simulation, and the *obs_model* directory contains subsetted model output and observations that are co-registered in space and time. Quicklook plots associated with the simulation are also available in a *plots* directory with *obs_model*. These are the same quicklook plots available directly from the Bundle Browser. The *metrics* directory contains the results of the skill scores comparing the simulation against observations. It also has a plots subdirectory that contains plots of the skill scores comparing all the simulations on the selected date. These are static quicklooks similar to what can be generated dynamically via the Bundle Browser. The files in the *metrics* directory have information for all simulations on the selected date. The contents of this directory are identical for all simulations for the selected date.

The second tar file has a filename structured as *sgplassodiagraw#C1.m1.YYYYMMDD.tar*. This tar file contains the raw model output and is, therefore, noticeably larger. For WRF simulations, this will include multiple wrfout files plus a wrfstat file as shown in Figure 7. For SAM simulations, this will contain the corresponding SAM 1D, 2D, and 3D files in netCDF format. The SAM model initially outputs in binary, which has been converted to netCDF. Note that the statistics variables from both SAM and WRF are time averaged over a 10-minute period with sampling every minute. *Note that the time label for these SAM statistics files represents the middle of the averaging period, whereas the time label represents the end of the period for the WRF statistics file*. The instantaneous output has time labels corresponding with the time step when the output occurs.

Details regarding the contents of the files are provided in Appendix C, "LASSO Alpha 2 File Contents," which contains header dumps of the netCDF files.

5.3 Support Files

Several support files are also available to users in addition to the data bundles. The support files are small and accessible directly from web links, as opposed to the data bundles, which are only accessible by downloading them from the ARM Data Center due to their large size. From the Bundle Browser there are links to:

- GOES visible image animation loops for each case date to provide a perspective of the cloud field.
- Plots of skill scores spanning all simulations in a case day. These are static versions appropriate for referencing and using in presentations compared to the dynamically generated versions produced directly by the browser. Additional varieties are also available, such as heat map plots showing the range of simulation behavior for a given case date.
- Quicklook plot for each case, where these are duplicates of the quicklooks within the data bundles, but are more easily accessible without going through the data ordering and download process.

In addition, from the LASSO Alpha 2 web page, there are links to the documentation and a spreadsheet listing all the simulations in the release. This spreadsheet is useful for quickly mapping simulation IDs and metadata traits, such as which simulations use certain forcing types, microphysics settings, model type, etc.

6.0 Alpha 2 Case Descriptions

Thirteen days were chosen during the period May to August 2016 for testing of model configurations and forcing calculation methodologies. The primary criteria used to select the cases were that they be classic shallow convective days and that sufficient ARM data be available for each case for forcing and evaluation purposes. The primary measurement products of interest were ARSCL, TSI, AERIoe, SONDE, and the Raman lidar. An initial scan of TSI movies revealed days with relevant cloud fractions, which were then verified as shallow convection with ARSCL. Days with too much missing data from primary measurements were excluded. Some cases were also selected that would be considered particularly challenging shallow convection days because they are near large synoptic events in the SGP region, and thus are influenced by atmospheric flows beyond the local diurnal cycle that is the primary forcer of classic shallow convection. This was done to better understand the sensitivity of the forcing to the conditions in the region surrounding the Central Facility.

The following pages show the overall meteorological conditions for each simulated day. The intent is to provide a general context for understanding the overall model behavior. Each case is accompanied by a series of figures illustrating the synoptic conditions and type of clouds present around SGP. Animations of the GOES visible channel for each date are also available using the "GOES Satellite Images" button within the left column of the Bundle Browser.

The following 2016 case days are included in LASSO Alpha 2 and details for these days are included in this section:

18-May-2016	19-Jun-2016	18-Aug-2016
30-May-2016	25-Jun-2016	19-Aug-2016
10-Jun-2016	16-Jul-2016	30-Aug-2016
11-Jun-2016	19-Jul-2016	
14-Jun-2016	20-Jul-2016	

In addition, the following 2015 case days are included as a supplement to LASSO Alpha 2 and details for these days are described in the *Description of the LASSO Alpha 1 Release* technical report (Gustafson et al. 2016):

6-Jun-2015	27-Jun-2015	29-Aug-2015
9-Jun-2015	1-Aug-2015	

Synoptic Conditions for 18-May-2016

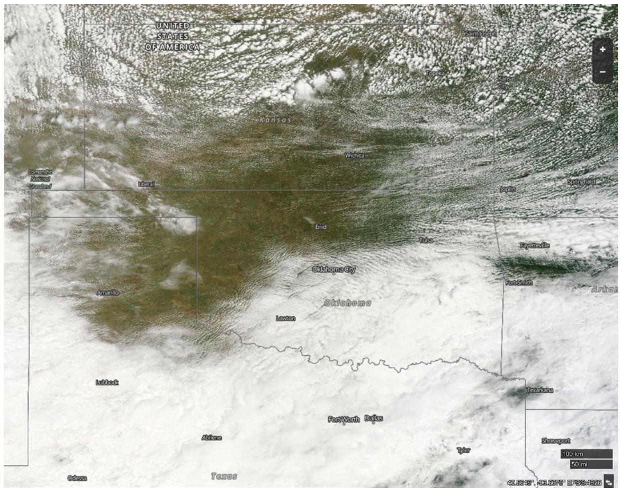


Figure 8. Moderate Resolution Imaging Spectro-radiometer (MODIS) Terra corrected reflectance for 18-May-2016. Acquired from National Aeronatics and Space Administration (NASA), Earth Observing System Data and Information System (EOSDIS) Worldview, https://go.nasa.gov/2uU0BY6.

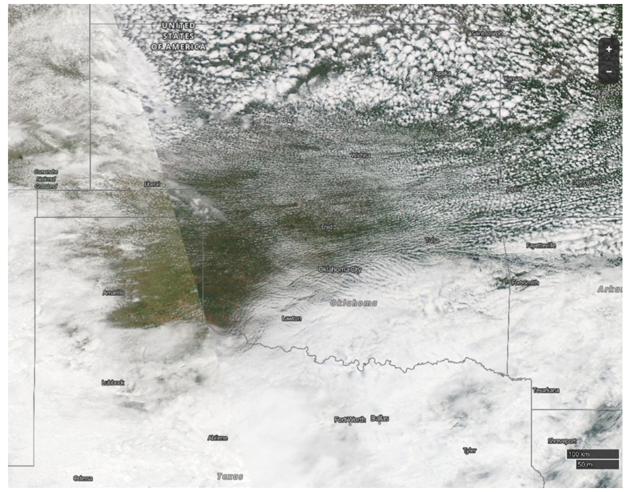


Figure 9. MODIS Aqua corrected reflectance for 18-May-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uU7E2T.

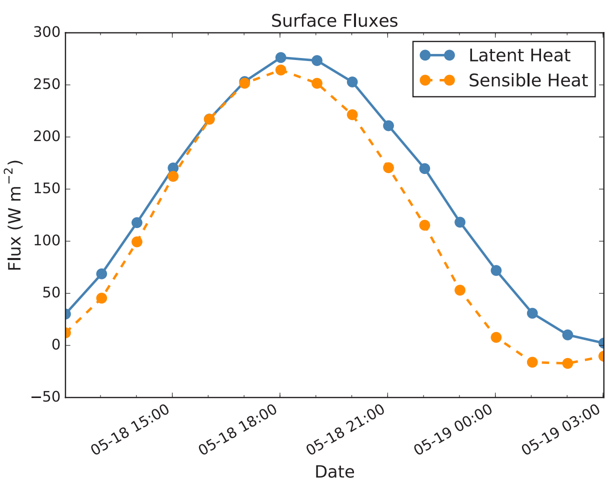


Figure 10. Surface sensible (orange, dashed) and latent (blue, solid) heat fluxes averaged for the SGP region. Hourly values taken from the variational analysis product. Date labels are UTC.

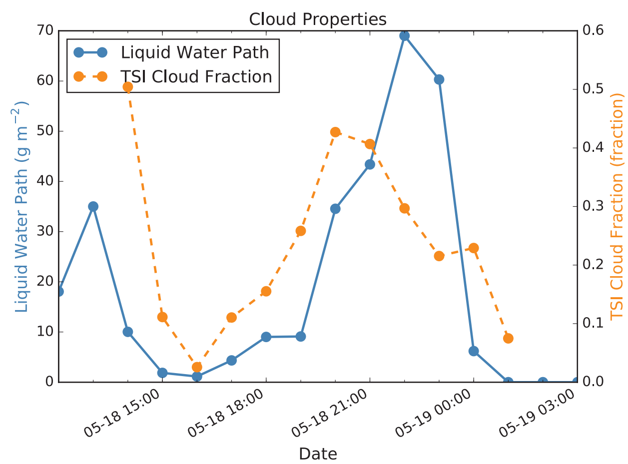


Figure 11. Cloud fraction (orange, dashed) from the TSI and cloud liquid water path (blue, solid) from a hybrid blending of MWRRet and AERIoe at the SGP Central Facility. Date labels are UTC.

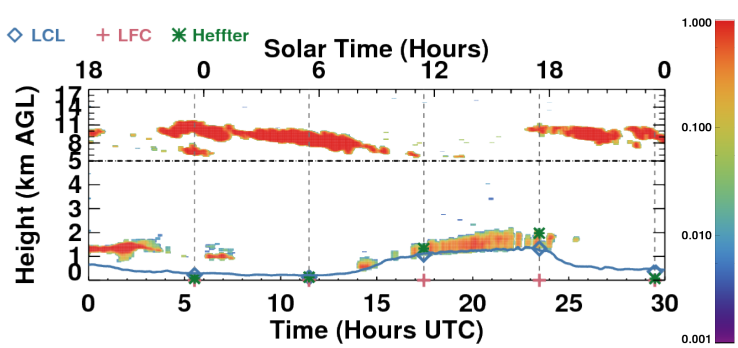


Figure 12. Cloud fraction profiles derived from the KAZR-ARSCL for 18-May-2016 at the Central Facility. Note the non-linear vertical axis that emphasizes the lower troposphere. Also indicated are the LCL, LFC, and PBL height based on the Heffter methodology, each of which are calculated from the SONDE product.

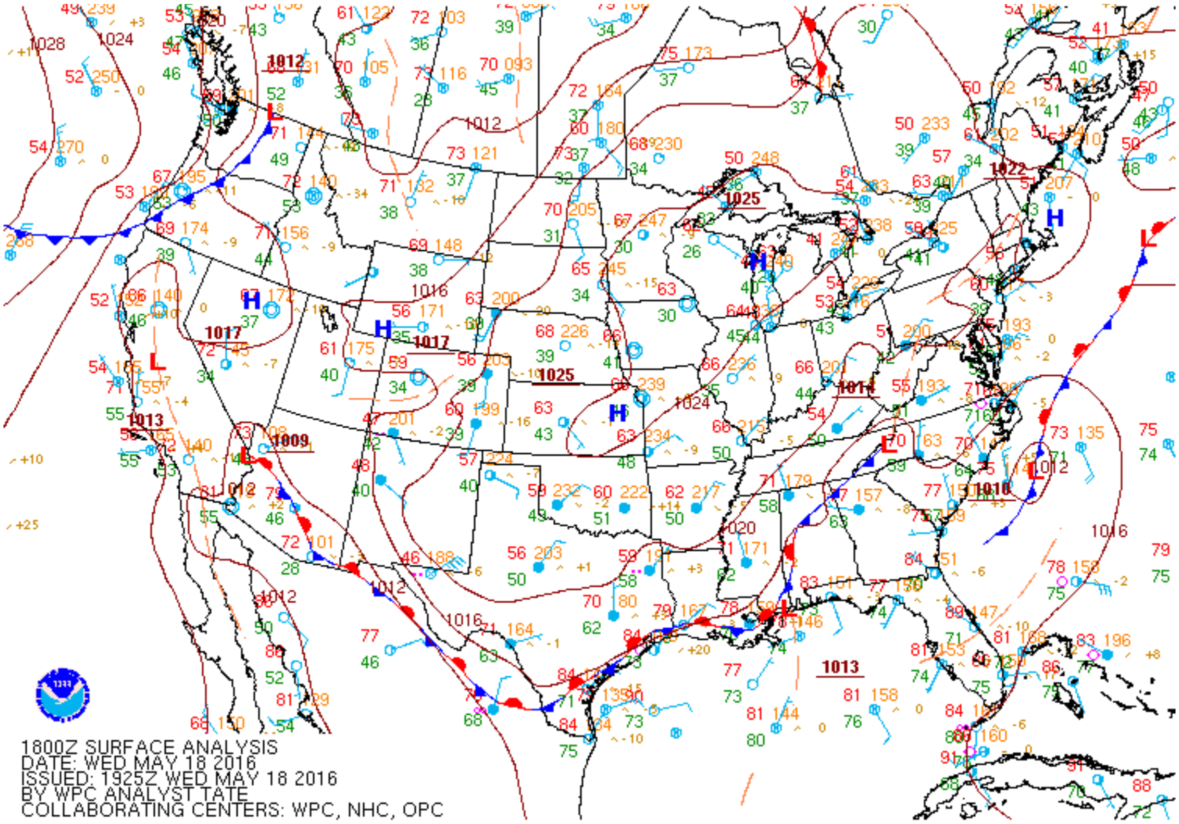


Figure 13. Surface analysis for 18-May-2016 18 UTC. Acquired from National Weather Service (NWS) Weather Prediction Center, http://www.wpc.ncep.noaa.gov/archives/web_pages/sfc/sfc_archive_maps.php?arc-date=05/18/2016&selmap=2016051818&maptype=namussfc.

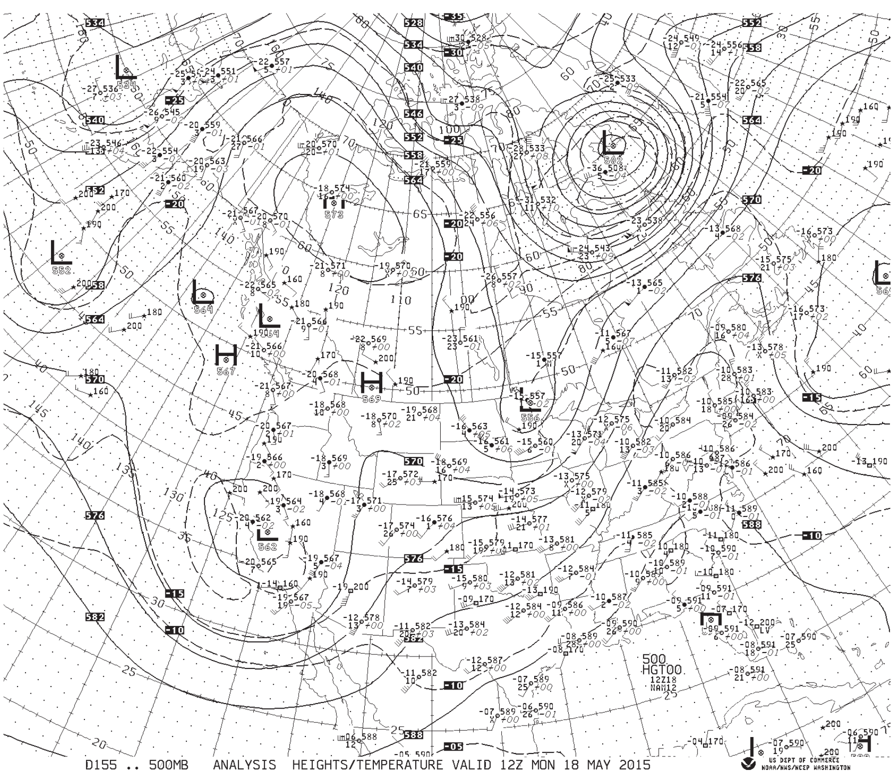


Figure 14. 500 hPa synoptic map for 18-May-2016 12 UTC. Acquired from Storm Research and Consulting, http://www.stormresearch.com/ncep/2015/2015_05/2015051818_500.tif

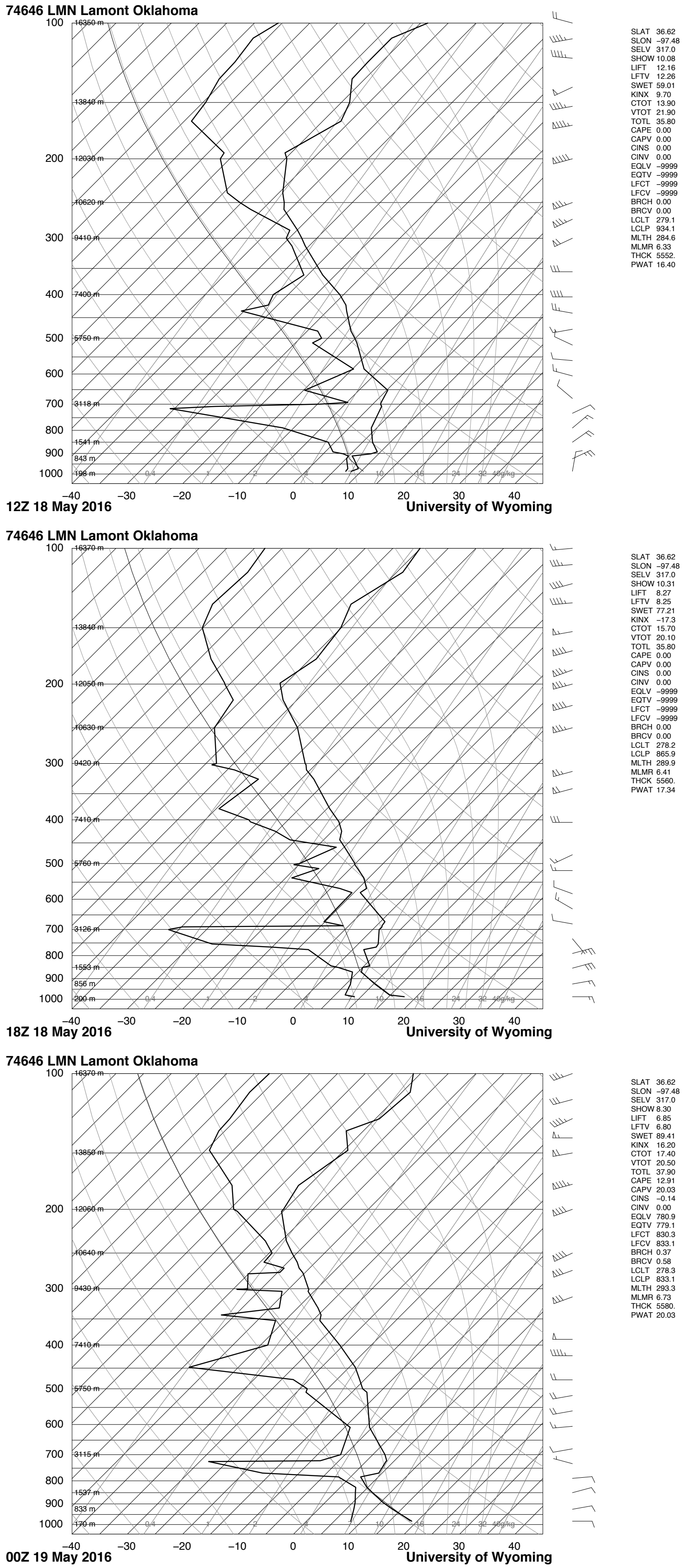


Figure 15. Skew-T log-P diagrams for Lamont, Oklahoma, for the 18-May-2016 LASSO case. Acquired from University of Wyoming, http://weather.uwyo.edu/upper-air/sounding.html.

Synoptic Conditions for 30-May-2016

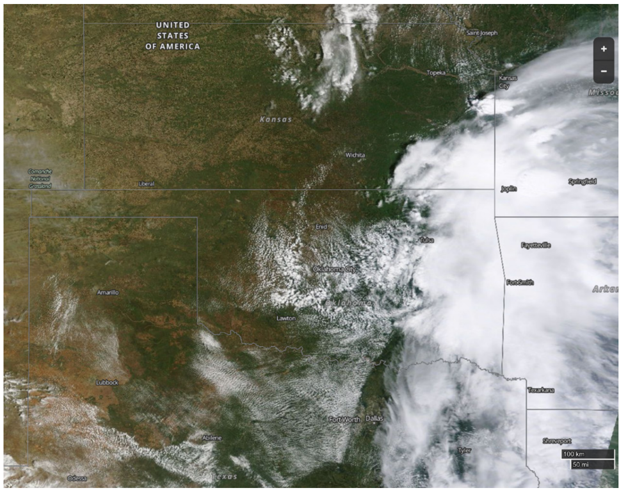


Figure 16. MODIS Terra corrected relectance for 30-May-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uTyF6A.

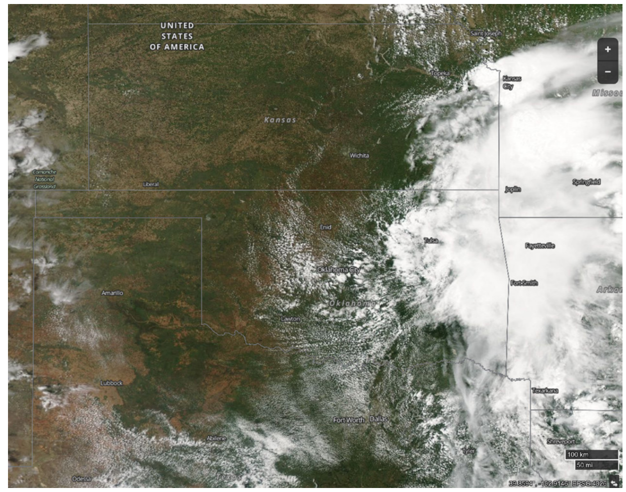


Figure 17. MODIS Aqua corrected relectance for 30-May-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uTK5XL.

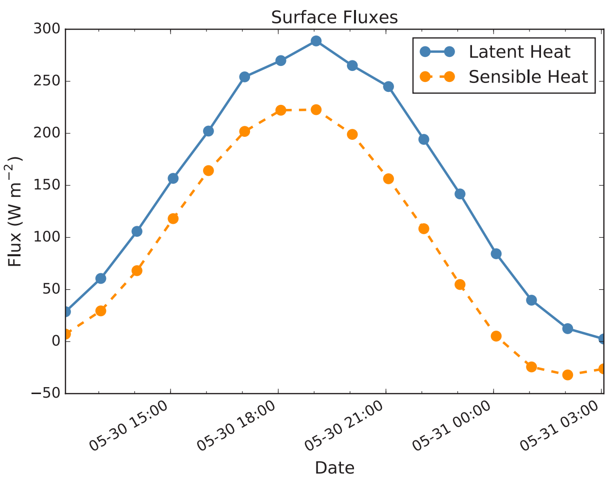


Figure 18. Surface sensible (orange, dashed) and latent (blue, solid) heat fluxes averaged for the SGP region. Hourly values taken from the variational analysis product. Date labels are UTC.

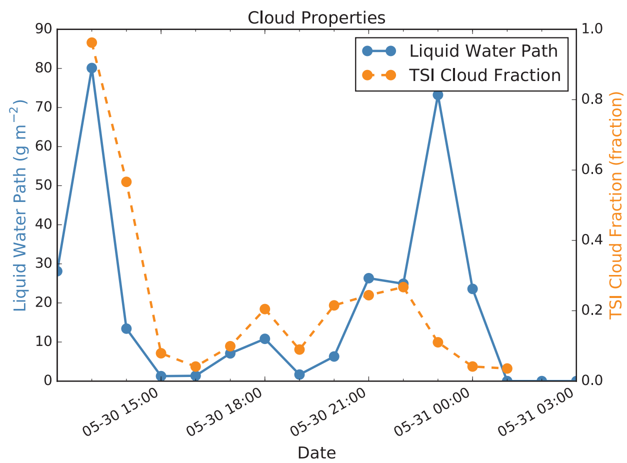


Figure 19. Cloud fraction (orange, dashed) from the TSI and cloud liquid water path (blue, solid) from a hybrid blending of MWRRet and AERIoe at the SGP Central Facility. Date labels are UTC.

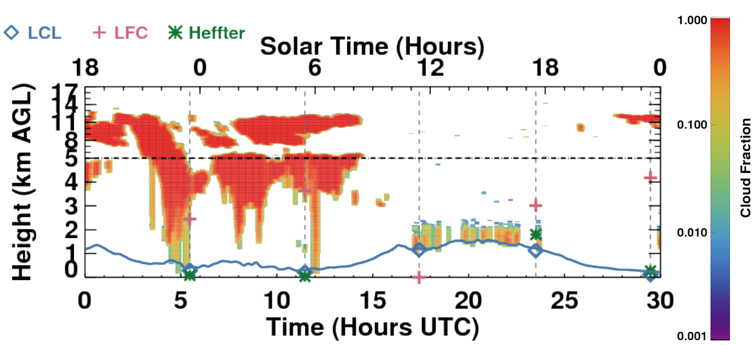


Figure 20. Cloud fraction profiles derived from the KAZR-ARSCL for 30-May-2016 at the Central Facility. Note the non-linear vertical axis that emphasizes the lower troposphere. Also indicated are the LCL, LFC, and PBL height based on the Heffter methodology, each of which are calculated from the SONDE product.

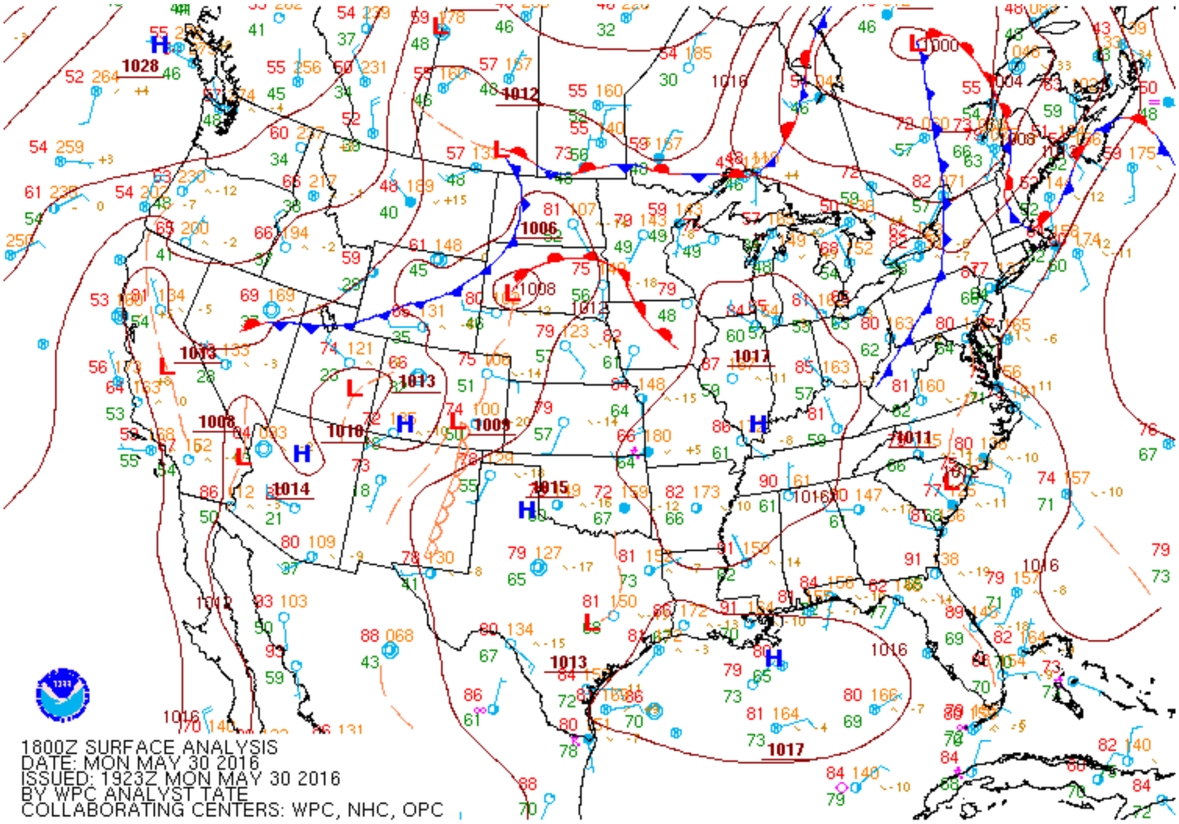


Figure 21. Surface analysis for 30-May-2016 18 UTC. Acquired from NWS Weather Prediction Center, http://www.wpc.ncep.noaa.gov/archives/web_pages/sfc/sfc_archive_maps.php?arcdate=05/30/2016&selmap=2016053018&maptype=namussfc.

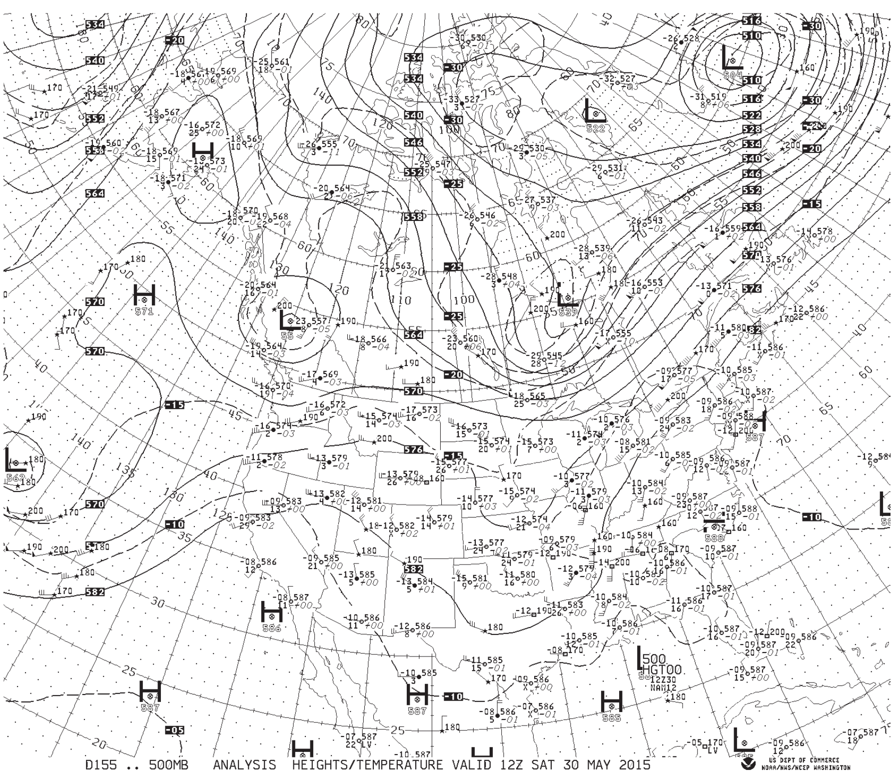


Figure 22. 500 hPa synoptic map for 30-May-2016 12 UTC. Acquired from Storm Research and Consulting, http://www.stormresearch.com/ncep/2015/2015_05/2015053012_500.tif.

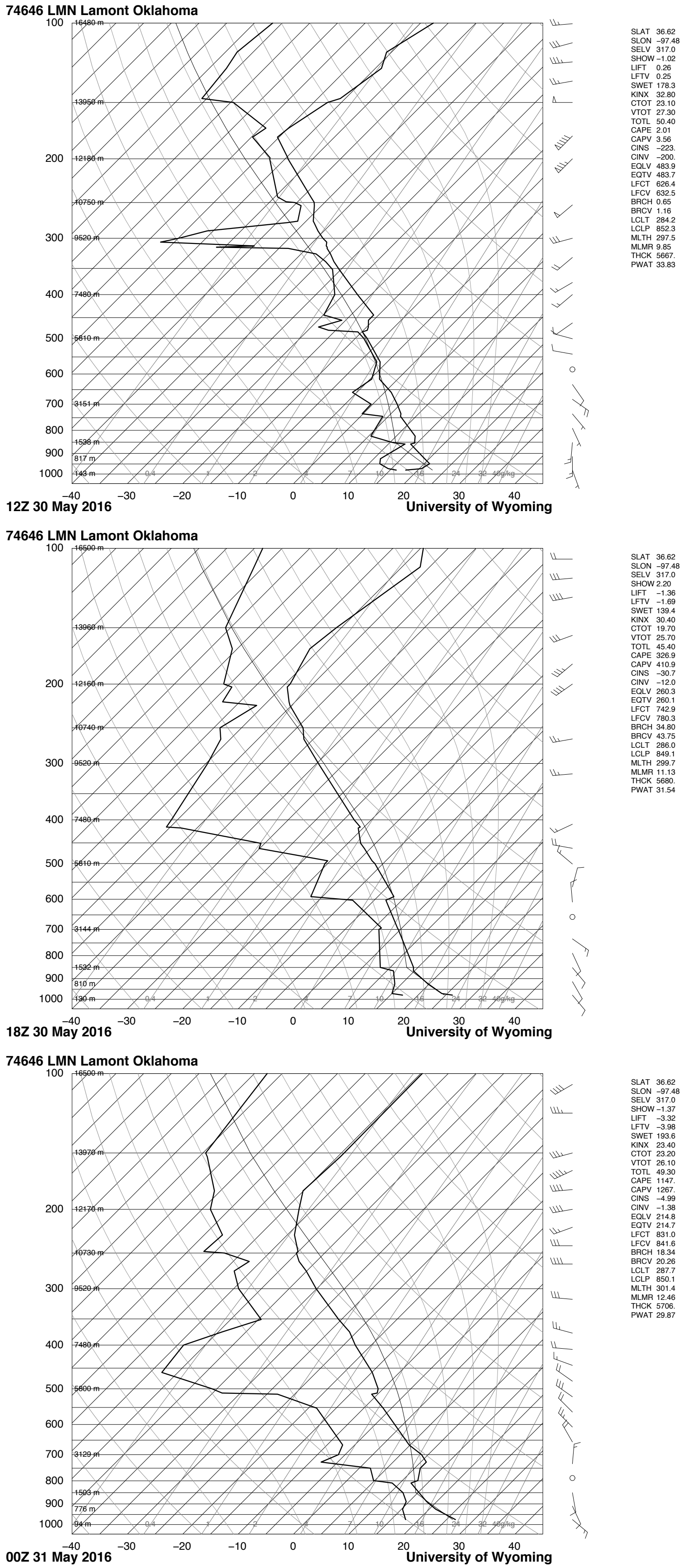


Figure 23. Skew-T log-P diagrams for Lamont, Oklahoma, for the 30-May-2016 LASSO case. Acquired from University of Wyoming, http://weather.uwyo.edu/upper-air/sounding.html.

Synoptic Conditions for 10-Jun-2016

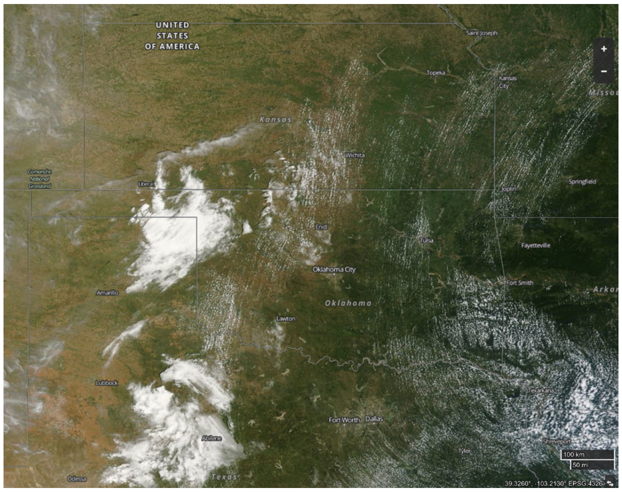


Figure 24. MODIS Terra corrected reflectance for 10-Jun-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2tRN8mz.

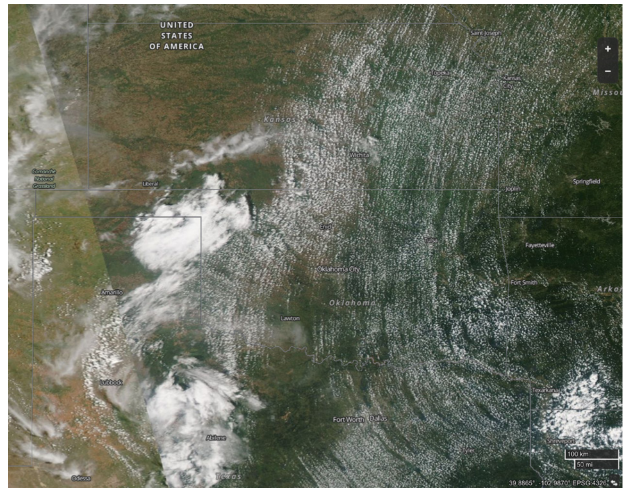


Figure 25. MODIS Aqua corrected reflectance for 10-Jun-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2tS64lj.

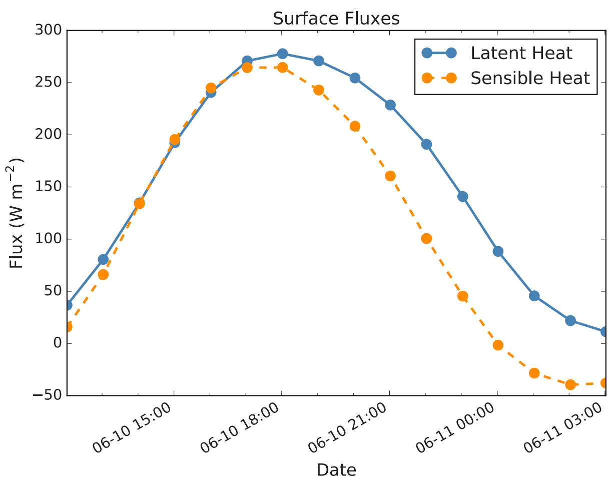


Figure 26. Surface sensible (orange, dashed) and latent (blue, solid) heat fluxes averaged for the SGP region. Hourly values taken from the variational analysis product. Date labels are UTC.

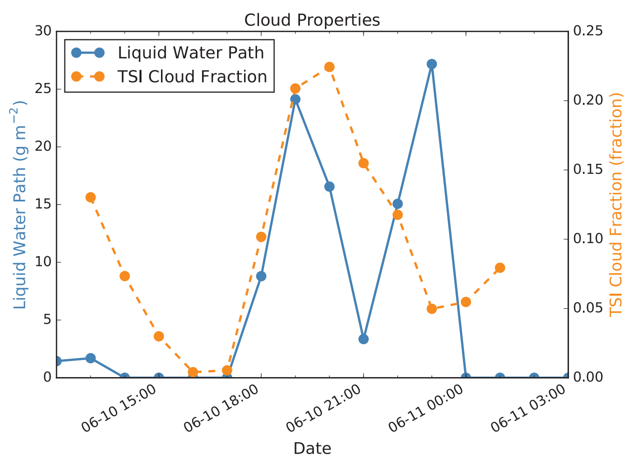


Figure 27. Cloud fraction (orange, dashed) from the TSI and cloud liquid water path (blue, solid) from a hybrid blending of MWRRet and AERIoe at the SGP Central Facility. Date labels are UTC.

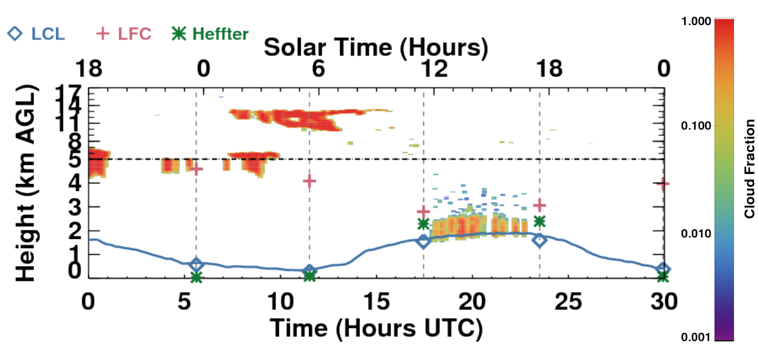


Figure 28. Cloud fraction profiles derived from the KAZR-ARSCL for 10-Jun-2016 at the Central Facility. Note the non-linear vertical axis that emphasizes the lower troposphere. Also indicated are the LCL, LFC, and PBL height based on the Heffter methodology, each of which are calculated from the SONDE product.

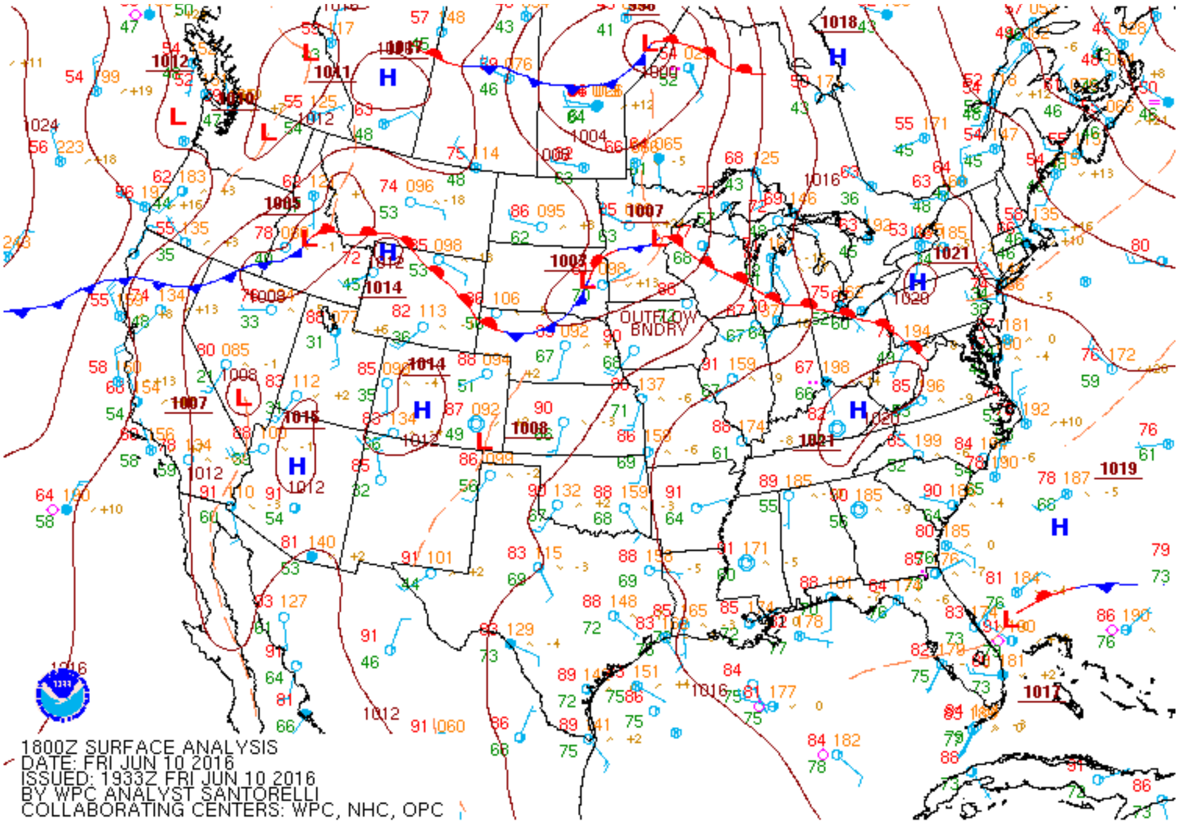


Figure 29. Surface analysis for 10-Jun-2016 18 UTC. Acquired from NWS Weather Prediction Center, http://www.wpc.ncep.noaa.gov/archives/web_pages/sfc/sfc_archive_maps.php?arc-date=06/10/2016&selmap=2016061018&maptype=namussfc.

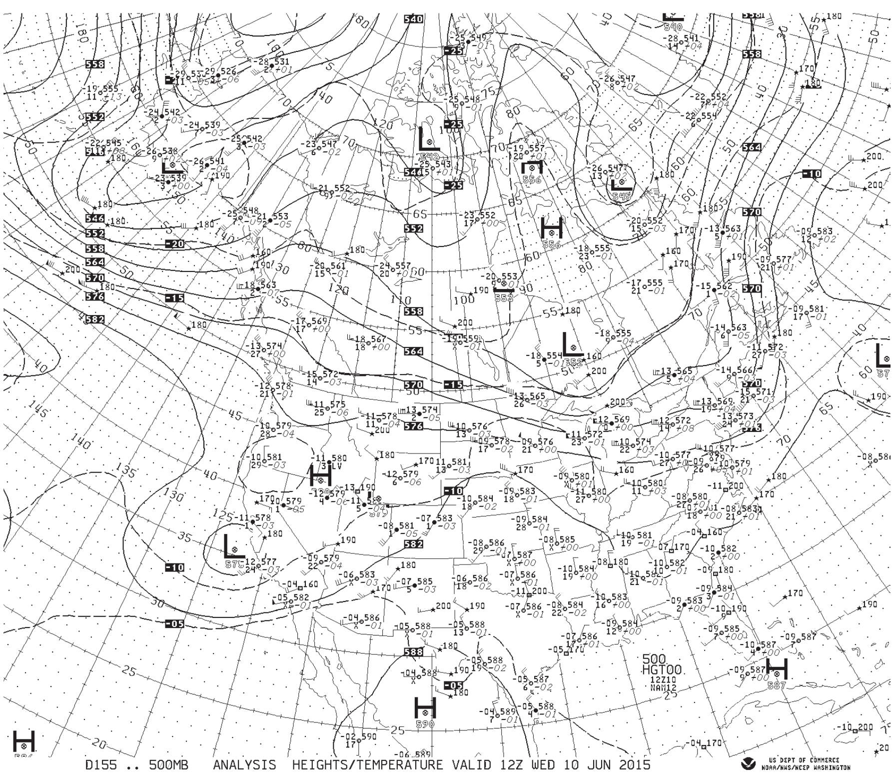


Figure 30. 500 hPa synoptic map for 10-Jun-2016 12 UTC. Acquired from Storm Research and Consulting, http://www.stormresearch.com/ncep/2015/2015_06/2015061012_500.tif.

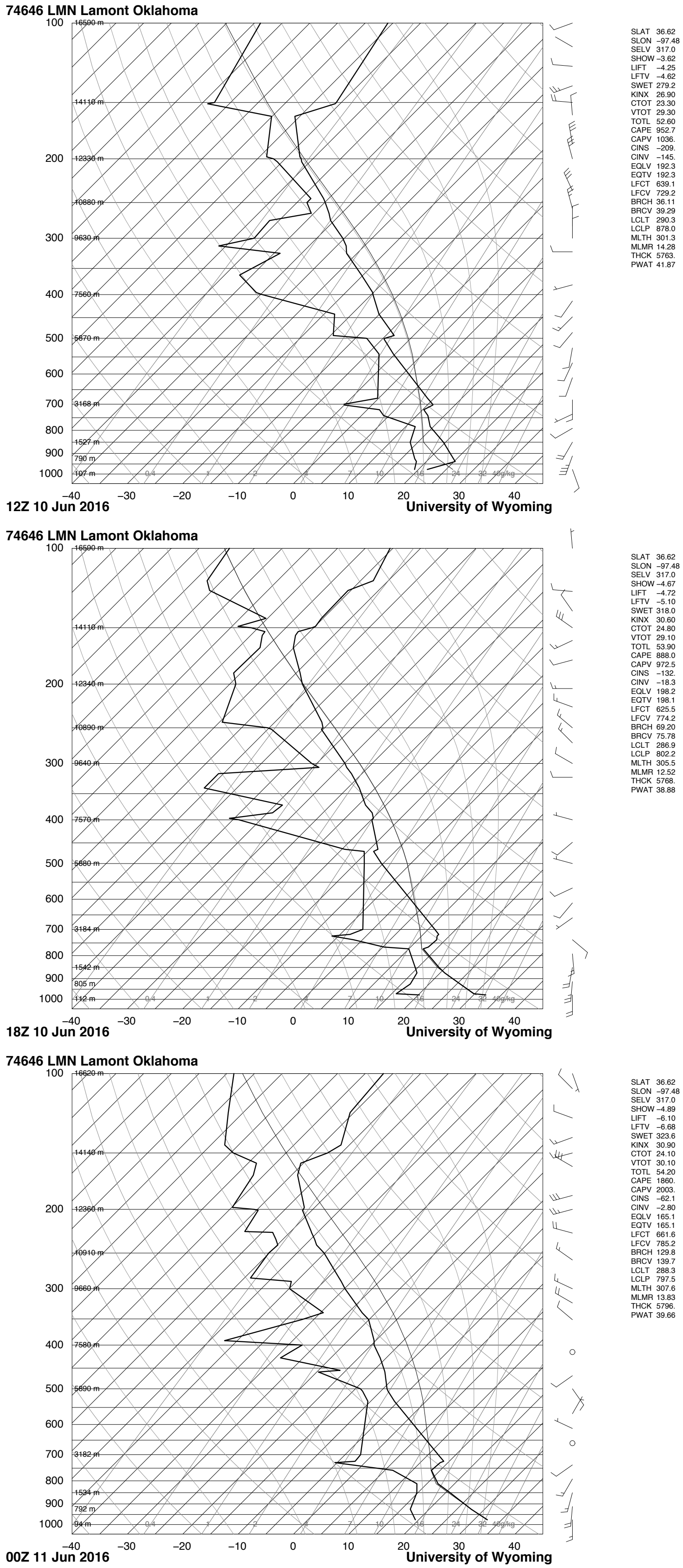


Figure 31. Skew-T log-P diagrams for Lamont, Oklahoma, for the 10-Jun-2016 LASSO case. Acquired from University of Wyoming, http://weather.uwyo.edu/upper-air/sounding.html.

Synoptic Conditions for 11-Jun-2016

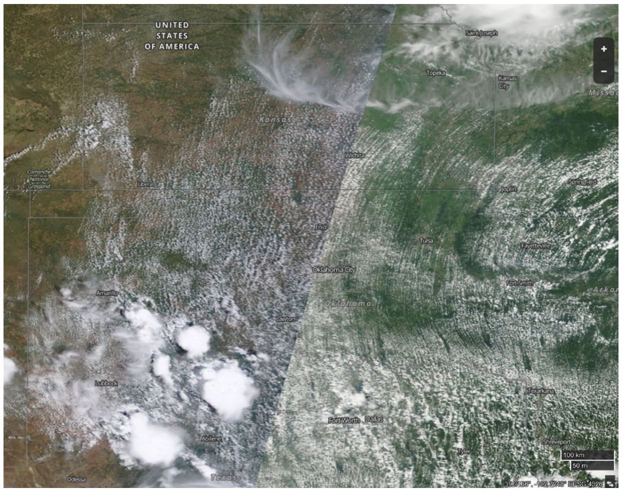


Figure 32. MODIS Terra corrected reflectance for 11-Jun-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uTwU9s.

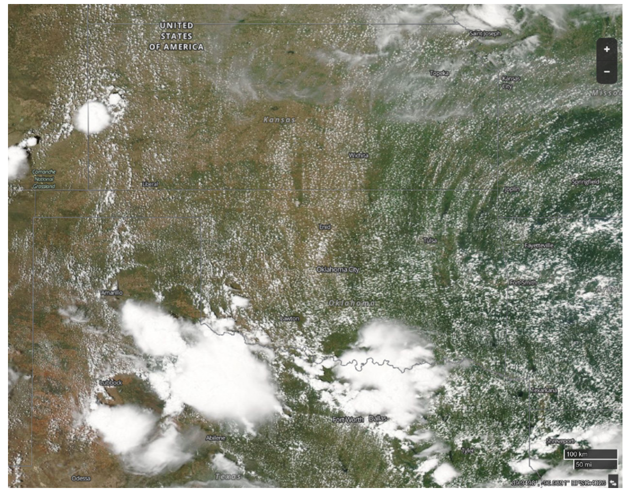


Figure 33. MODIS Aqua corrected reflectance for 11-Jun-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uTzAnl.

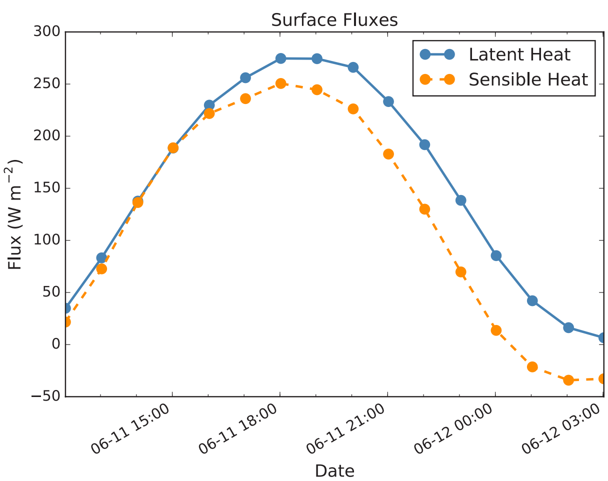


Figure 34. Surface sensible (orange, dashed) and latent (blue, solid) heat fluxes averaged for the SGP region. Hourly values taken from the variational analysis product. Date labels are UTC.

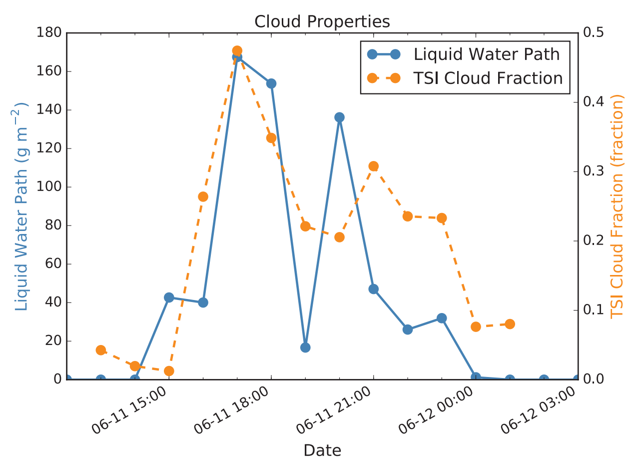


Figure 35. Cloud fraction (orange, dashed) from the TSI and cloud liquid water path (blue, solid) from a hybrid blending of MWRRet and AERIoe at the SGP Central Facility. Date labels are UTC.

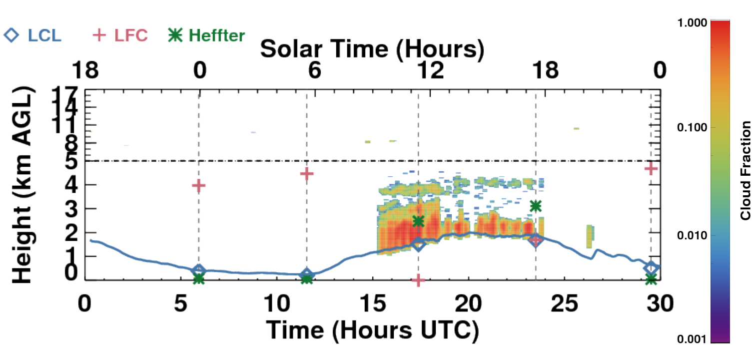


Figure 36. Cloud fraction profiles derived from the KAZR-ARSCL for 11-Jun-2016 at the Central Facility. Note the non-linear vertical axis that emphasizes the lower troposphere. Also indicated are the LCL, LFC, and PBL height based on the Heffter methodology, each of which are calculated from the SONDE product.

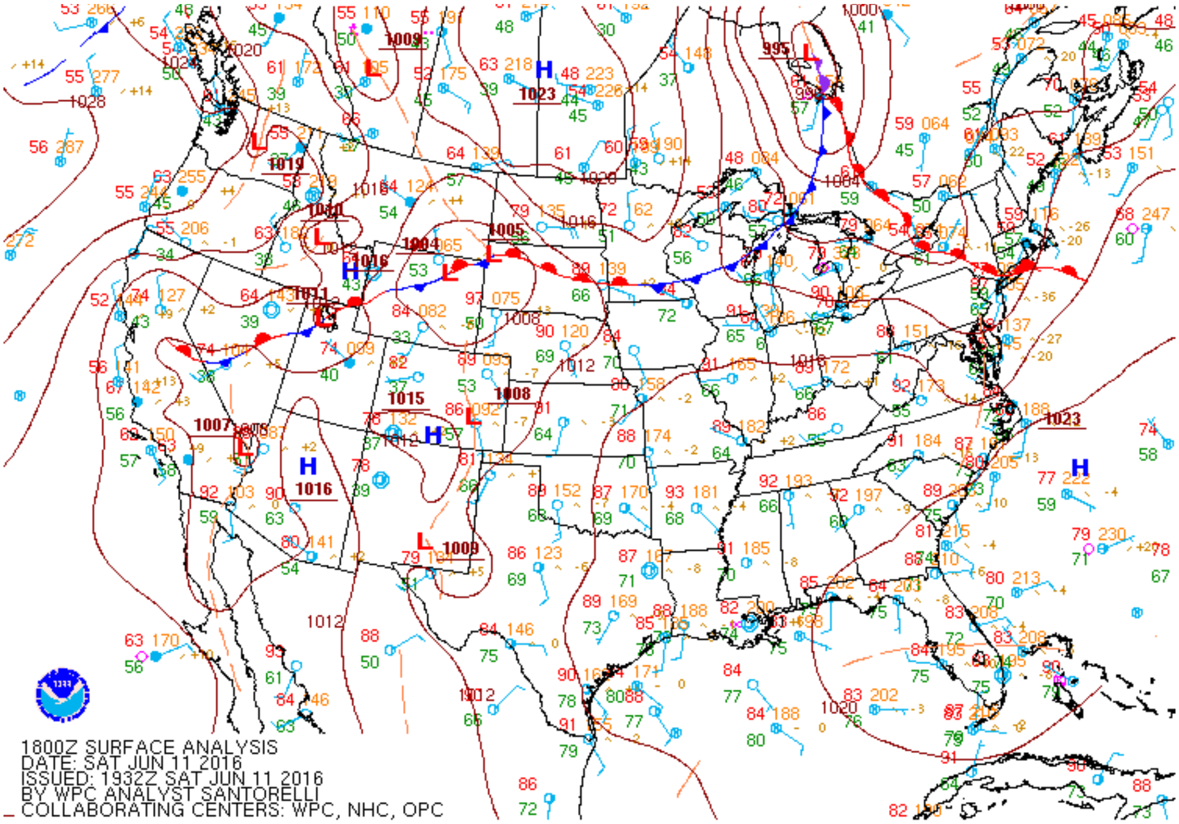


Figure 37. Surface analysis for 11-Jun-2016 18 UTC. Acquired from NWS Weather Prediction Center, http://www.wpc.ncep.noaa.gov/archives/web_pages/sfc/sfc_archive_maps.php?arc-date=06/11/2016&sel-map=2016061118&maptype=namussfc.

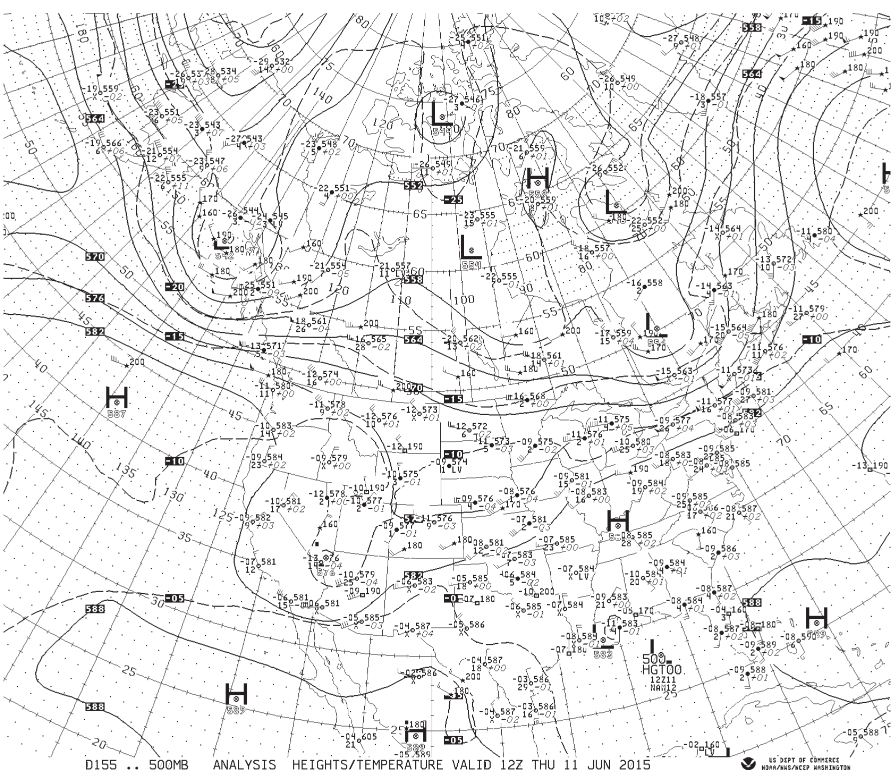


Figure 38. 500 hPa synoptic map for 11-Jun-2016 12 UTC. Acquired from Storm Research and Consulting, http://www.stormresearch.com/ncep/2015/2015_06/2015061112_500.tif.

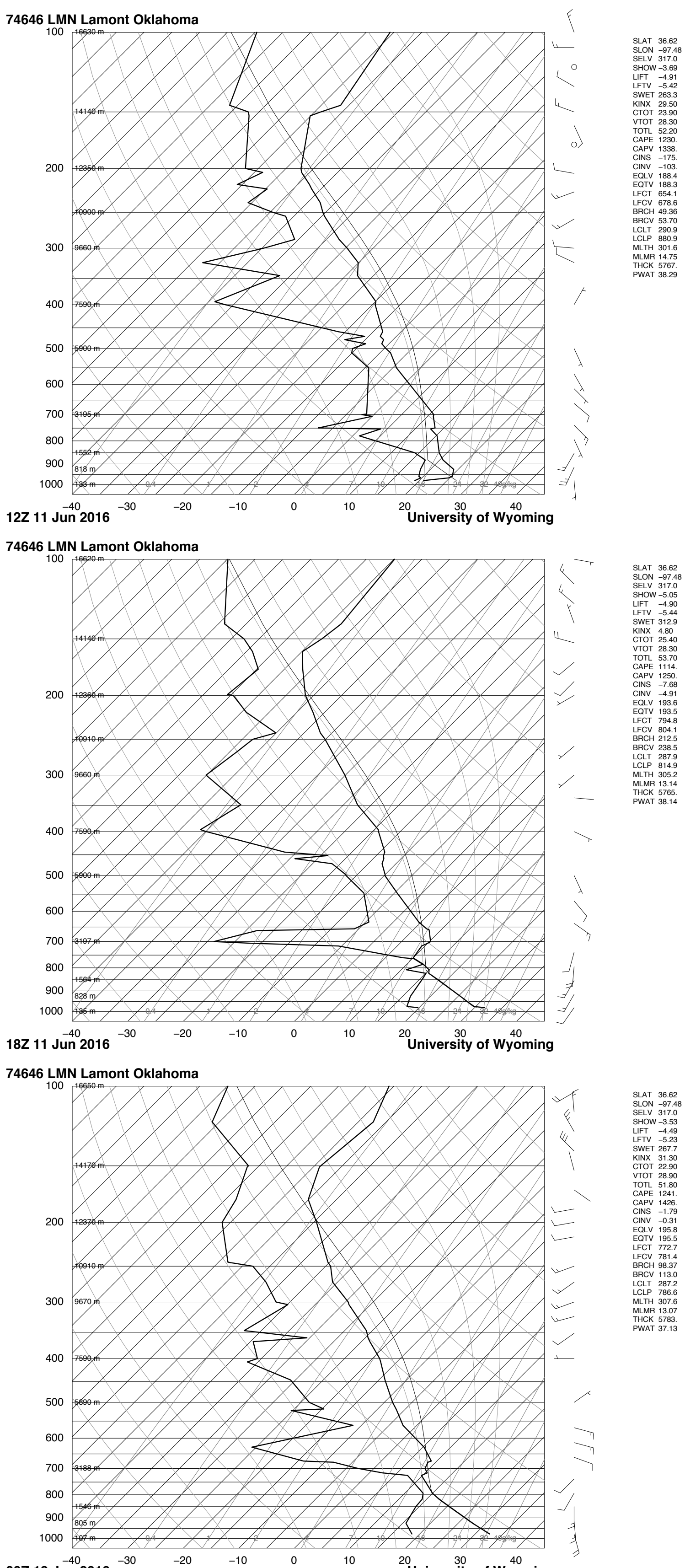


Figure 39. Skew-T log-P diagrams for Lamont, Oklahoma, for the 11-Jun-2016 LASSO case. Acquired from University of Wyoming, http://weather.uwyo.edu/upper-air/sounding.html.

Synoptic Conditions for 14-Jun-2016

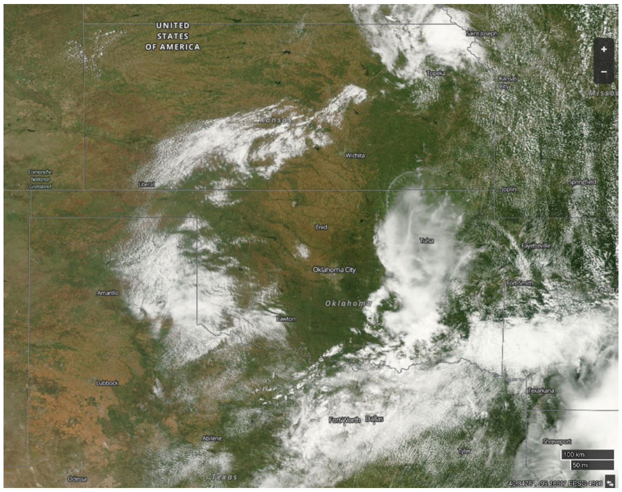


Figure 40. MODIS Terra corrected reflectance for 14-Jun-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2tRQrKH.

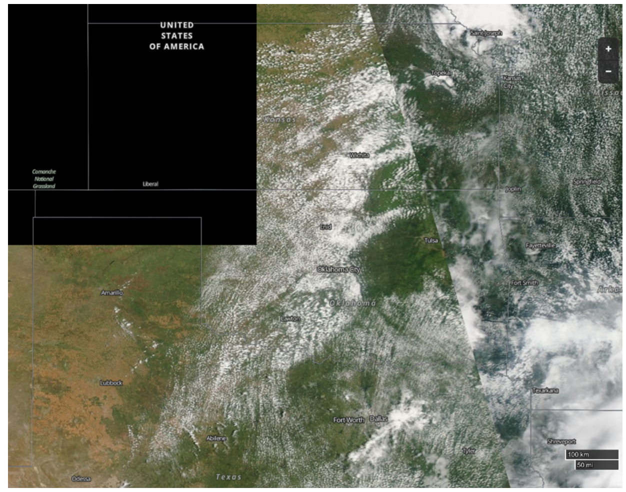


Figure 41. MODIS Aqua corrected reflectance for 14-Jun-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uTtAew.

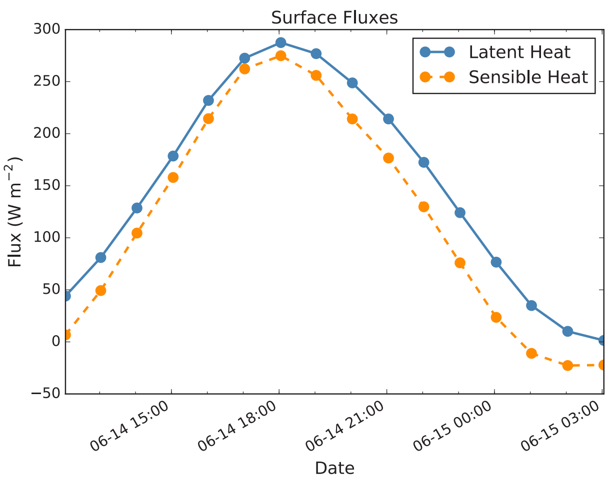


Figure 42. Surface sensible (orange, dashed) and latent (blue, solid) heat fluxes averaged for the SGP region. Hourly values taken from the variational analysis product. Date labels are UTC.

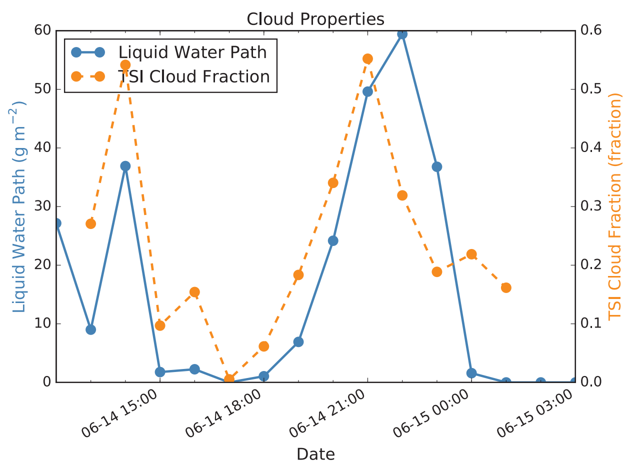


Figure 43. Cloud fraction (orange, dashed) from the TSI and cloud liquid water path (blue, solid) from a hybrid blending of MWRRet and AERIoe at the SGP Central Facility. Date labels are UTC.

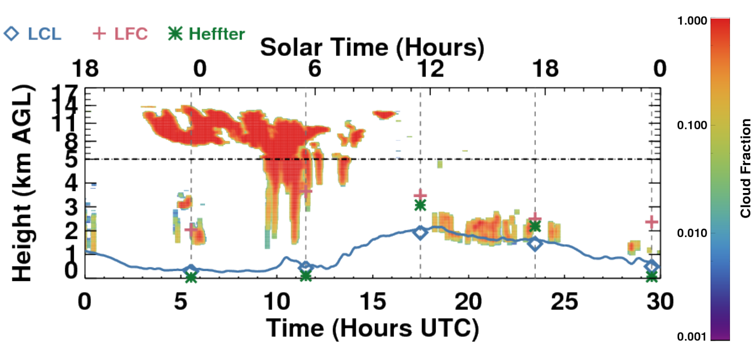


Figure 44. Cloud fraction profiles derived from the KAZR-ARSCL for 14-Jun-2016 at the Central Facility. Note the non-linear vertical axis that emphasizes the lower troposphere. Also indicated are the LCL, LFC, and PBL height based on the Heffter methodology, each of which are calculated from the SONDE product.

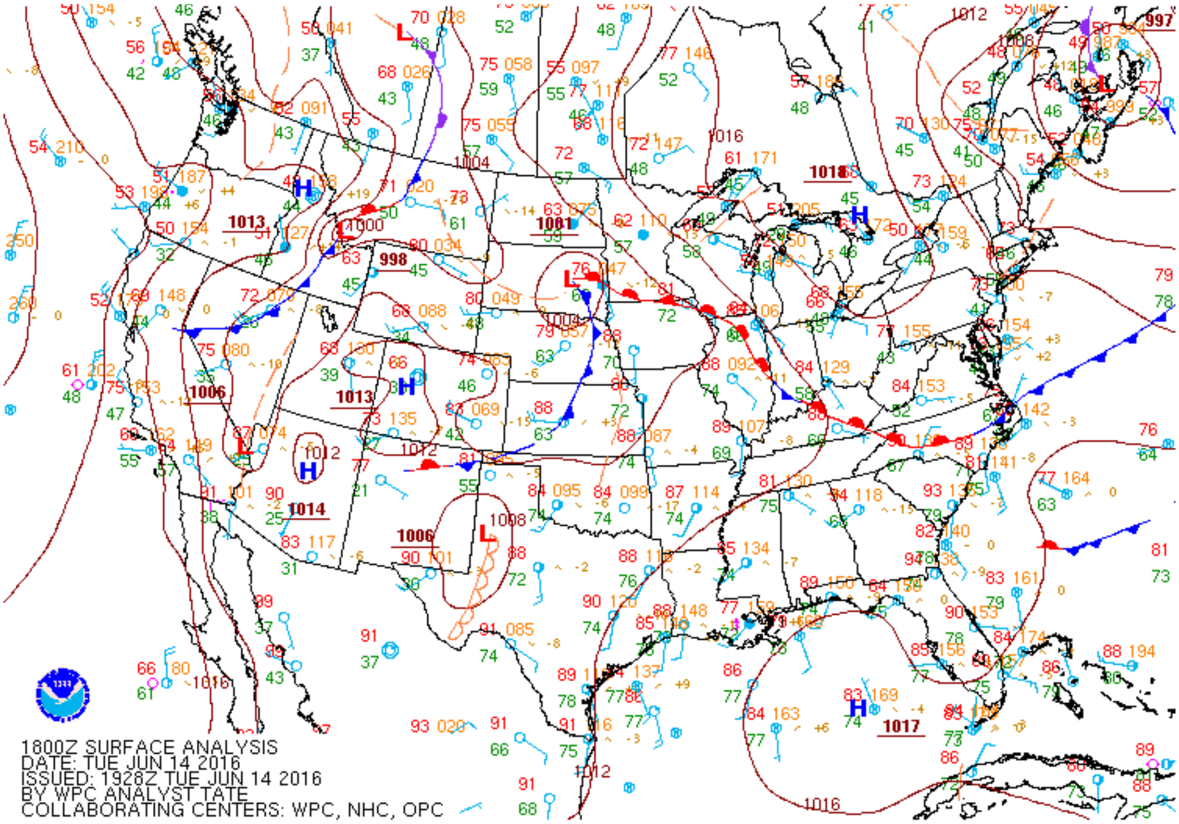


Figure 45. Surface analysis for 14-Jun-2016 18 UTC. Acquired from NWS Weather Prediction Center, http://www.wpc.ncep.noaa.gov/archives/web_pages/sfc/sfc_archive_maps.php?arc-date=06/14/2016&sel-map=2016061418&maptype=namussfc.

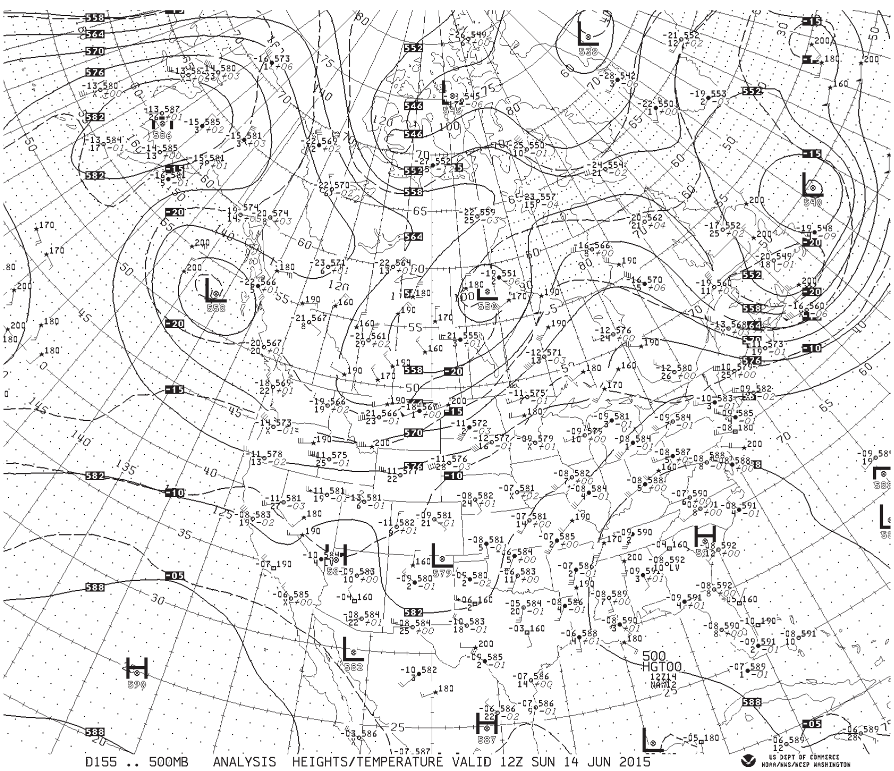


Figure 46. 500 hPa synoptic map for 14-Jun-2016 12 UTC. Acquired from Storm Research and Consulting, http://www.stormresearch.com/ncep/2015/2015_06/2015061412_500.tif.

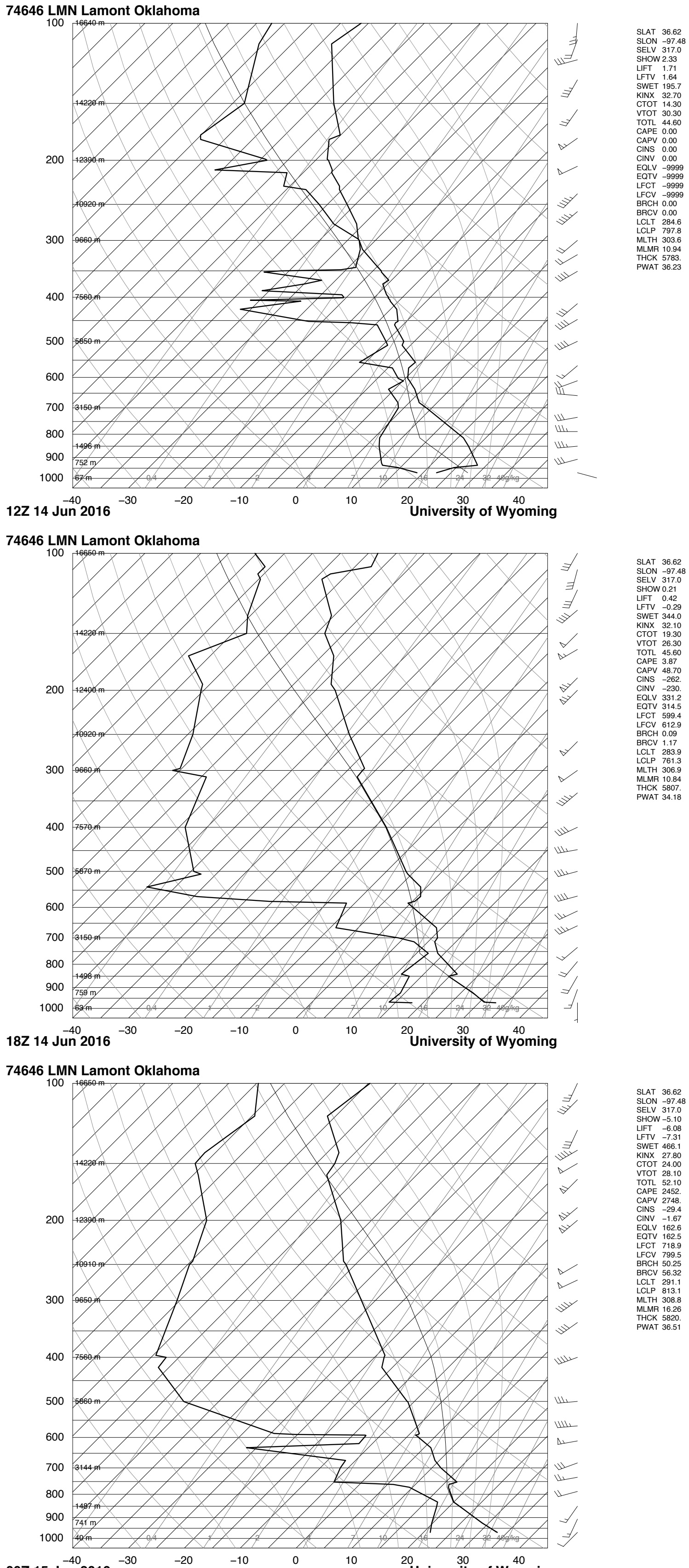


Figure 47. Skew-T log-P diagrams for Lamont, Oklahoma, for the 14-Jun-2016 LASSO case. Acquired from University of Wyoming, http://weather.uwyo.edu/upper-air/sounding.html.

Synoptic Conditions for 19-Jun-2016

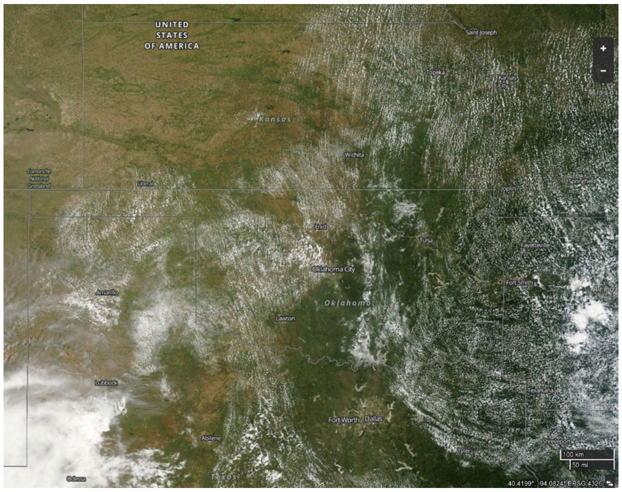


Figure 48. MODIS Terra corrected reflectance for 19-Jun-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2tRN3z8.

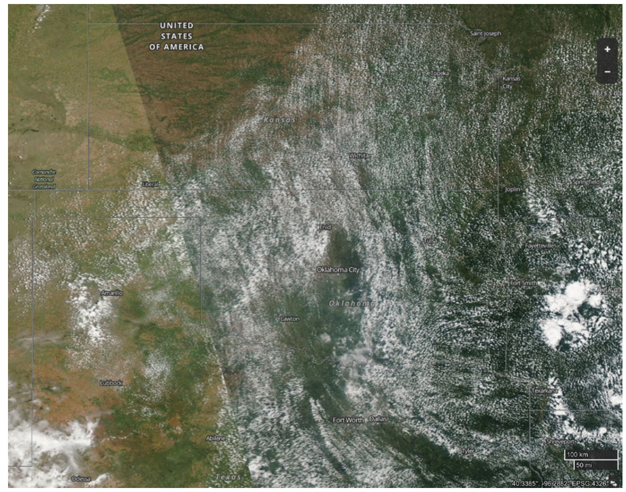


Figure 49. MODIS Aqua corrected reflectance for 19-Jun-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2tSjGga.

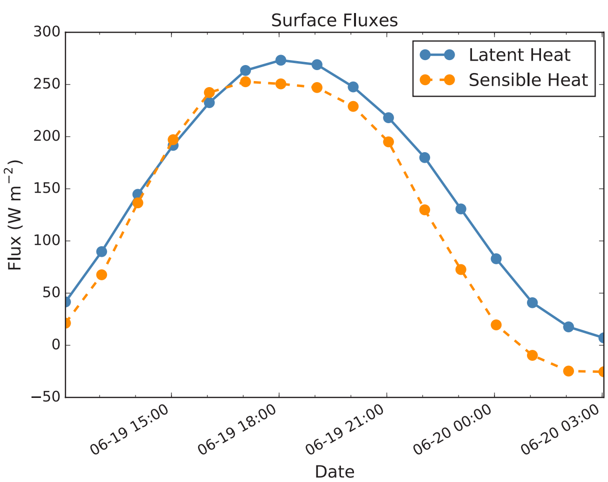


Figure 50. Surface sensible (orange, dashed) and latent (blue, solid) heat fluxes averaged for the SGP region. Hourly values taken from the variational analysis product. Date labels are UTC.

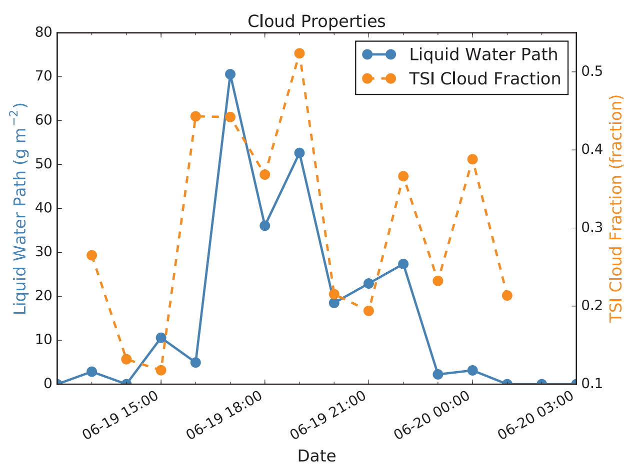


Figure 51. Cloud fraction (orange, dashed) from the TSI and cloud liquid water path (blue, solid) from a hybrid blending of MWRRet and AERIoe at the SGP Central Facility. Date labels are UTC.

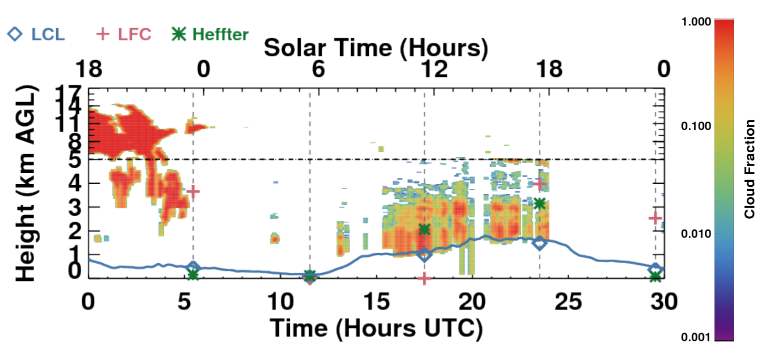


Figure 52. Cloud fraction profiles derived from the KAZR-ARSCL for 19-Jun-2016 at the Central Facility. Note the non-linear vertical axis that emphasizes the lower troposphere. Also indicated are the LCL, LFC, and PBL height based on the Heffter methodology, each of which are calculated from the SONDE product.

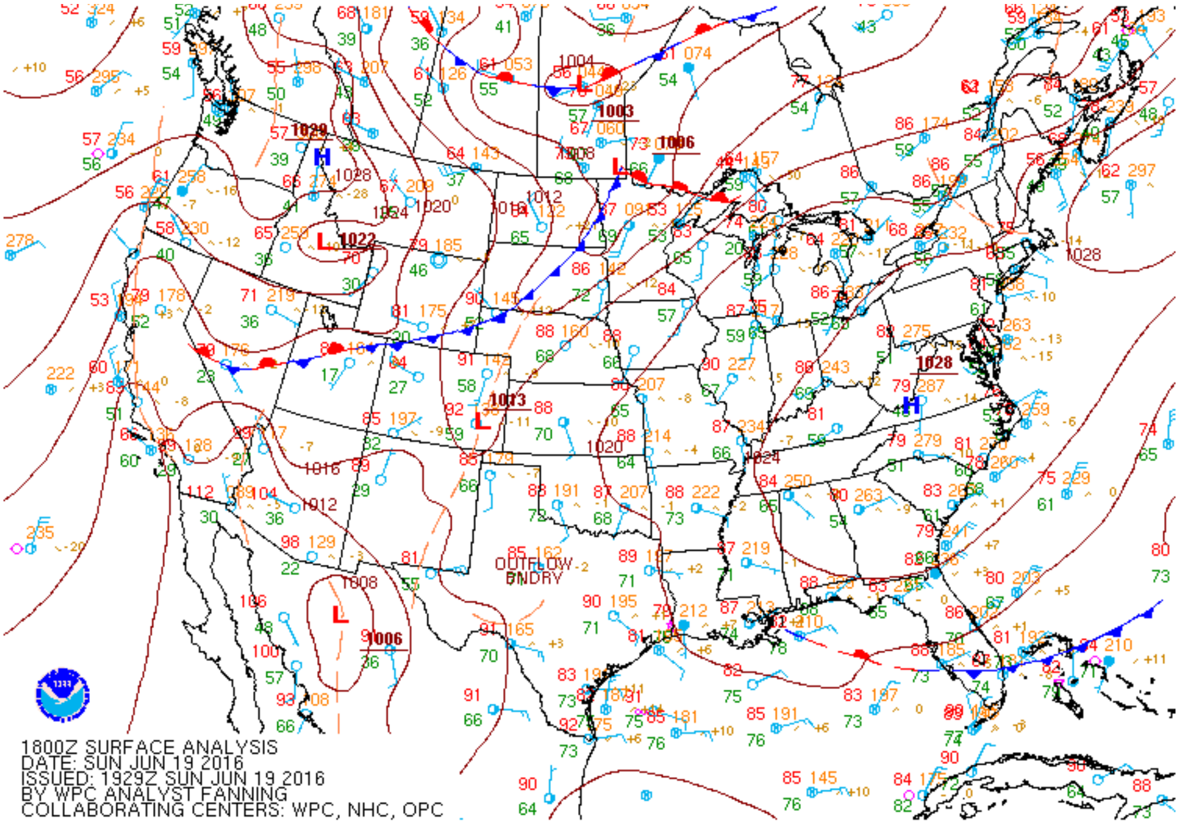


Figure 53. Surface analysis for 19-Jun-2016 18 UTC. Acquired from NWS Weather Prediction Center, http://www.wpc.ncep.noaa.gov/archives/web_pages/sfc/sfc_archive_maps.php?arc-date=06/19/2016&sel-map=2016061918&maptype=namussfc.

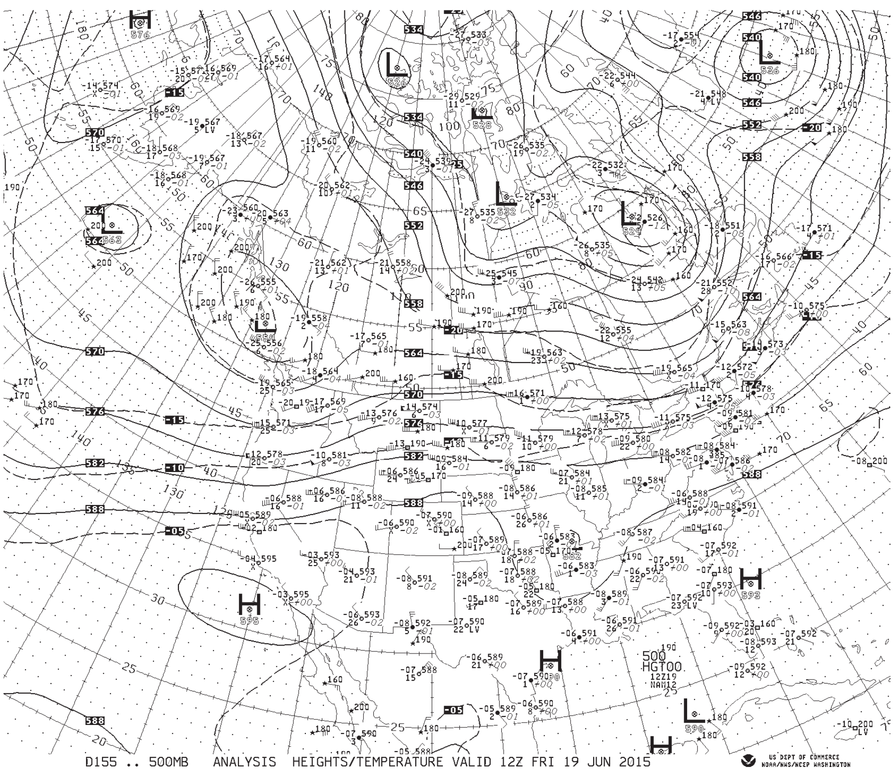


Figure 54. 500 hPa synoptic map for 19-Jun-2016 12 UTC. Acquired from Storm Research and Consulting, http://www.stormresearch.com/ncep/2015/2015_06/2015061912_500.tif.

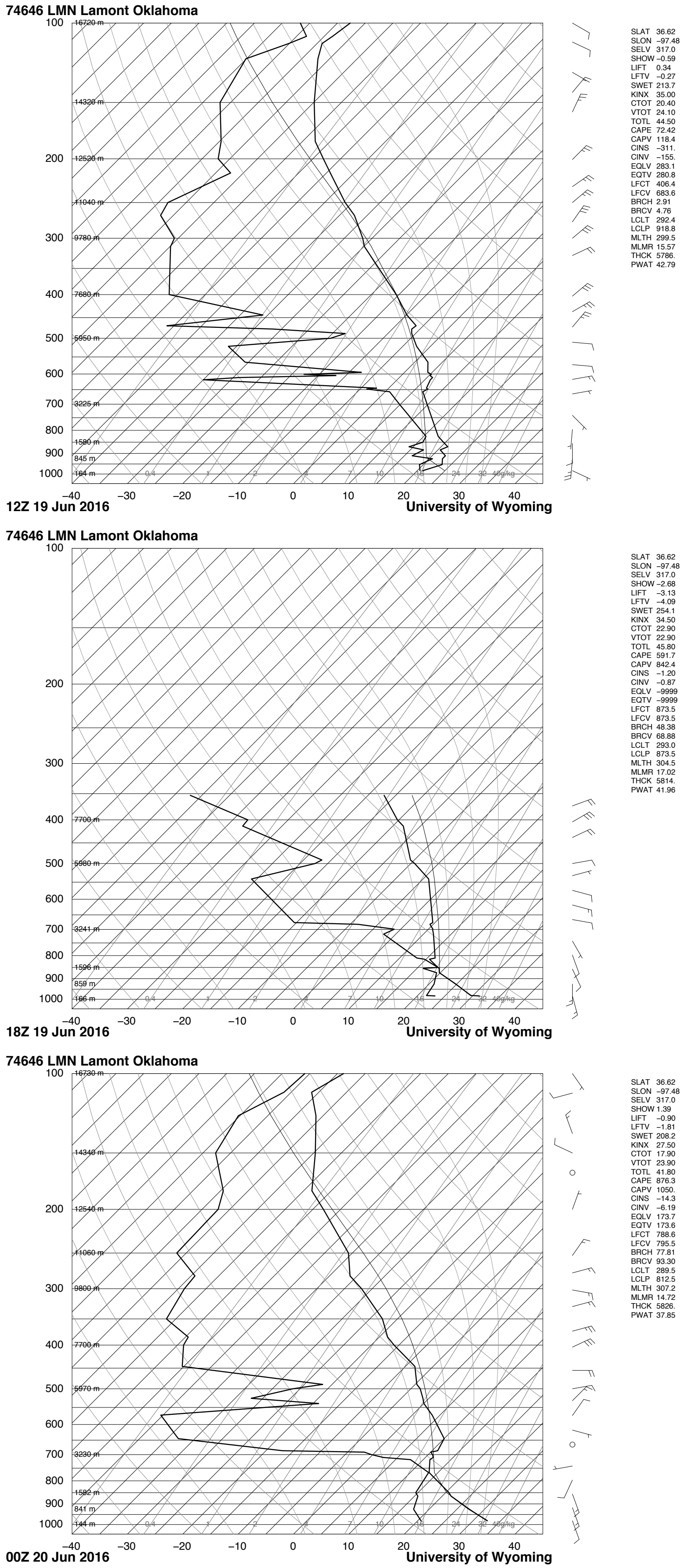


Figure 55. Skew-T log-P diagrams for Lamont, Oklahoma, for the 19-Jun-2016 LASSO case. Acquired from University of Wyoming, http://weather.uwyo.edu/upper-air/sounding.html.

Synoptic Conditions for 25-Jun-2016

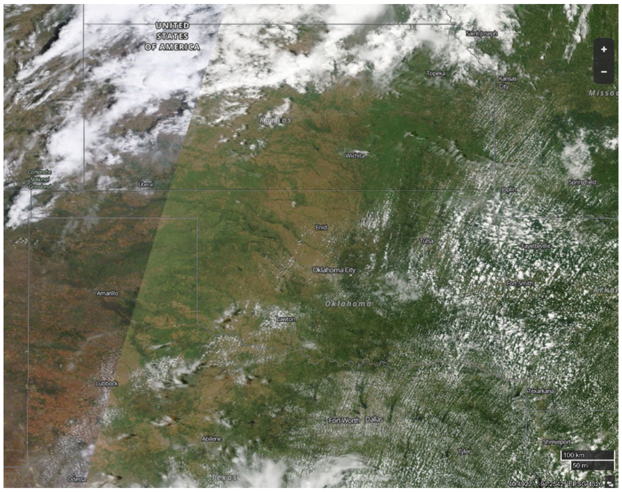


Figure 56. MODIS Terra corrected reflectance for 25-Jun-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uTYEuq.

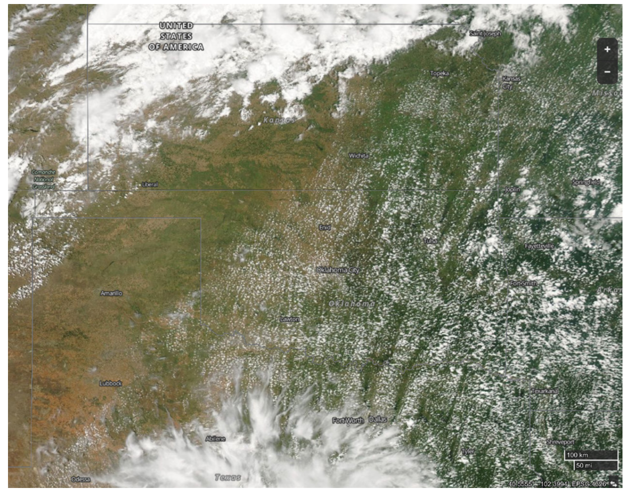


Figure 57. MODIS Aqua corrected reflectance for 25-Jun-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2tS35t3.

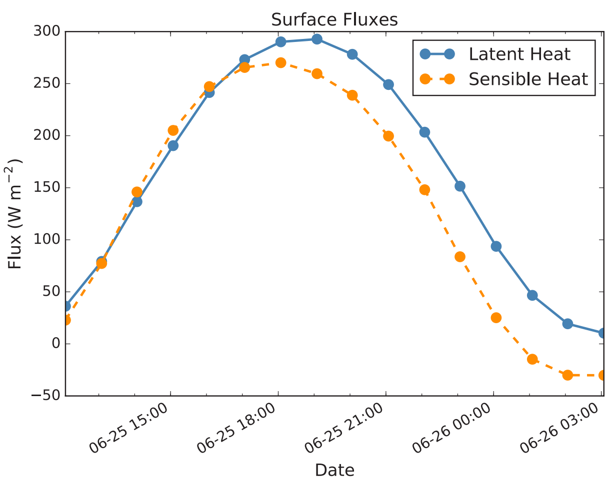


Figure 58. Surface sensible (orange, dashed) and latent (blue, solid) heat fluxes averaged for the SGP region. Hourly values taken from the variational analysis product. Date labels are UTC.

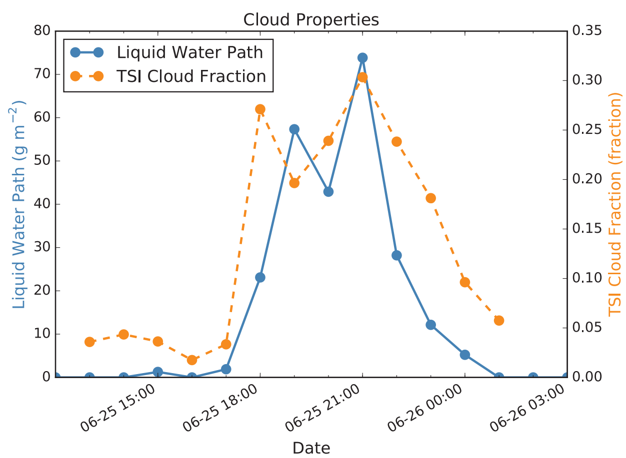


Figure 59. Cloud fraction (orange, dashed) from the TSI and cloud liquid water path (blue, solid) from a hybrid blending of MWRRet and AERIoe at the SGP Central Facility. Date labels are UTC.

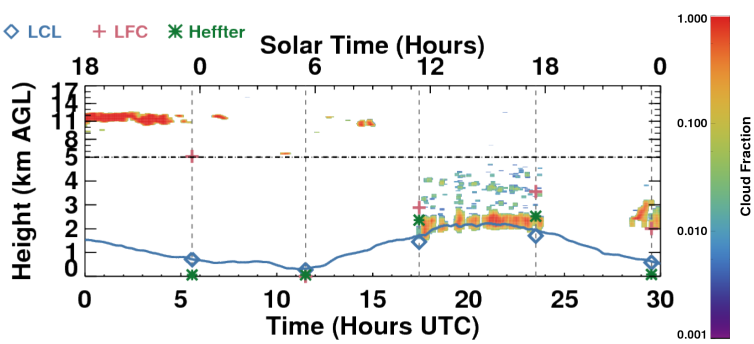


Figure 60. Cloud fraction profiles derived from the KAZR-ARSCL for 25-Jun-2016 at the Central Facility. Note the non-linear vertical axis that emphasizes the lower troposphere. Also indicated are the LCL, LFC, and PBL height based on the Heffter methodology, each of which are calculated from the SONDE product.

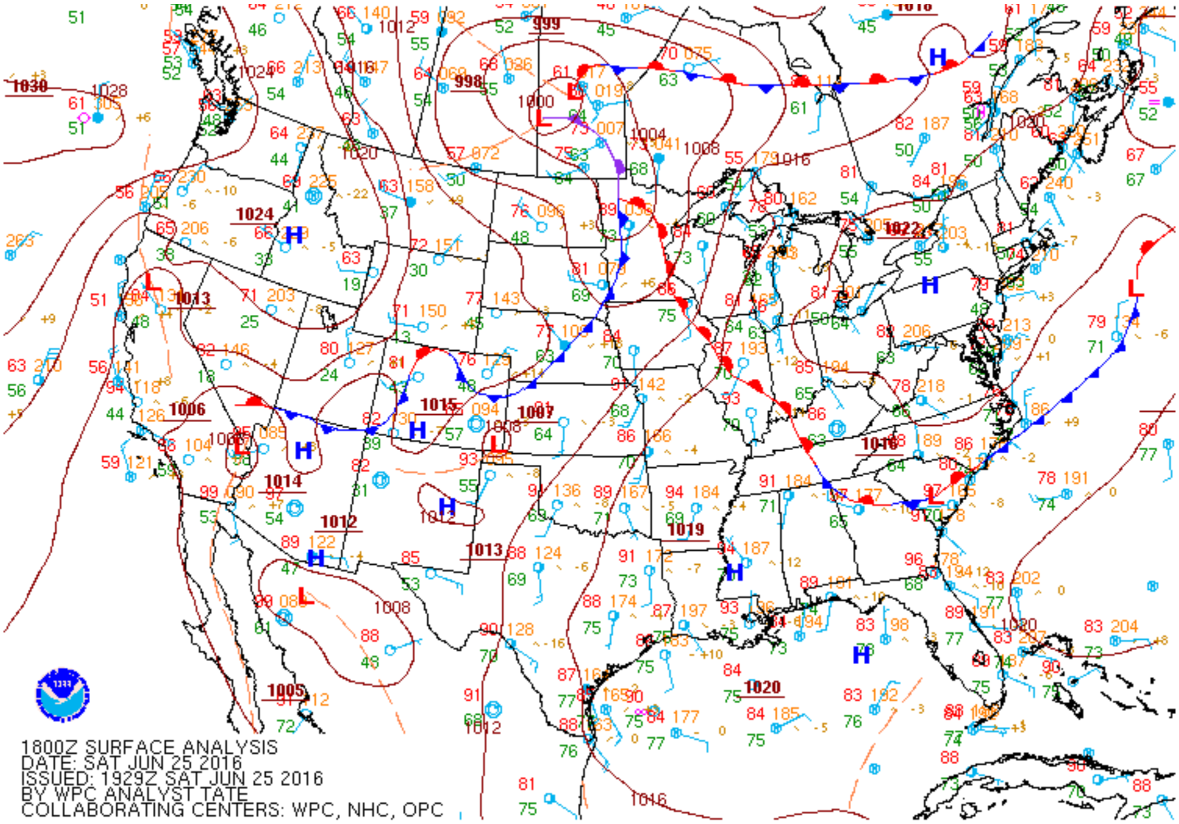


Figure 61. Surface analysis for 25-Jun-2016 18 UTC. Acquired from NWS Weather Prediction Center, http://www.wpc.ncep.noaa.gov/archives/web_pages/sfc/sfc_archive_maps.php?arc-date=06/25/2016&sel-map=2016062518&maptype=namussfc.

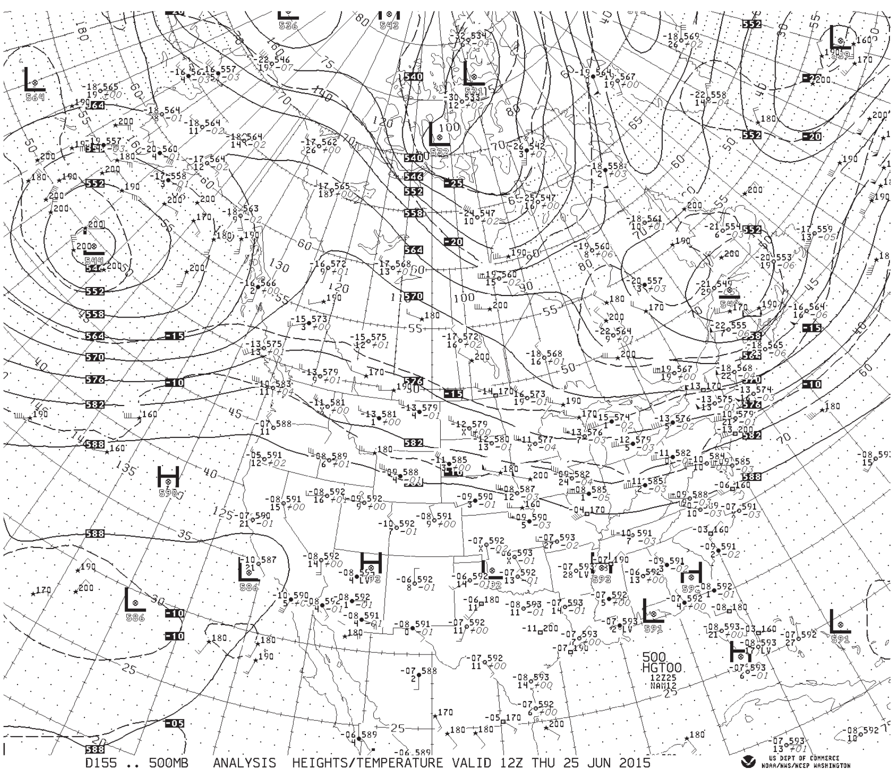


Figure 62. 500 hPa synoptic map for 25-Jun-2016 12 UTC. Acquired from Storm Research and Consulting, http://www.stormresearch.com/ncep/2015/2015_06/2015062512_500.tif.

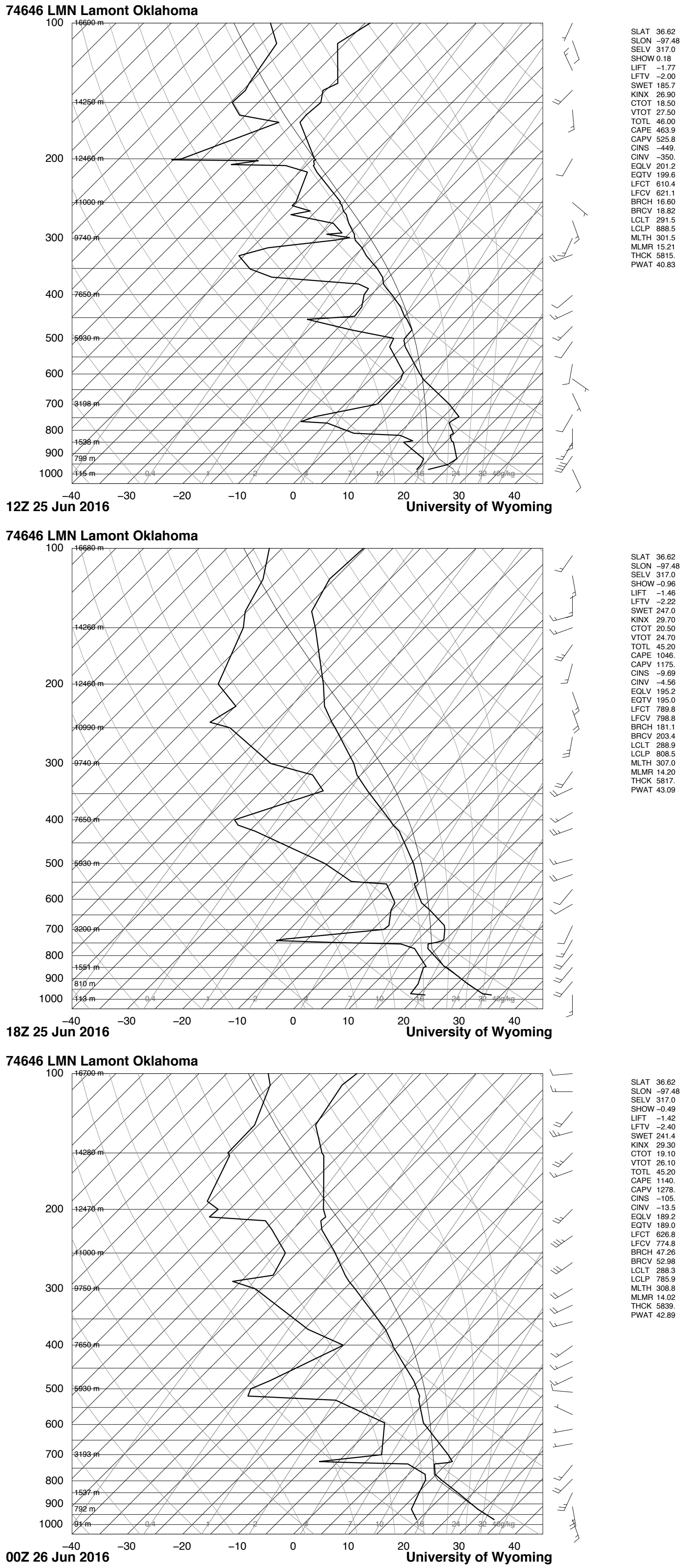


Figure 63. Skew-T log-P diagrams for Lamont, Oklahoma, for the 25-Jun-2016 LASSO case. Acquired from University of Wyoming, http://weather.uwyo.edu/upper-air/sounding.html.

Synoptic Conditions for 16-Jul-2016

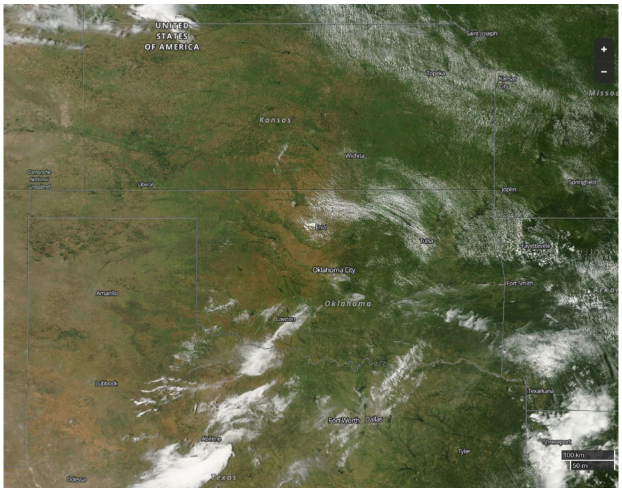


Figure 64. MODIS Terra corrected reflectance for 16-Jul-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uTP9vy.

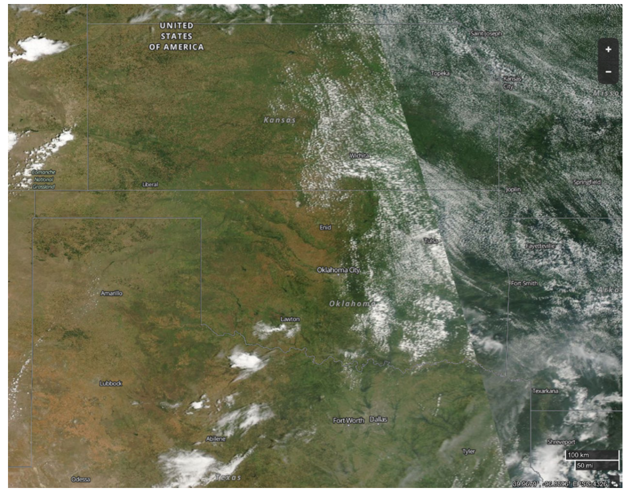


Figure 65. MODIS Aqua corrected reflectance for 16-Jul-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uTEoZR.

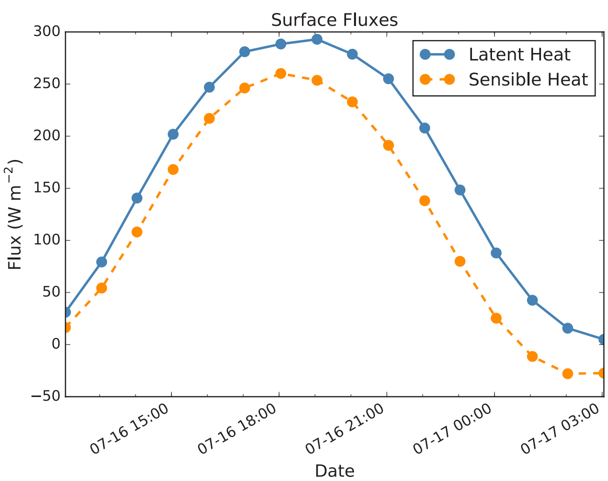


Figure 66. Surface sensible (orange, dashed) and latent (blue, solid) heat fluxes averaged for the SGP region. Hourly values taken from the variational analysis product. Date labels are UTC.

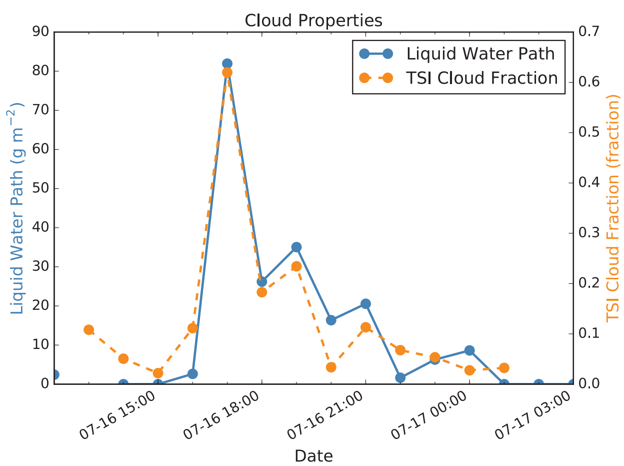


Figure 67. Cloud fraction (orange, dashed) from the TSI and cloud liquid water path (blue, solid) from a hybrid blending of MWRRet and AERIoe at the SGP Central Facility. Date labels are UTC.

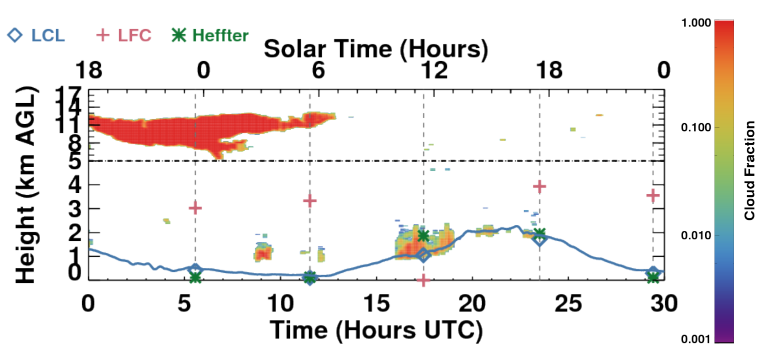


Figure 68. Cloud fraction profiles derived from the KAZR-ARSCL for 16-Jul-2016 at the Central Facility. Note the non-linear vertical axis that emphasizes the lower troposphere. Also indicated are the LCL, LFC, and PBL height based on the Heffter methodology, each of which are calculated from the SONDE product.

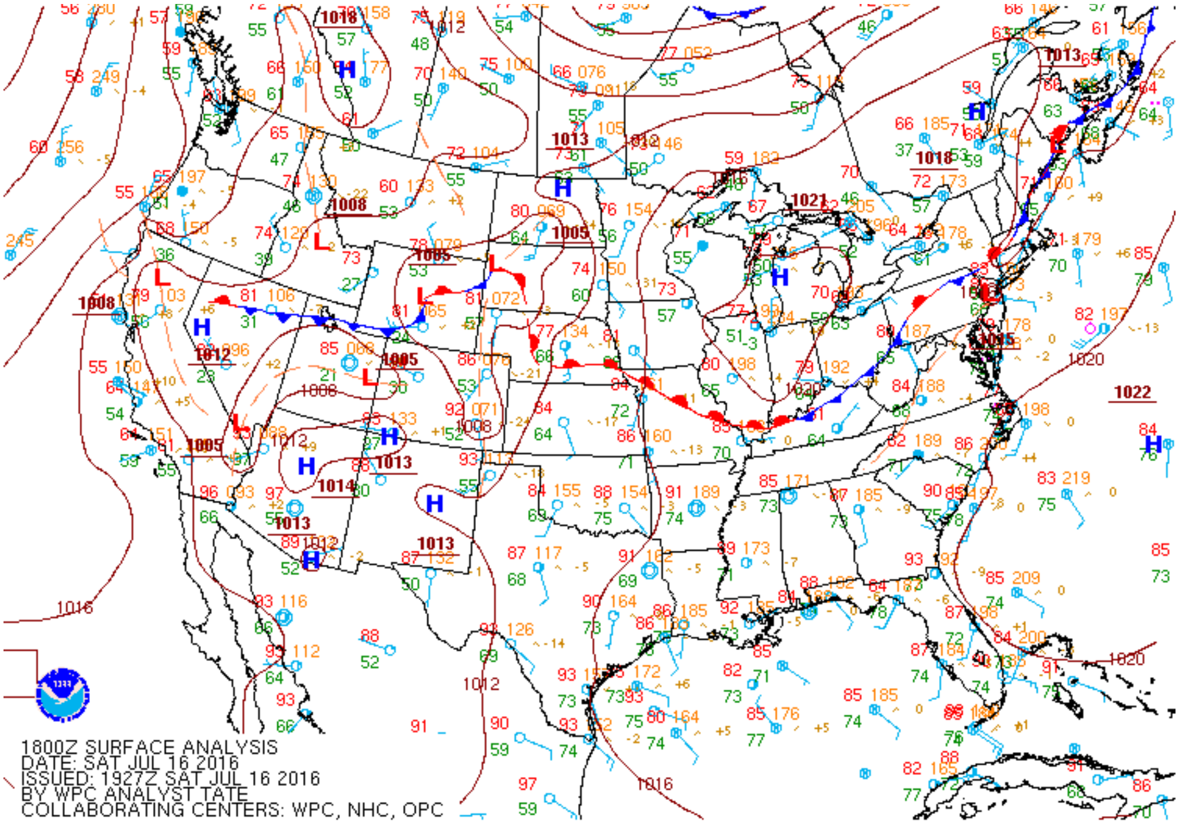


Figure 69. Surface analysis for 16-Jul-2016 18 UTC. Acquired from NWS Weather Prediction Center, http://www.wpc.ncep.noaa.gov/archives/web_pages/sfc/sfc_archive_maps.php?arc-date=07/16/2016&sel-map=2016071618&maptype=namussfc.

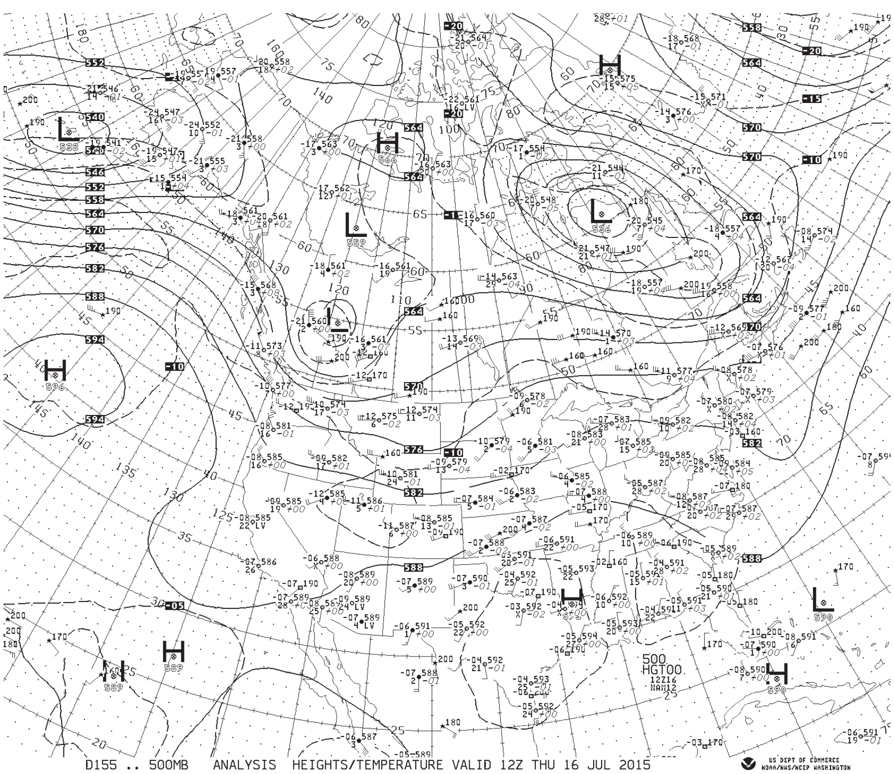


Figure 70. 500 hPa synoptic map for 16-Jul-2016 12 UTC. Acquired from Storm Research and Consulting, http://www.stormresearch.com/ncep/2015/2015_07/2015071612_500.tif.

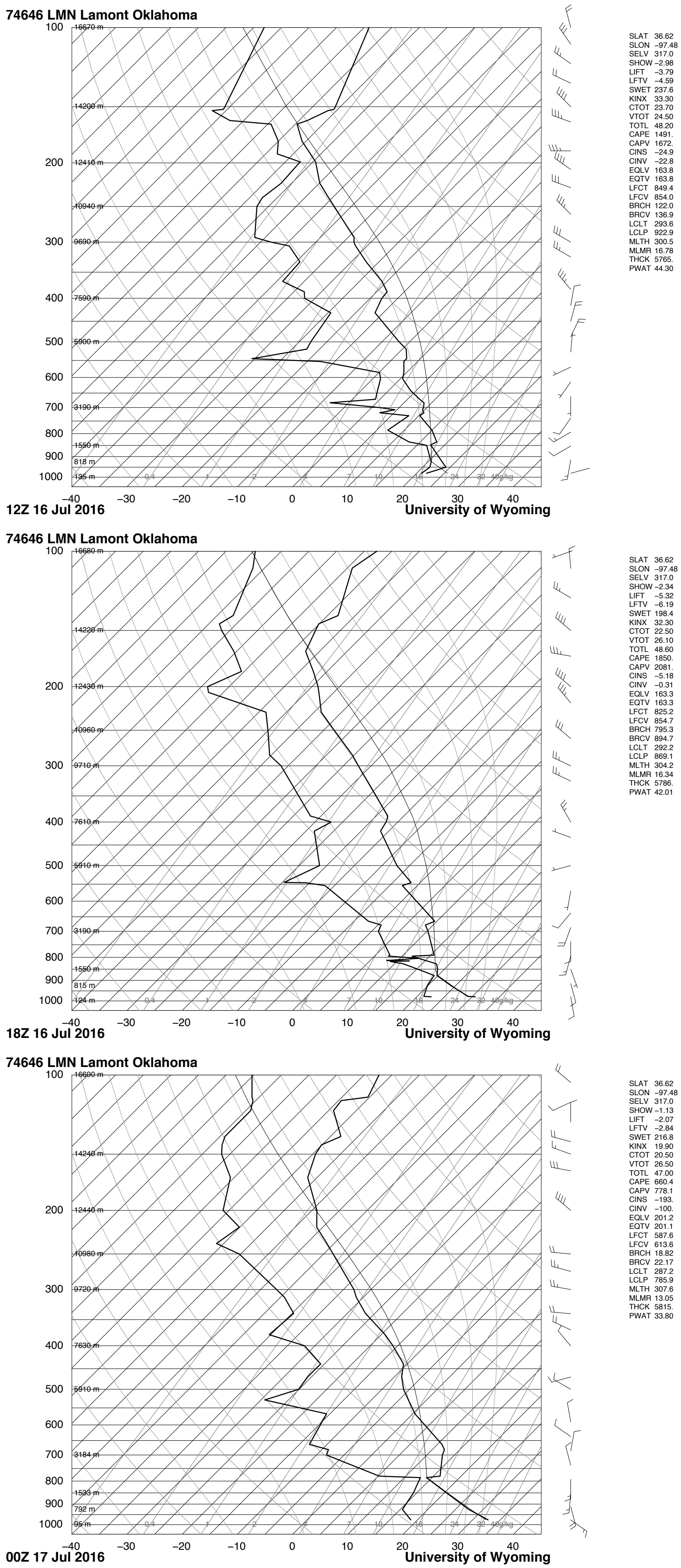


Figure 71. Skew-T log-P diagrams for Lamont, Oklahoma, for the 16-Jul-2016 LASSO case. Acquired from University of Wyoming, http://weather.uwyo.edu/upper-air/sounding.html.

Synoptic Conditions for 19-Jul-2016

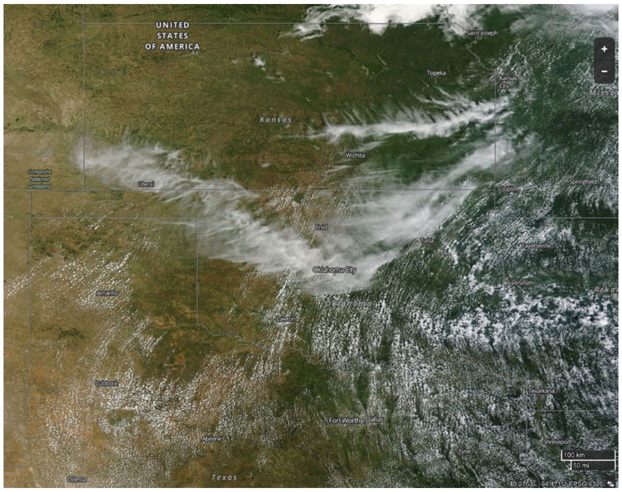


Figure 72. MODIS Terra corrected reflectance for 19-Jul-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2tRPIsF.

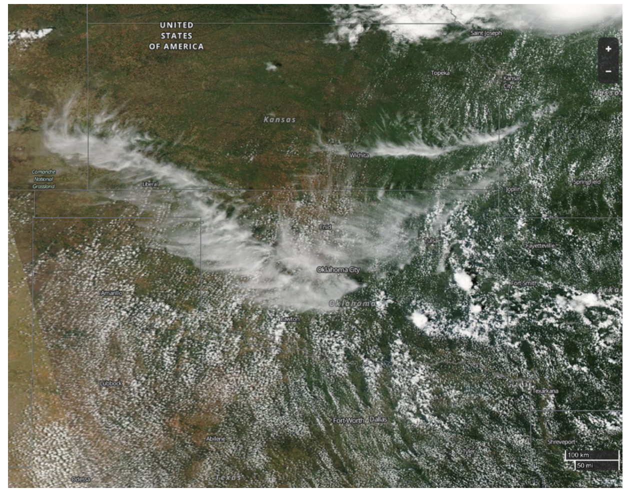


Figure 73. MODIS Aqua corrected reflectance for 19-Jul-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uTI1PB.

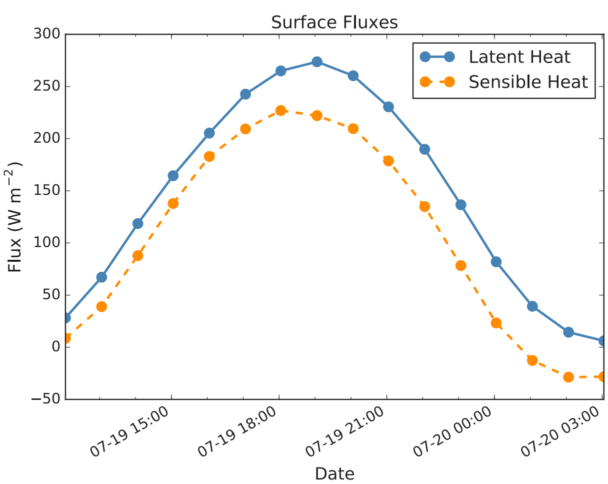


Figure 74. Surface sensible (orange, dashed) and latent (blue, solid) heat fluxes averaged for the SGP region. Hourly values taken from the variational analysis product. Date labels are UTC.

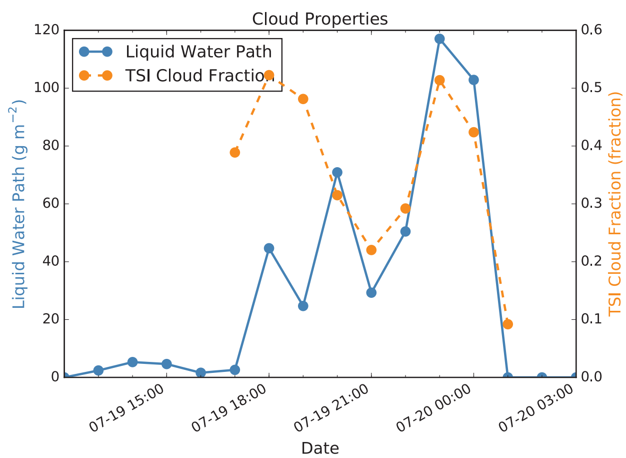


Figure 75. Cloud fraction (orange, dashed) from the TSI and cloud liquid water path (blue, solid) from a hybrid blending of MWRRet and AERIoe at the SGP Central Facility. Date labels are UTC.

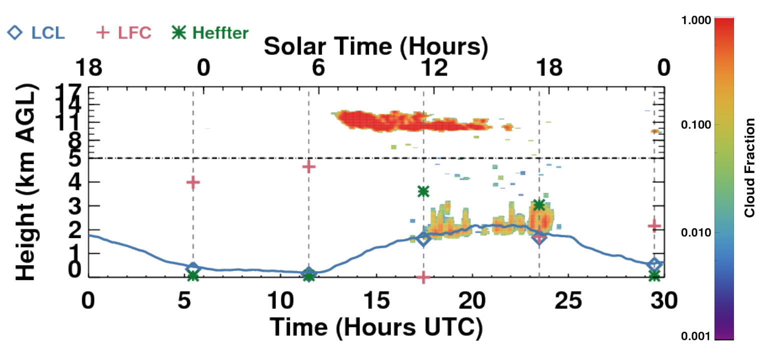


Figure 76. Cloud fraction profiles derived from the KAZR-ARSCL for 19-Jul-2016 at the Central Facility. Note the non-linear vertical axis that emphasizes the lower troposphere. Also indicated are the LCL, LFC, and PBL height based on the Heffter methodology, each of which are calculated from the SONDE product.

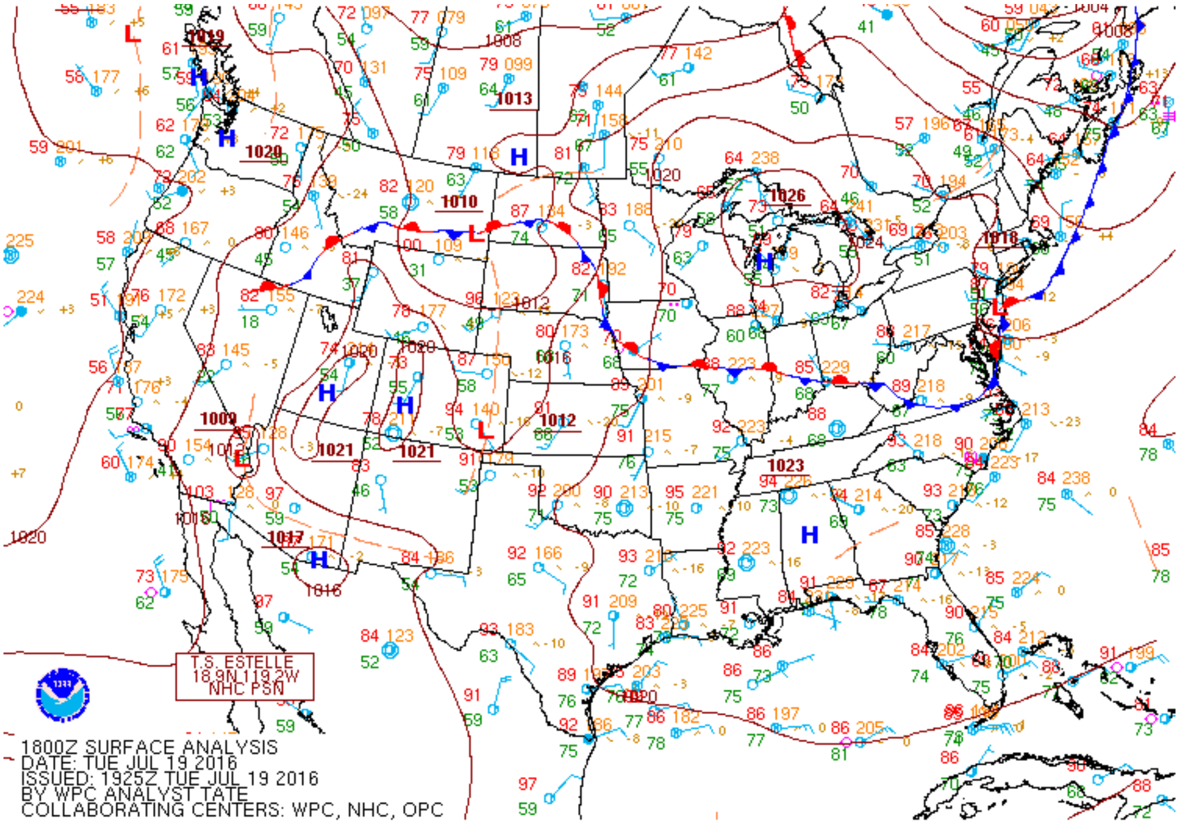


Figure 77. Surface analysis for 19-Jul-2016 18 UTC. Acquired from NWS Weather Prediction Center, http://www.wpc.ncep.noaa.gov/archives/web_pages/sfc/sfc_archive_maps.php?arc-date=07/19/2016&sel-map=2016071918&maptype=namussfc.

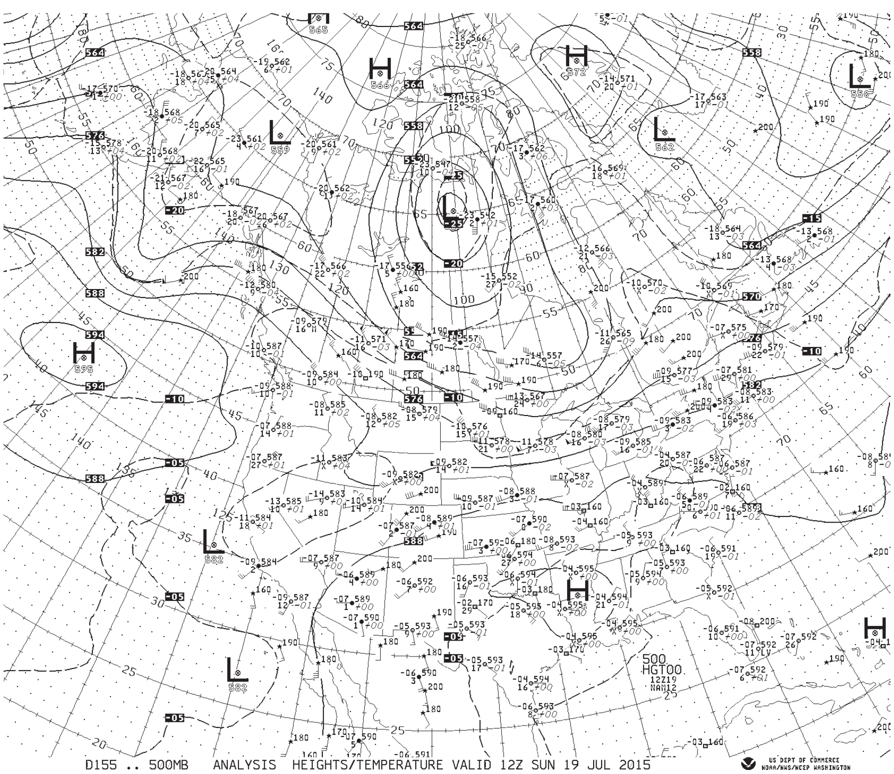


Figure 78. 500 hPa synoptic map for 19-Jul-2016 12 UTC. Acquired from Storm Research and Consulting, http://www.stormresearch.com/ncep/2015/2015_07/2015071912_500.tif.

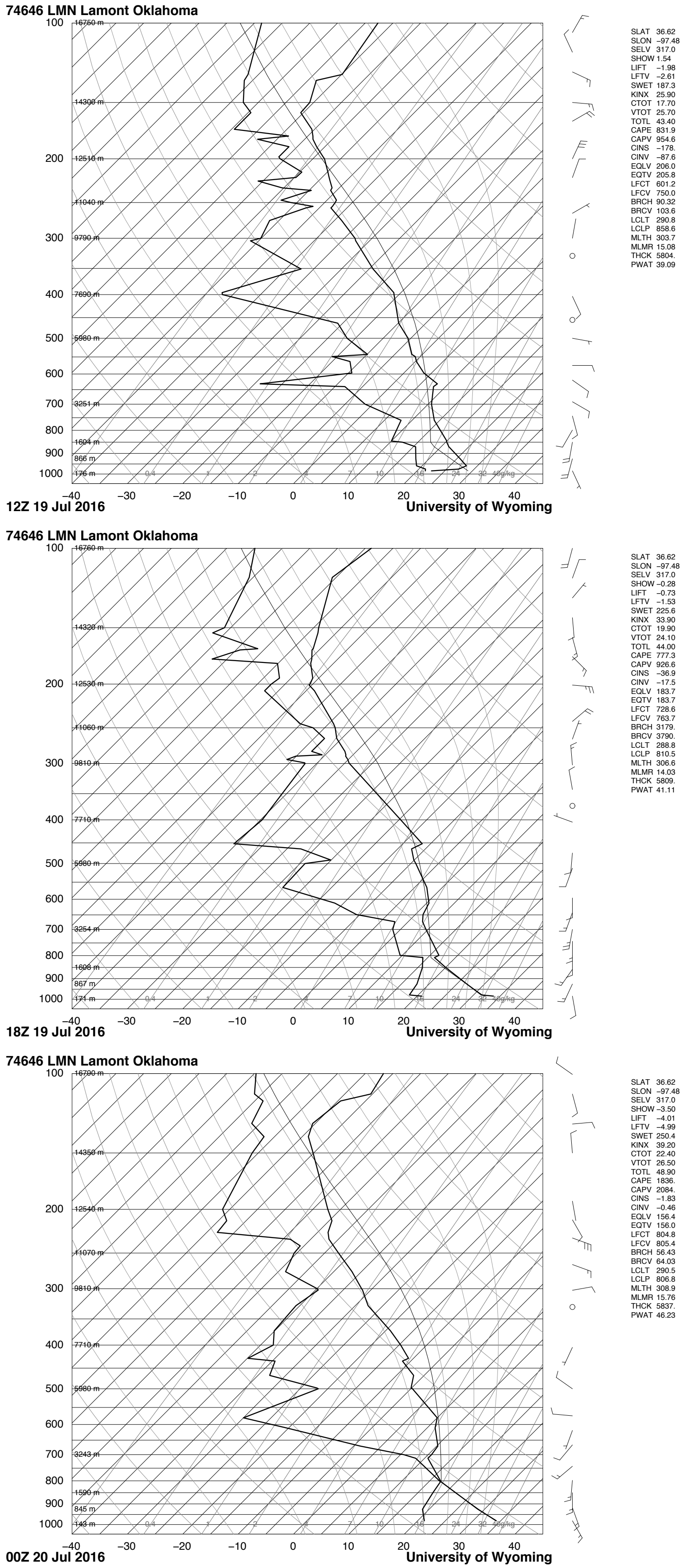


Figure 79. Skew-T log-P diagrams for Lamont, Oklahoma, for the 19-Jul-2016 LASSO case. Acquired from University of Wyoming, http://weather.uwyo.edu/upper-air/sounding.html.

Synoptic Conditions for 20-Jul-2016

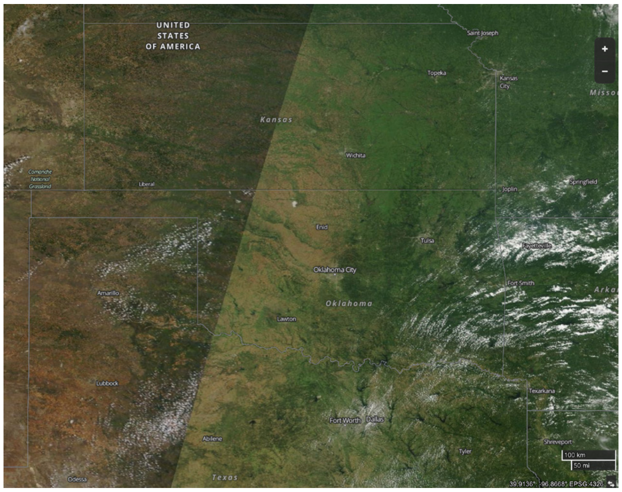


Figure 80. MODIS Terra corrected reflectance for 20-Jul-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2tROEoA.

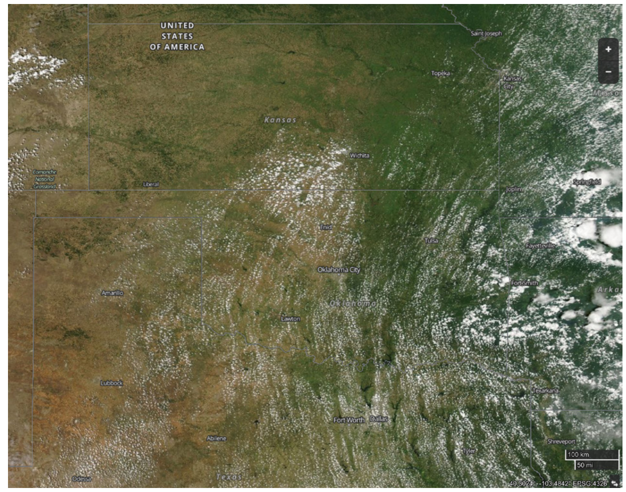


Figure 81. MODIS Aqua corrected reflectance for 20-Jul-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2tS3Pyk.

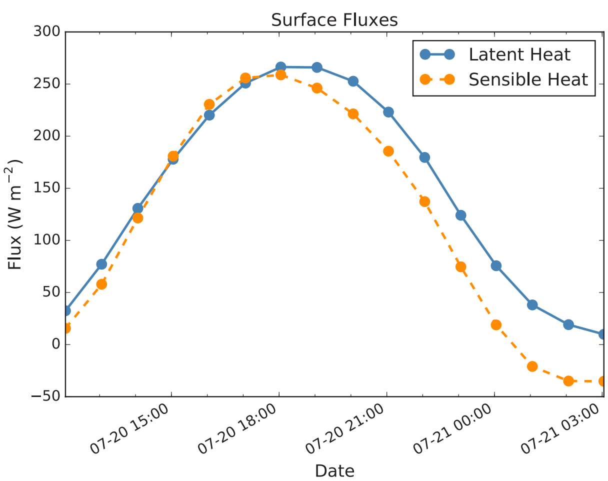


Figure 82. Surface sensible (orange, dashed) and latent (blue, solid) heat fluxes averaged for the SGP region. Hourly values taken from the variational analysis product. Date labels are UTC.

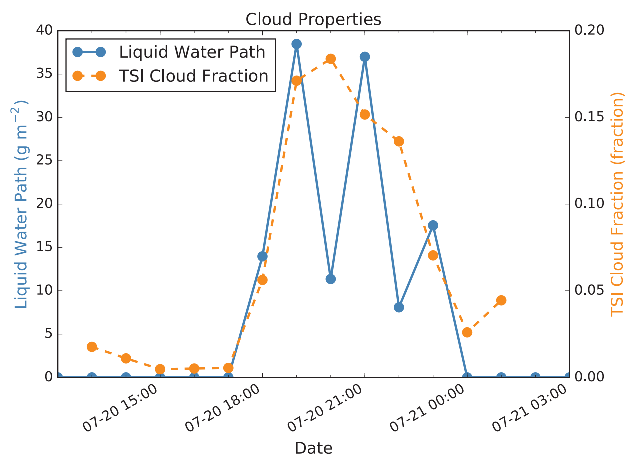


Figure 83. Cloud fraction (orange, dashed) from the TSI and cloud liquid water path (blue, solid) from a hybrid blending of MWRRet and AERIoe at the SGP Central Facility. Date labels are UTC.

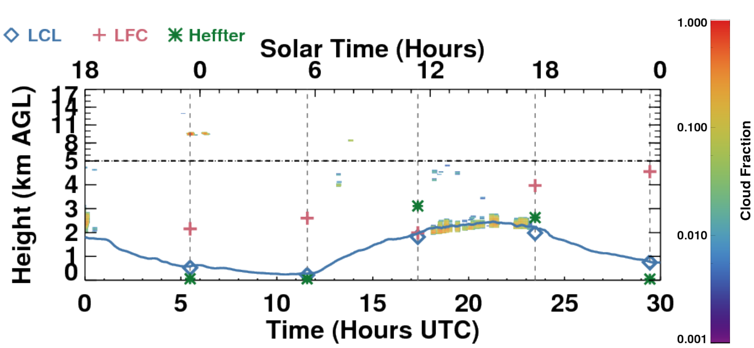


Figure 84. Cloud fraction profiles derived from the KAZR-ARSCL for 20-Jul-2016 at the Central Facility. Note the non-linear vertical axis that emphasizes the lower troposphere. Also indicated are the LCL, LFC, and PBL height based on the Heffter methodology, each of which are calculated from the SONDE product.

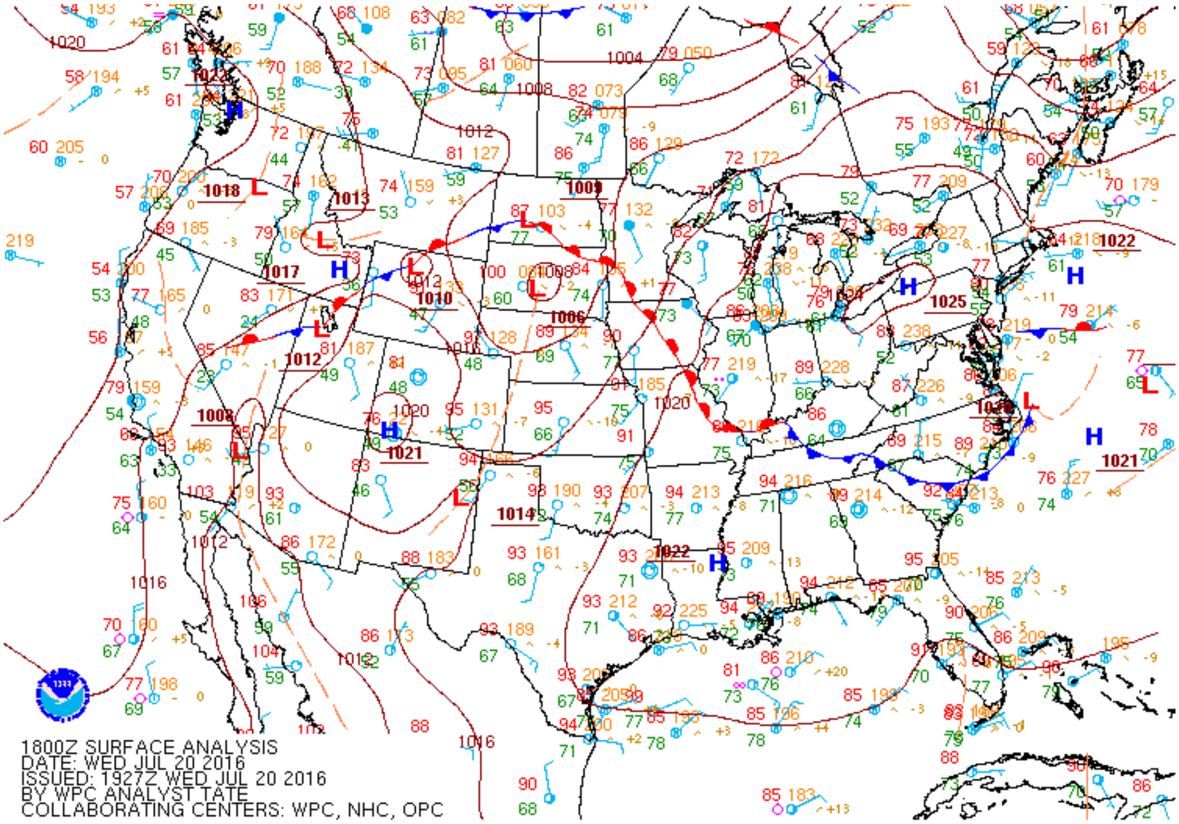


Figure 85. Surface analysis for 20-Jul-2016 18 UTC. Acquired from NWS Weather Prediction Center, http://www.wpc.ncep.noaa.gov/archives/web_pages/sfc/sfc_archive_maps.php?arc-date=07/20/2016&sel-map=2016072018&maptype=namussfc.

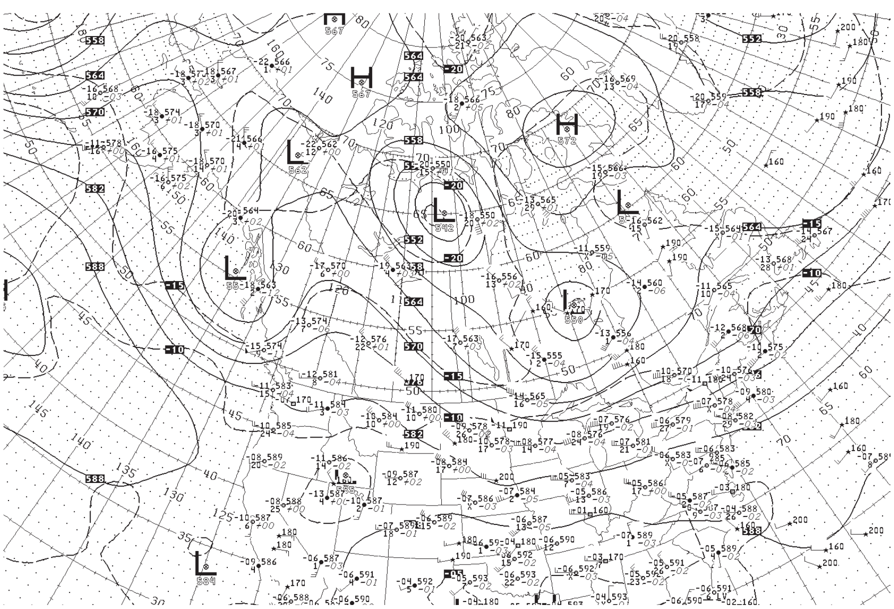


Figure 86. 500 hPa synoptic map for 20-Jul-2016 12 UTC. Acquired from Storm Research and Consulting, http://www.stormresearch.com/ncep/2015/2015_07/2015072012_500.tif.

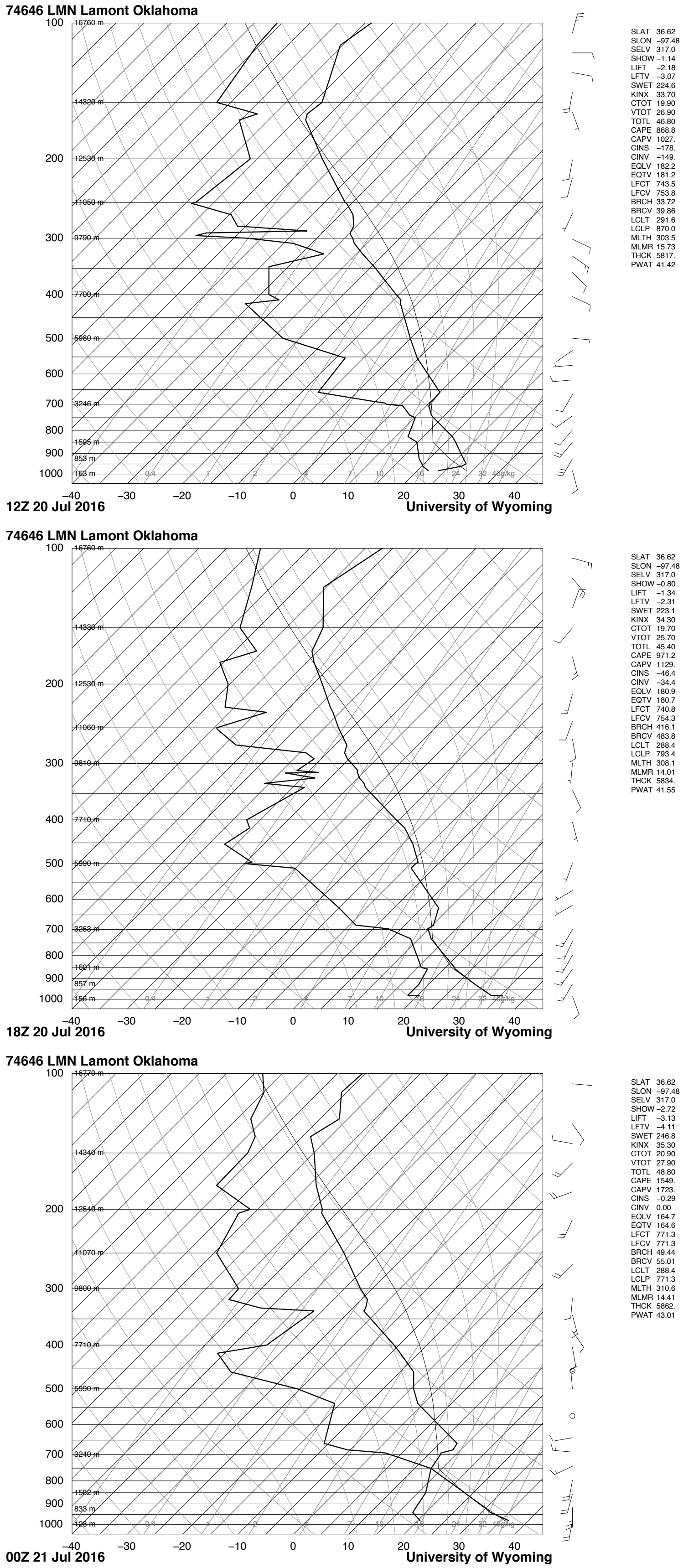


Figure 87. Skew-T log-P diagrams for Lamont, Oklahoma, for the 20-Jul-2016 LASSO case. Acquired from University of Wyoming, http://weather.uwyo.edu/upper-air/sounding.html.

Synoptic Conditions for 18-Aug-2016

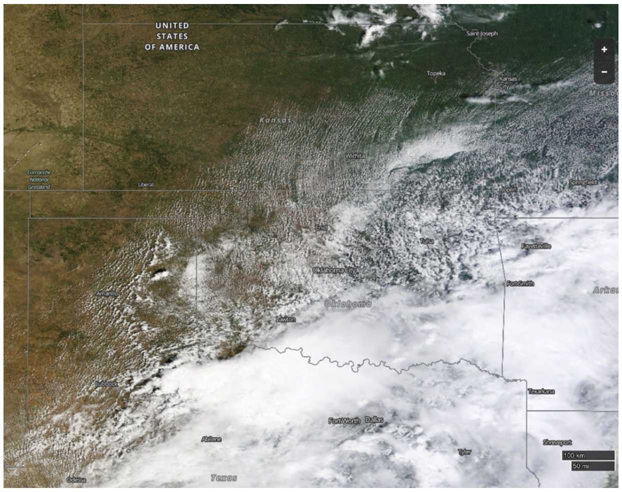


Figure 88. MODIS Terra corrected reflectance for 18-Aug-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2tRnmPo.

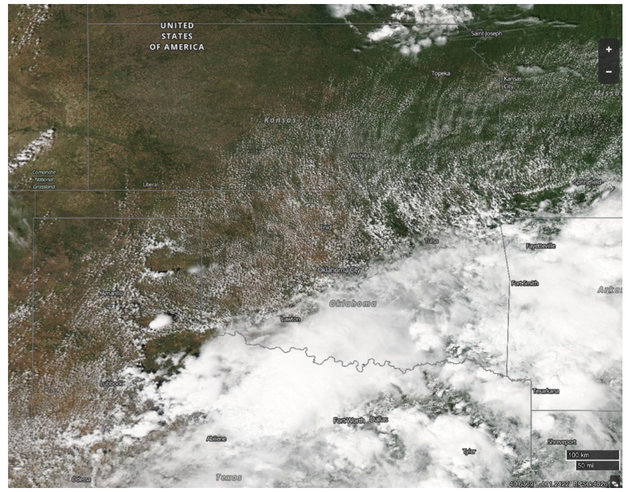


Figure 89. MODIS Aqua corrected reflectance for 18-Aug-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2tRW5MJ.

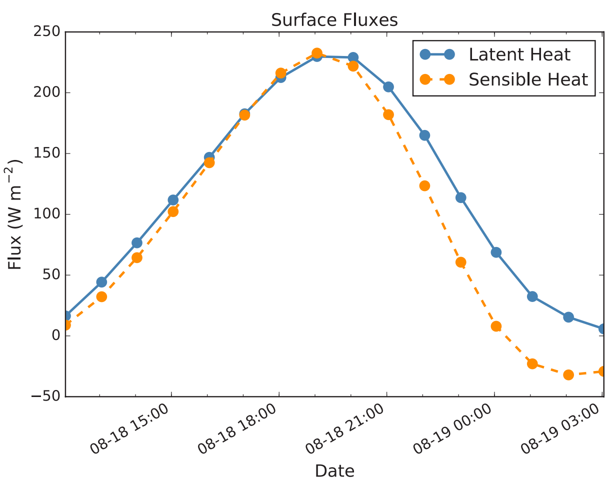


Figure 90. Surface sensible (orange, dashed) and latent (blue, solid) heat fluxes averaged for the SGP region. Hourly values taken from the variational analysis product. Date labels are UTC.

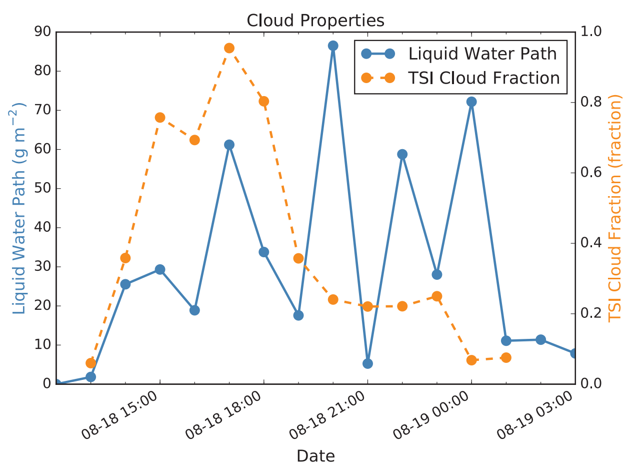


Figure 91. Cloud fraction (orange, dashed) from the TSI and cloud liquid water path (blue, solid) from a hybrid blending of MWRRet and AERIoe at the SGP Central Facility. Date labels are UTC.

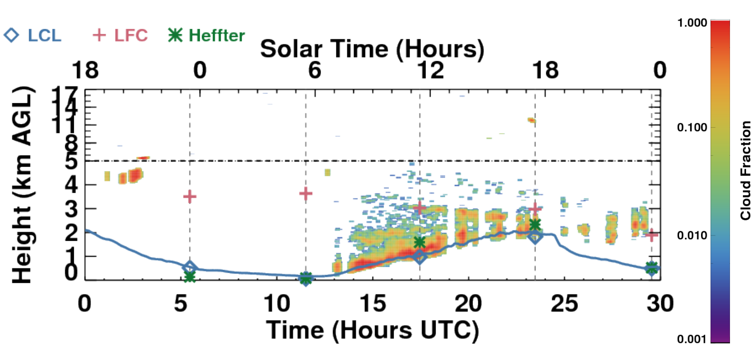


Figure 92. Cloud fraction profiles derived from the KAZR-ARSCL for 18-Aug-2016 at the Central Facility. Note the non-linear vertical axis that emphasizes the lower troposphere. Also indicated are the LCL, LFC, and PBL height based on the Heffter methodology, each of which are calculated from the SONDE product.

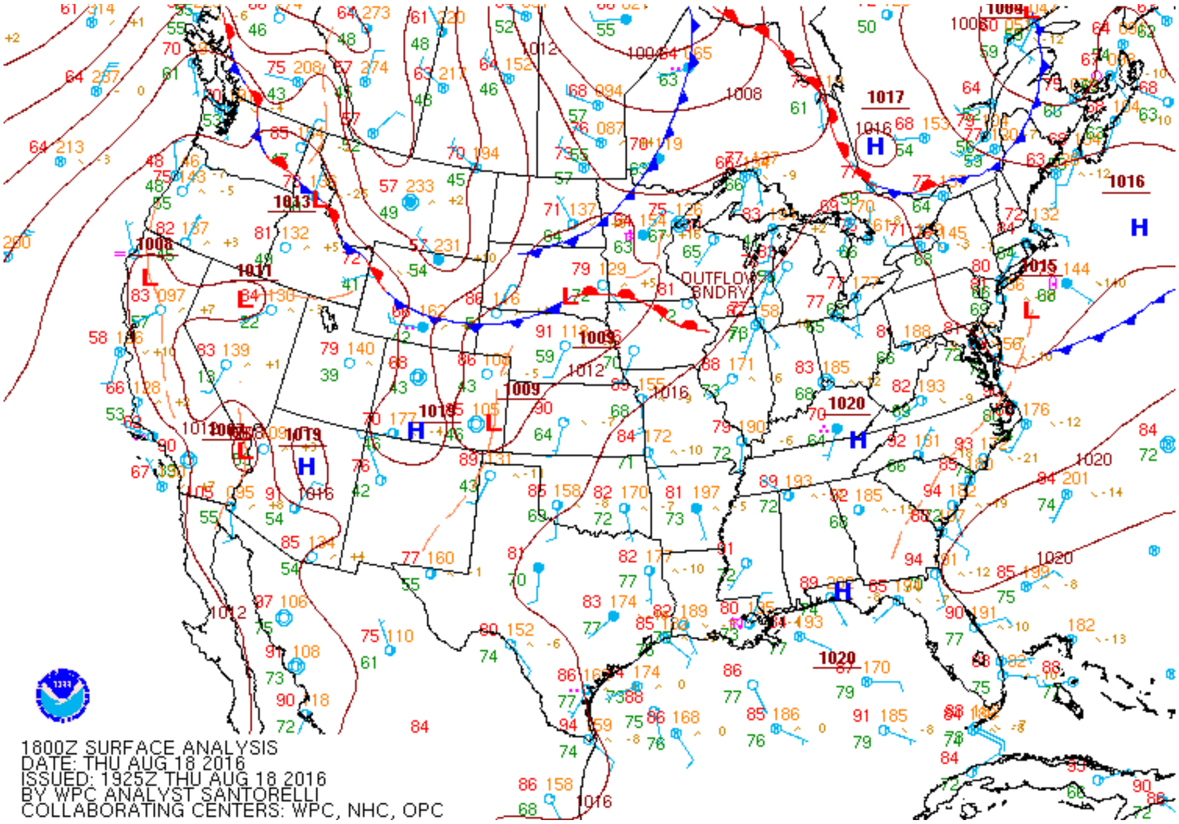


Figure 93. Surface analysis for 18-Aug-2016 18 UTC. Acquired from NWS Weather Prediction Center, http://www.wpc.ncep.noaa.gov/archives/web_pages/sfc/sfc_archive_maps.php?arc-date=08/18/2016&selmap=2016081818&map-type=namussfc.

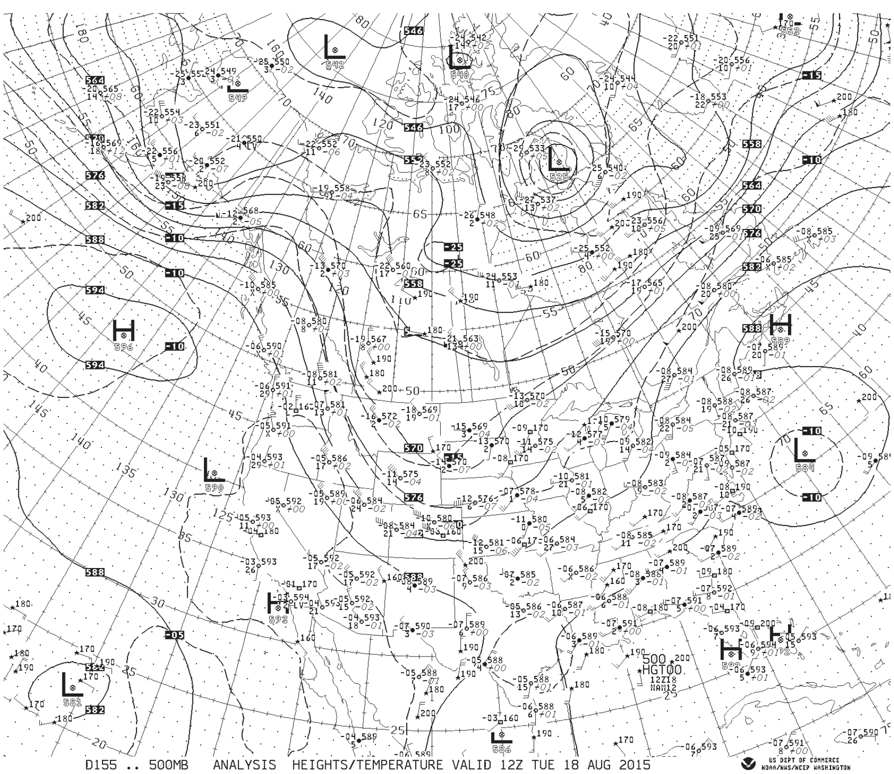
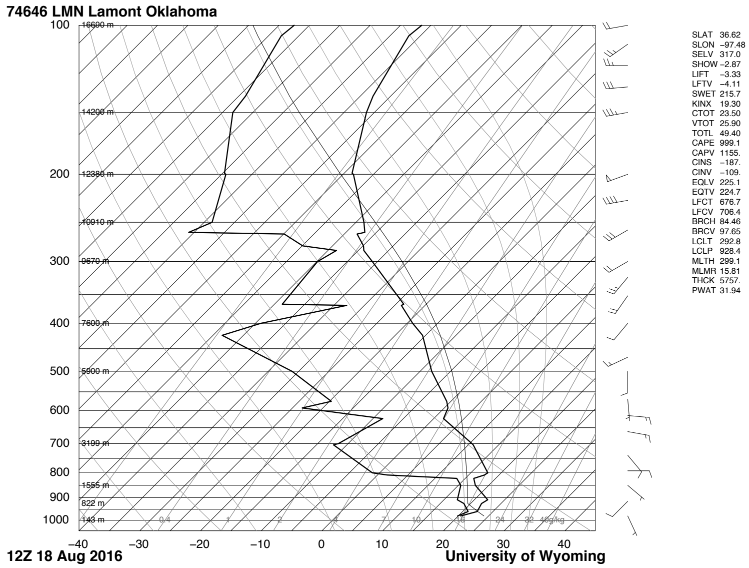
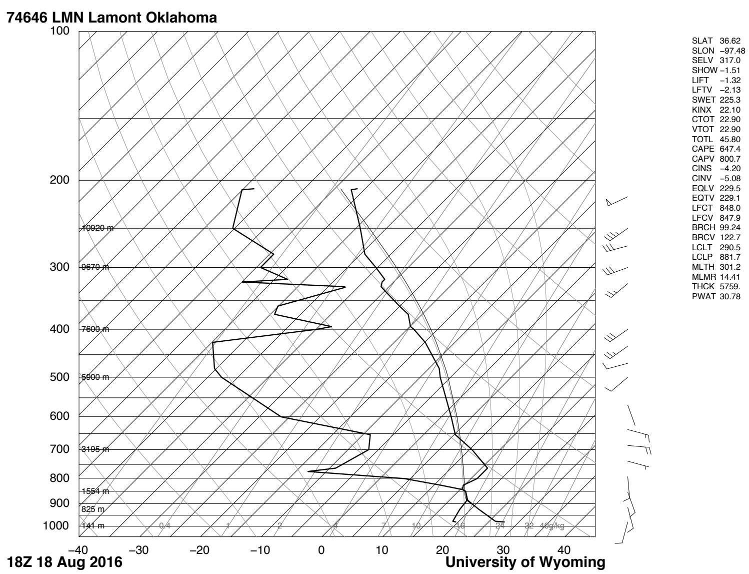


Figure 94. 500 hPa synoptic map for 18-Aug-2016 12 UTC. Acquired from Storm Research and Consulting, http://www.stormresearch.com/ncep/2015/2015_08/2015081812_500.tif.





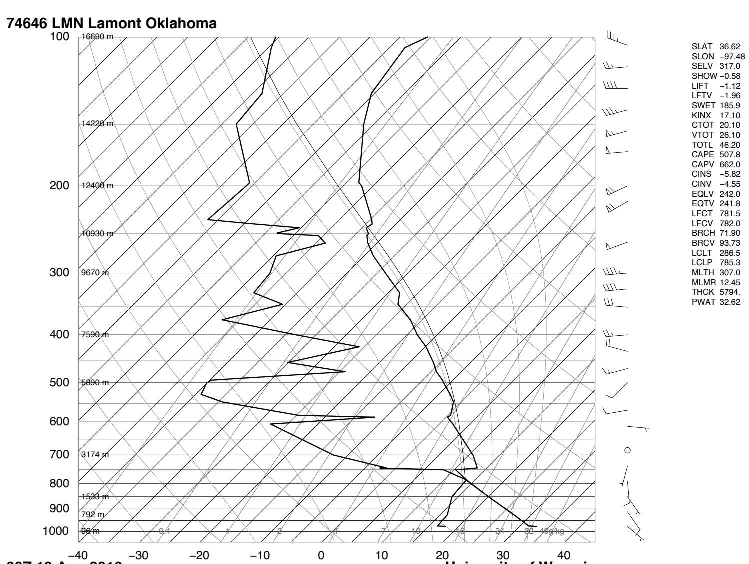


Figure 95. Skew-T log-P diagrams for Lamont, Oklahoma, for the 18-Aug-2016 LASSO case. Acquired from University of Wyoming, http://weather.uwyo.edu/upperair/sounding.html.

Synoptic Conditions for 19-Aug-2016

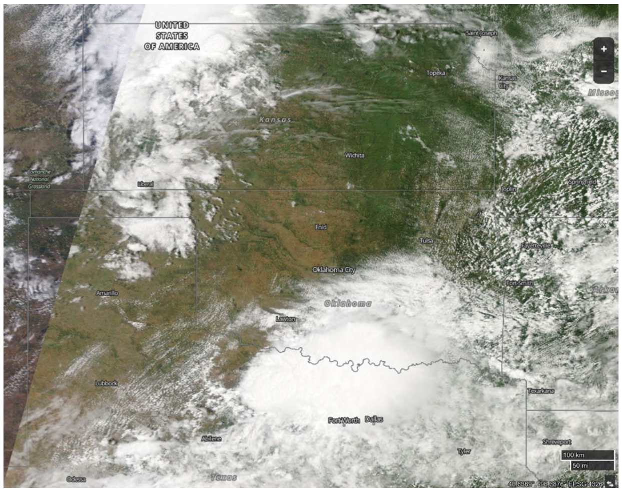


Figure 96. MODIS Terra corrected reflectance for 19-Aug-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uTCCIf.

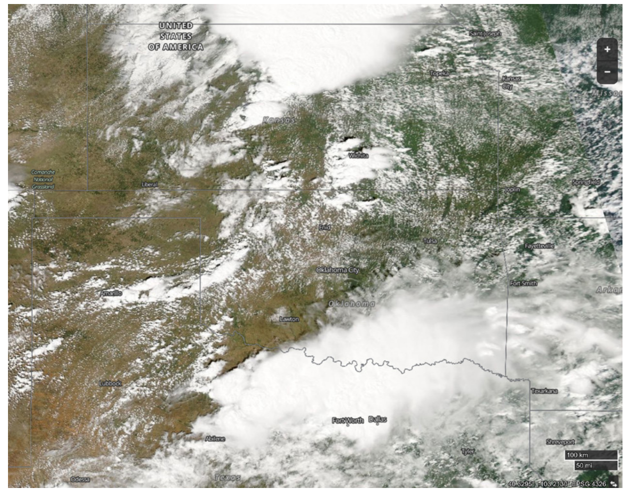


Figure 97. MODIS Aqua corrected reflectance for 19-Aug-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uTIQrF.

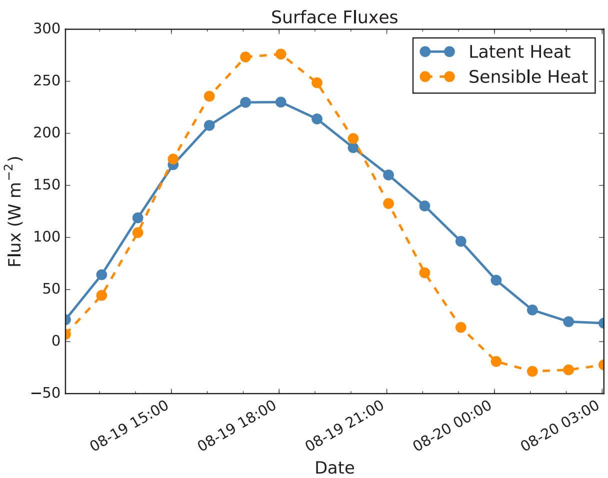


Figure 98. Surface sensible (orange, dashed) and latent (blue, solid) heat fluxes averaged for the SGP region. Hourly values taken from the variational analysis product. Date labels are UTC.

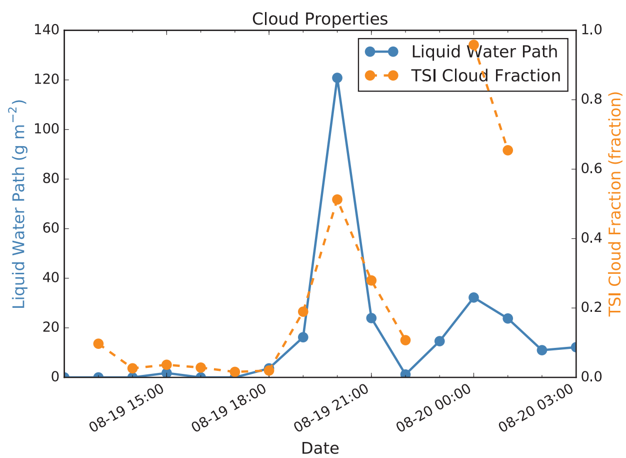


Figure 99. Cloud fraction (orange, dashed) from the TSI and cloud liquid water path (blue, solid) from a hybrid blending of MWRRet and AERIoe at the SGP Central Facility. Date labels are UTC.

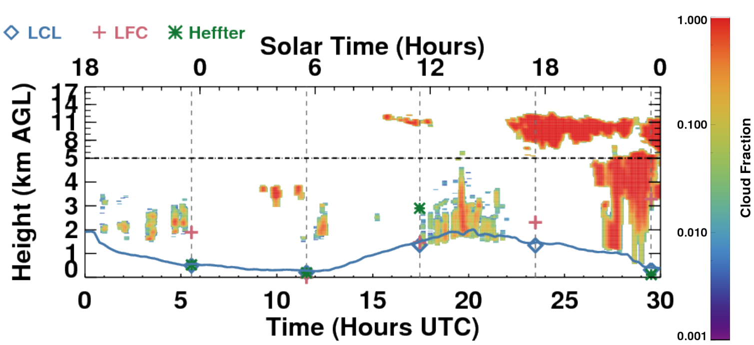


Figure 100. Cloud fraction profiles derived from the KAZR-ARSCL for 19-Aug-2016 at the Central Facility. Note the non-linear vertical axis that emphasizes the lower troposphere. Also indicated are the LCL, LFC, and PBL height based on the Heffter methodology, each of which are calculated from the SONDE product.

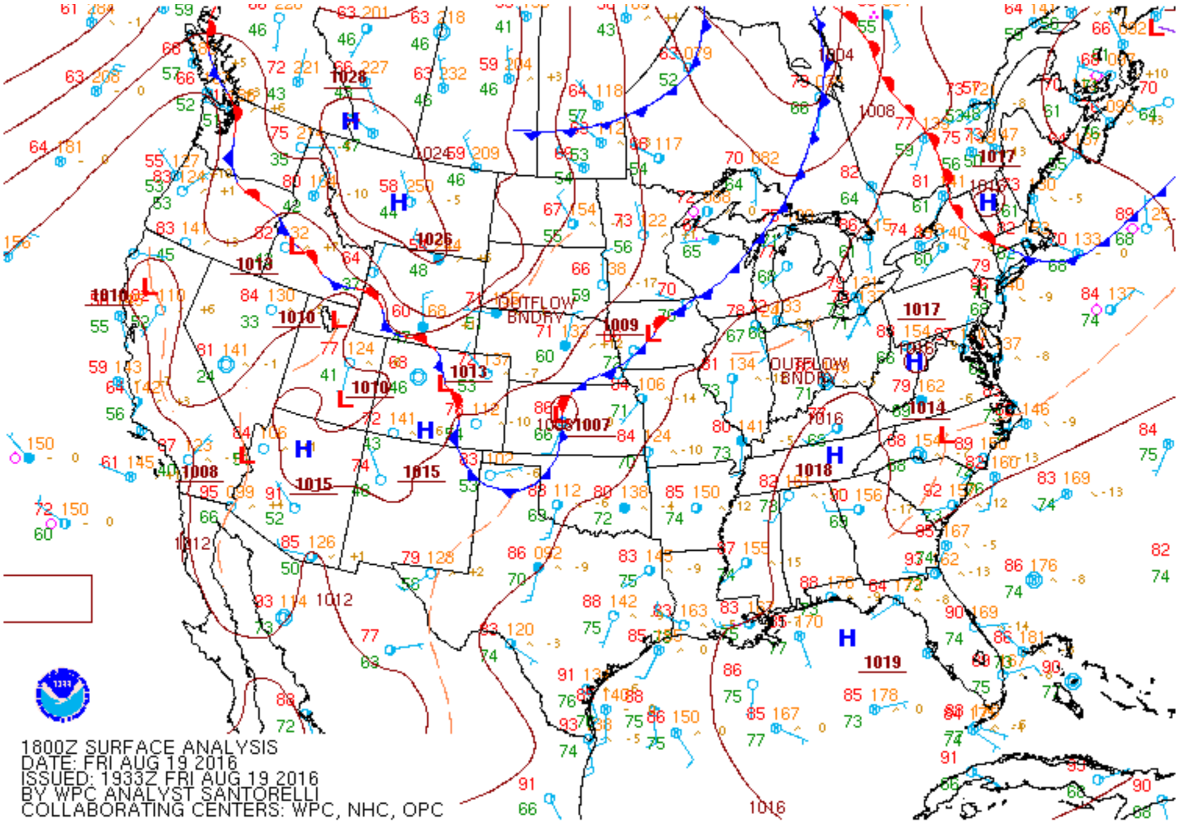


Figure 101. Surface analysis for 19-Aug-2016 18 UTC. Acquired from NWS Weather Prediction Center, http://www.wpc.ncep.noaa.gov/archives/web_pages/sfc/sfc_archive_maps.php?arc-date=08/19/2016&sel-map=2016081918&maptype=namussfc.

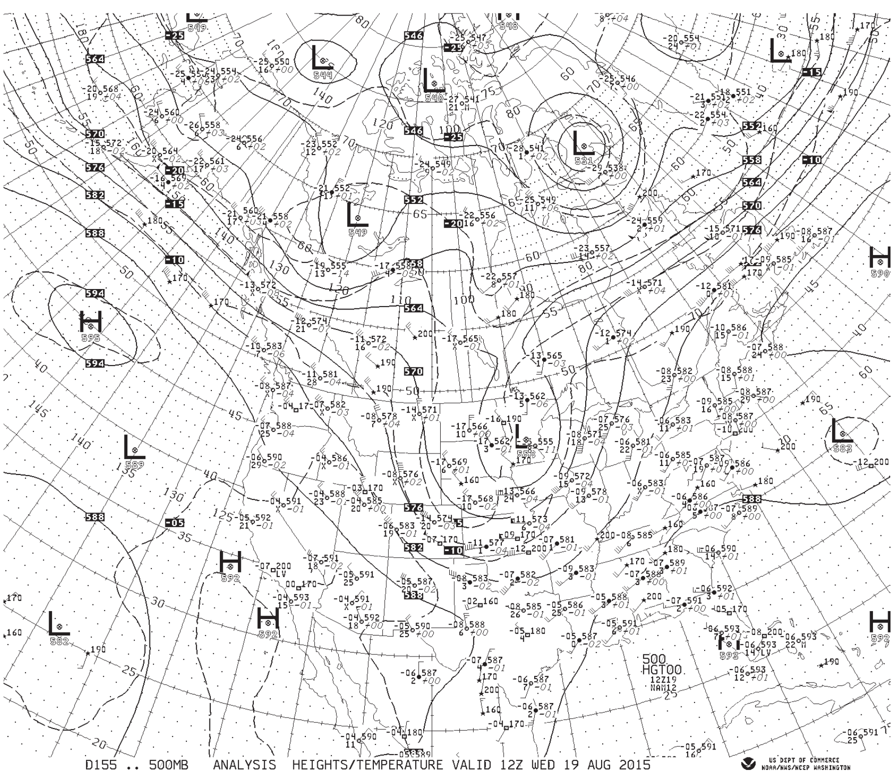


Figure 102. 500 hPa synoptic map for 19-Aug-2016 12 UTC. Acquired from Storm Research and Consulting, http://www.stormresearch.com/ncep/2015/2015_08/2015081912_500.tif.

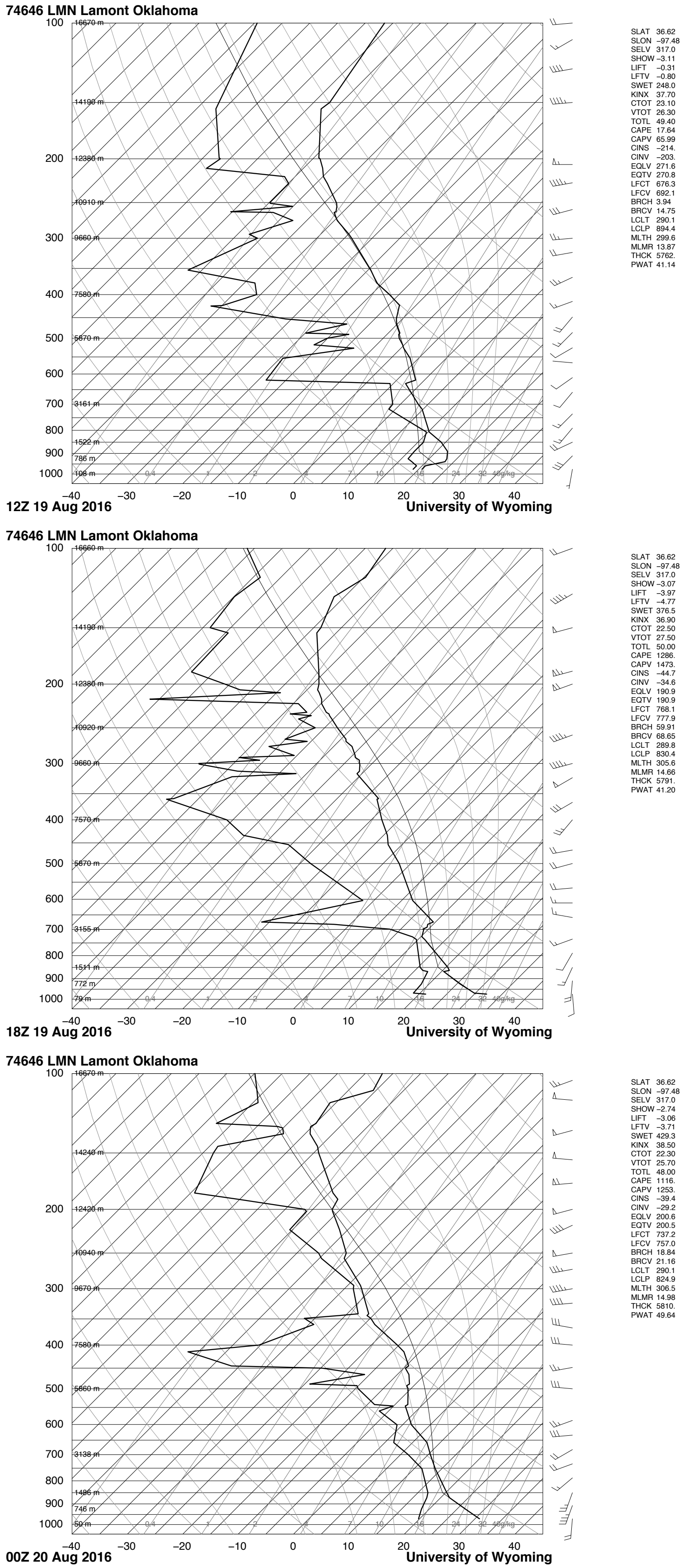


Figure 103. Skew-T log-P diagrams for Lamont, Oklahoma, for the 19-Aug-2016 LASSO case. Acquired from University of Wyoming, http://weather.uwyo.edu/up-perair/sounding.html.

Synoptic Conditions for 30-Aug-2016

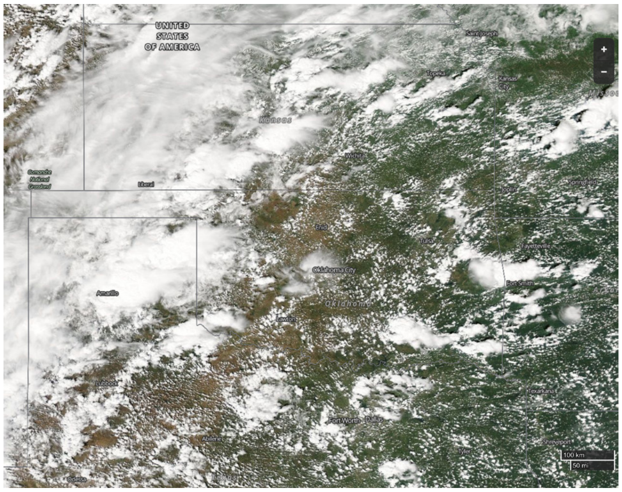


Figure 104. MODIS Terra corrected reflectance for 30-Aug-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uTGOrN.

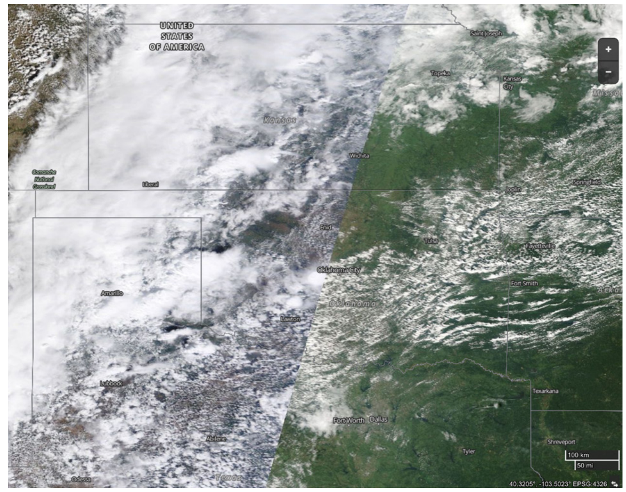


Figure 105. MODIS Aqua corrected reflectance for 30-Aug-2016. Acquired from NASA EOSDIS Worldview, https://go.nasa.gov/2uTP37b.

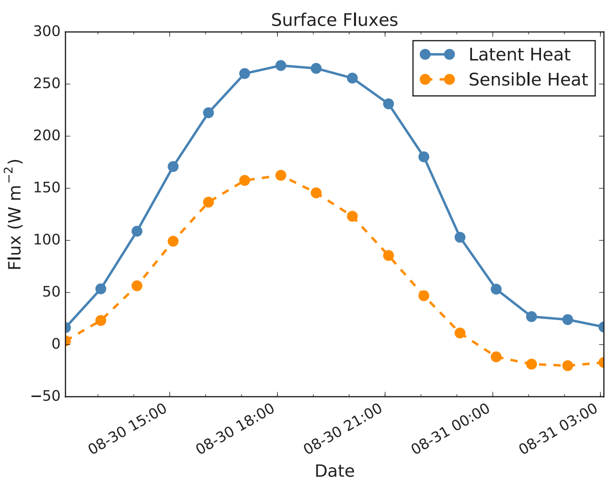


Figure 106. Surface sensible (orange, dashed) and latent (blue, solid) heat fluxes averaged for the SGP region. Hourly values taken from the variational analysis product. Date labels are UTC.

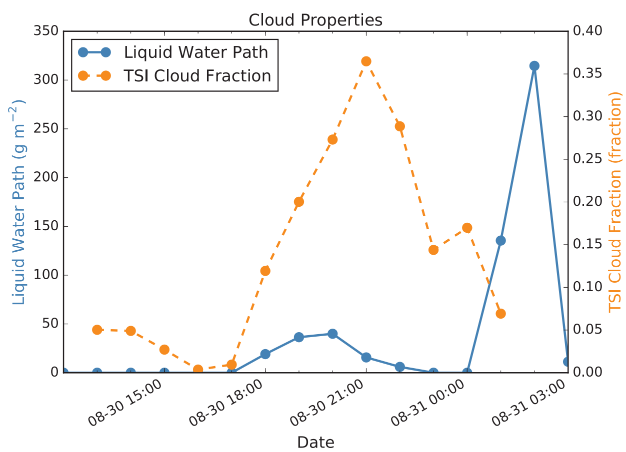


Figure 107. Cloud fraction (orange, dashed) from the TSI and cloud liquid water path (blue, solid) from a hybrid blending of MWRRet and AERIoe at the SGP Central Facility. Date labels are UTC.

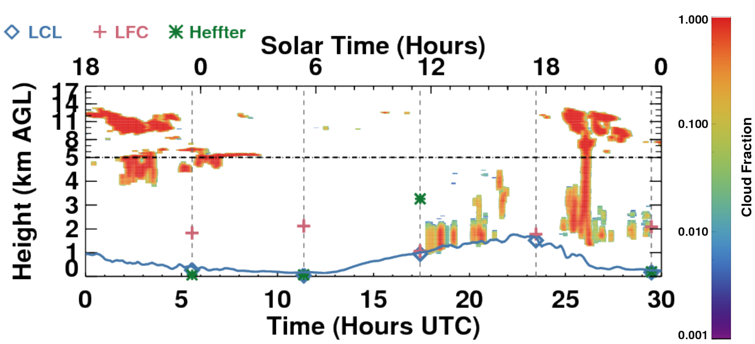


Figure 108. Cloud fraction profiles derived from the KAZR-ARSCL for 30-Aug-2016 at the Central Facility. Note the non-linear vertical axis that emphasizes the lower troposphere. Also indicated are the LCL, LFC, and PBL height based on the Heffter methodology, each of which are calculated from the SONDE product.

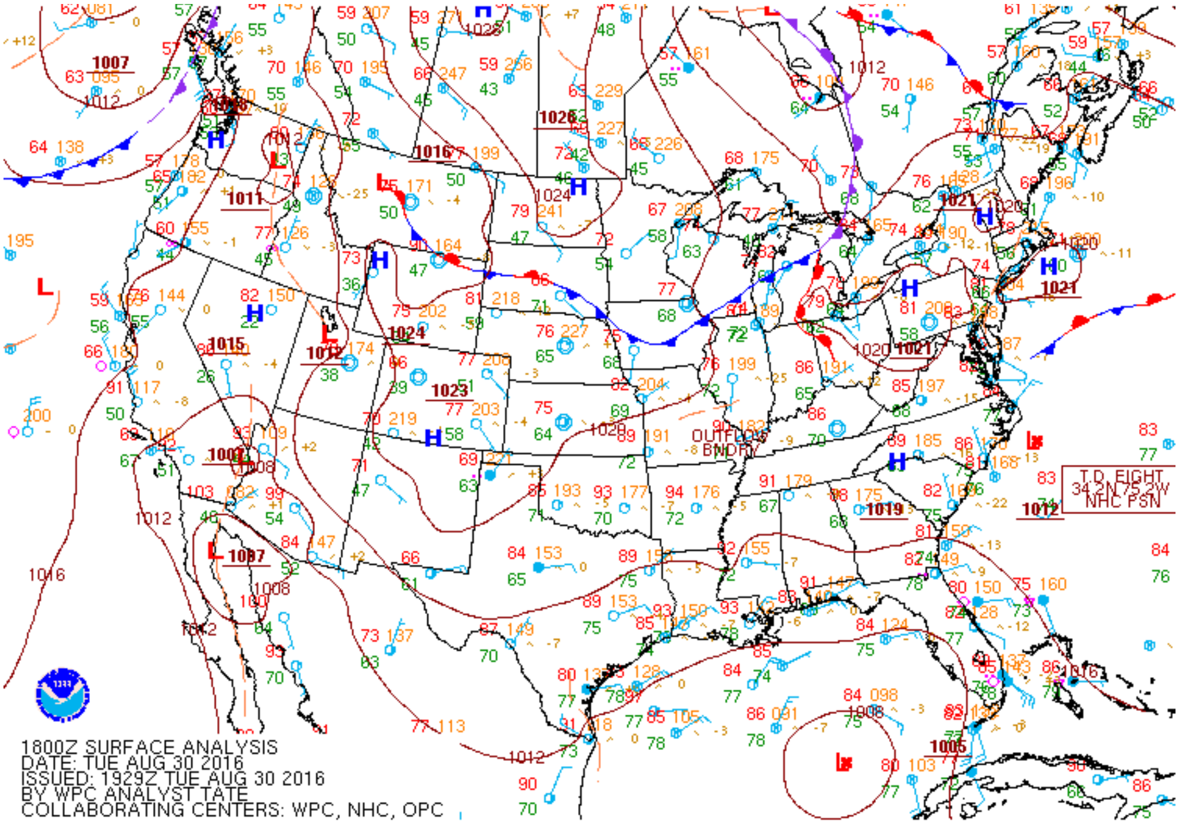


Figure 109. Surface analysis for 30-Aug-2016 18 UTC. Acquired from NWS Weather Prediction Center, http://www.wpc.ncep.noaa.gov/archives/web_pages/sfc/sfc_archive_maps.php?arc-date=08/30/2016&sel-map=2016083018&maptype=namussfc.

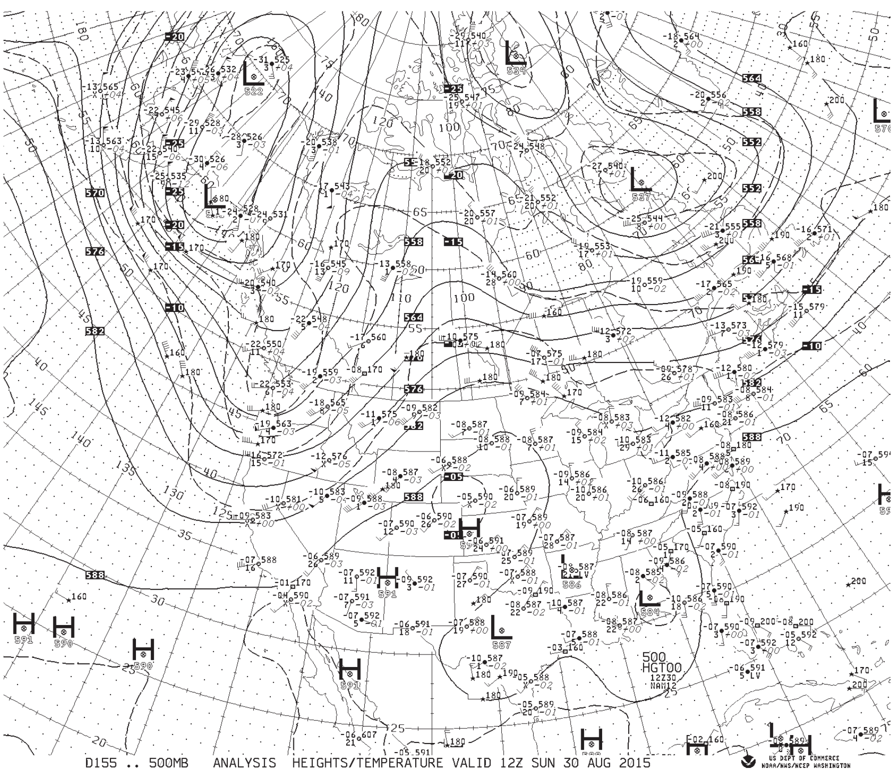
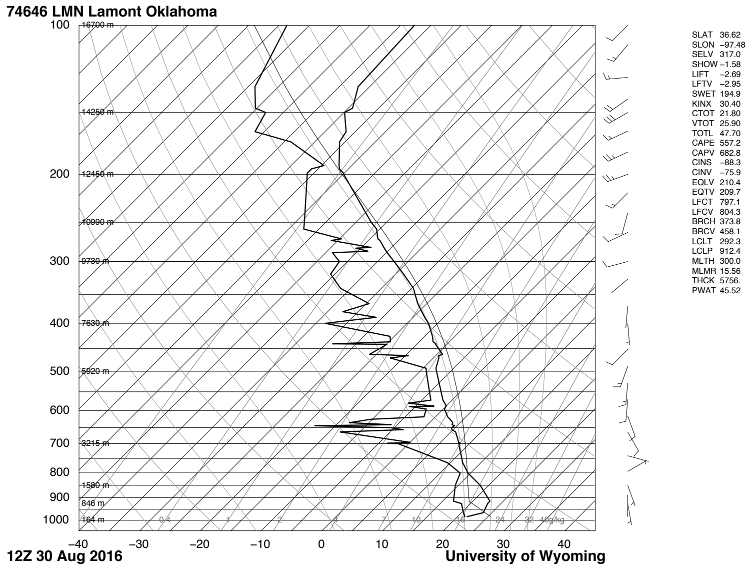
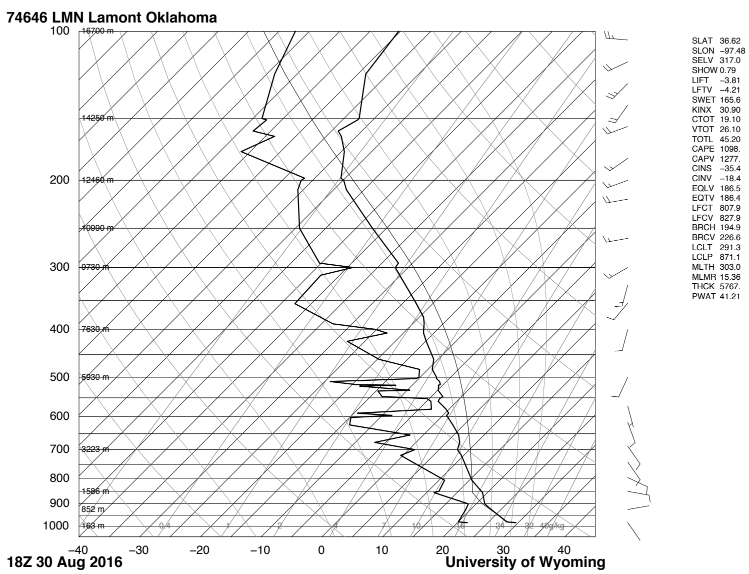


Figure 110. 500 hPa synoptic map for 30-Aug-2016 12 UTC. Acquired from Storm Research and Consulting, http://www.stormresearch.com/ncep/2015/2015_08/2015083012_500.tif.





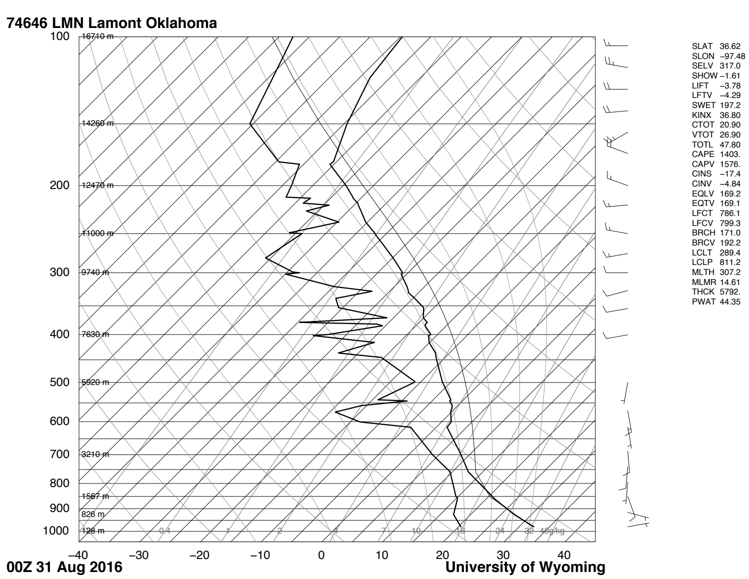


Figure 111. Skew-T log-P diagrams for Lamont, Oklahoma, for the 30-Aug-2016 LASSO case. Acquired from University of Wyoming, http://weather.uwyo.edu/up-perair/sounding.

7.0 References

Clothiaux, EE, TP Ackerman, GG Mace, KP Moran, RT Marchand, MA Miller, and BE Martner. 2000. "Objective determination of cloud heights and radar reflectivities using a combination of active remote sensors at the ARM cart sites." *Journal of Applied Meteorology* 39(5): 645–665, doi:10.1175/1520-0450(2000)039<0645:ODOCHA>2.0.CO;2.

Clough, SA, MW Shephard, EJ Mlawer, JS Delamere, MJ Iacono, K Cady-Pereira, S Boukabara, and PD Brown. 2005. "Atmospheric radiative transfer modeling: A summary of the AER codes." *Journal of Quantitative Spectroscopy and Radiative Transfer* 91(2): 233–244, doi:10.1016/j.jqsrt.2004.05.058.

Deardorff, JW. 1980. "Stratocumulus-capped mixed layers derived from a 3-dimensional model." *Boundary-Layer Meteorology* 18(4): 495–527, doi:10.1007/Bf00119502.

Endo, S, AM Fridlind, W Lin, AM Vogelmann, T Toto, AS Ackerman, GM McFarquhar, RC Jackson, HH Jonsson, and Y Liu. 2015. "RACORO continental boundary layer cloud investigations: 2. Large-eddy simulations of cumulus clouds and evaluation with in situ and ground-based observations." *Journal of Geophysical Research – Atmospheres* 120(12): 5993–6014, doi:10.1002/2014jd022525.

Goldsmith, JEM, FH Blair, SE Bisson, and DD Turner. 1998. "Turn-key Raman lidar for profiling atmospheric water vapor, clouds, and aerosols." *Applied Optics* 37(21): 4979–4990, doi:10.1364/Ao.37.004979.

Gustafson, WI, AM Vogelmann, X Cheng, S Endo, B Krishna, Z Li, T Toto, and H Xiao. 2016. Description of the LASSO Alpha 1 Release. U.S. Department of Energy. DOE/SC-ARM-TR-194, https://www.arm.gov/publications/tech_reports/doe-sc-arm-tr-194.pdf, doi:10.2172/1373564.

Heffter, JL. 1980. "Transport layer depth calculations." *Second Joint Conference on Applications of Air Pollution Meteorology*, New Orleans, Louisiana, American Meteorological Society, https:// books.google.com/books?id=VtSDXwAACAAJ.

Iacono, MJ, JS Delamere, EJ Mlawer, MW Shephard, SA Clough, and WD Collins. 2008. "Radiative forcing by long-lived greenhouse gases: Calculations with the AER radiative transfer models." *Journal of Geophysical Research – Atmospheres* 113: D13103, doi:10.1029/2008jd009944.

Khairoutdinov, MF, and DA Randall. 2003. "Cloud resolving modeling of the ARM summer 1997 IOP: Model formulation, results, uncertainties, and sensitivities." *Journal of the Atmospheric Sciences* 60(4): 607–625, doi:10.1175/ 1520-0469(2003)060<0607:Crmota>2.0.Co;2.

Lamer, K, and P Kollias. 2015. "Observations of fair-weather cumuli over land: Dynamical factors controlling cloud size and cover." *Geophysical Research Letters* 42(20): 8693–8701, doi:10.1002/2015gl064534.

Li, Z, S Feng, Y Liu, W Lin, M Zhang, T Toto, AM Vogelmann, and S Endo. 2015a. "Development of fine-resolution analyses and expanded large-scale forcing properties: 1. Methodology and evaluation." *Journal of Geophysical Research – Atmospheres* 120: 654–666, doi:10.1002/2014jd022245.

Li, ZJ, JC McWilliams, K Ide, and JD Farrara. 2015b. "A multiscale variational data assimilation scheme: Formulation and illustration." *Monthly Weather Review* 143(9): 3804–3822, doi:10.1175/mwr-d-14-00384.1.

Li, ZJ, XP Cheng, WI Gustafson, and AM Vogelmann. 2016. "Spectral characteristics of background error covariance and multiscale data assimilation." *International Journal for Numerical Methods in Fluids* 82: 1035–1048, doi:10.1002/fld.4253.

Mesinger, F, and TL Black. 1992. "On the impact on forecast accuracy of the step-mountain (eta) vs. sigma coordinate." *Meteorological and Atmospheric Physics* 50(1-3): 47–60, doi:10.1007/bf01025504.

Mlawer, EJ, SJ Taubman, PD Brown, MJ Iacono, and SA Clough. 1997. "Radiative transfer for inhomogeneous atmospheres: RRTM, a validated correlated-k model for the longwave." *Journal of Geophysical Research – Atmospheres* 102(D14): 16663–16682, doi:10.1029/97jd00237.

Morrison, H, JA Curry, and VI Khvorostyanov. 2005. "A new double-moment microphysics parameterization for application in cloud and climate models. Part I: Description." *Journal of the Atmospheric Sciences* 62(6): 1665–1677, doi:10.1175/jas3446.1.

Morrison, H, G Thompson, and V Tatarskii. 2009. "Impact of cloud microphysics on the development of trailing stratiform precipitation in a simulated squall line: Comparison of one- and two-moment schemes." *Monthly Weather Review* 137(3): 991–1007, doi:10.1175/2008mwr2556.1.

Newsom, RK, DD Turner, and JEM Goldsmith. 2013. "Long-term evaluation of temperature profiles measured by an operational Raman lidar." *Journal of Atmospheric and Oceanic Technology* 30(8): 1616–1634, doi:10.1175/Jtech-D-12-00138.1.

Prakash, G, J Kumar, E Rush, R Records, A Clodfelter, and J Voyles. 2016. "HPC infrastructure to support the next-generation ARM facility data operations." *2016 IEEE International Conference on Big Data (Big Data)*, 4026–4028, doi:10.1109/BigData.2016.7841098.

Skamarock, WC, JB Klemp, J Dudhia, DO Gill, DM Barker, MG Duda, W Wang, and JG Powers. 2008. "A description of the advanced research WRF version 3." NCAR Technical Note, NCAR/TN-475+STR, National Center for Atmospheric Research, doi:10.5065/D68S4MVH.

Taylor, KE. 2001. "Summarizing multiple aspects of model performance in a single diagram." *Journal of Geophysical Research – Atmospheres* 106(D7): 7183–7192, doi:10.1029/2000jd900719.

Thompson, G, RM Rasmussen, and K Manning. 2004. "Explicit forecasts of winter precipitation using an improved bulk microphysics scheme. Part I: Description and sensitivity analysis." *Monthly Weather Review* 132(2): 519–542, doi:10.1175/1520-0493(2004)132<0519:EFOWPU>2.0.CO;2.

Thompson, G, PR Field, RM Rasmussen, and WD Hall. 2008. "Explicit forecasts of winter precipitation using an improved bulk microphysics scheme. Part II: Implementation of a new snow parameterization. *Monthly Weather Review* 136(12): 5095–5115, doi:10.1175/2008MWR2387.1.

Turner, DD, RA Ferrare, LAH Brasseur, and WF Feltz. 2002. "Automated retrievals of water vapor and aerosol profiles from an operational Raman lidar." *Journal of Atmospheric and Oceanic Technology* 19(1): 37–50, doi:10.1175/ 1520-0426(2002)019<0037:Arowva>2.0.Co;2.

Turner, DD, SA Clough, JC Liljegren, EE Clothiaux, KE Cady-Pereira, and KL Gaustad. 2007. "Retrieving liquid water path and precipitable water vapor from the Atmospheric Radiation Measurement (ARM) microwave radiometers." *IEEE Transactions on Geoscience and Remote Sensing* 45(11): 3680–3690, doi:10.1109/Tgrs.2007.903703.

Turner, DD, and U Löhnert. 2014. "Information content and uncertainties in thermodynamic profiles and liquid cloud properties retrieved from the ground-based atmospheric emitted radiance interferometer (AERI)." *Journal of Applied Meteorology and Climatology* 53(3): 752–771, doi:10.1175/JAMC-D-13-0126.1.

Vogelmann, AM, AM Fridlind, T Toto, S Endo, W Lin, J Wang, S Feng, Y Zhang, DD Turner, Y Liu, Z Li, S Xie, AS Ackerman, M Zhang, and M Khairoutdinov. 2015. "RACORO continental boundary layer cloud investigations: 1. Case study development and ensemble large-scale forcings." *Journal of Geophysical Research – Atmospheres* 120(12): 5962–5992, doi:10.1002/2014jd022713.

Wulfmeyer, V, S Pal, DD Turner, and E Wagner. 2010. "Can water vapour Raman lidar resolve profiles of turbulent variables in the convective boundary layer?" *Boundary-Layer Meteorology* 136(2): 253–284, doi:10.1007/s10546-010-9494-z.

Xie, S, RB McCoy, SA Klein, RT Cederwall, WJ Wiscombe, MP Jensen, KL Johnson, EE Clothiaux, KL Gaustad, CN Long, JH Mather, SA McFarlane, Y Shi, J-C Golaz, Y Lin, SD Hall, RA McCord, G Palanisamy, and DD Turner. 2010. "Clouds and more: ARM climate modeling best estimate data: A new data product for climate studies." *Bulletin of the American Meteorological Society* 91(1): 13–20, doi:10.1175/2009BAMS2891.1.

Xie, SC, RT Cederwall, and MH Zhang. 2004. "Developing long-term single-column model/cloud system-resolving model forcing data using numerical weather prediction products constrained by surface and top of the atmosphere observations." *Journal of Geophysical Research – Atmospheres* 109(D1): D01104, doi:10.1029/2003jd004045.

Zhang, MH, and JL Lin. 1997. "Constrained variational analysis of sounding data based on column-integrated budgets of mass, heat, moisture, and momentum: Approach and application to ARM measurements." *Journal of the Atmospheric Sciences* 54(11): 1503–1524, doi:10.1175/1520-0469(1997)054<1503:Cvaosd>2.0.Co;2.

Zhang, MH, JL Lin, RT Cederwall, JJ Yio, and SC Xie. 2001. "Objective analysis of ARM IOP data: Method and sensitivity." *Monthly Weather Review* 129(2): 295–311, doi:10.1175/1520-0493(2001)129<0295:Oa oaid>2.0.Co;2.

Appendix A: Alpha 2 Updates

The Alpha 2 release includes changes to both the modeling and analysis portions of LASSO. The following lists the updates for Alpha 2.

Forcing Data Set-Related Updates

- Versions of MSDA are included that incorporate wind profiles from the four SGP radar wind profilers.

LES-Related Updates

- Standardized on 10-minute output interval.
- Standardized vertical grid to 227 levels.
- SAM model changes
  - Fixed uninitialized array in relation to background aerosol in Morrison microphysics.
  - Patch for handling long file names.
- WRF model changes
  - Updated to v3.8.1.
  - Changed cloud masking for statistical output in WRF to consider cloud ice in addition to cloud water.
  - Added low-cloud, vertical-velocity variance, and cloud-core vertical-velocity variance to statistical output.
  - Added time-averaged values for each grid column (CSV variables for volume variables and CSS for surface/slab variables) in the WRF statistical output to complement the domain-wide time-averaged values (CSP for profile variables and CST for time series variables).
  - Fixed many bugs related to averaging in the statistics output (recommend avoiding using these variables from the Alpha 1 release).
  - Fixed halo treatment of TKE for periodic boundaries.
  - To enable parallel execution of WRF initialization, improved handling of random temperature perturbations during model initialization to avoid imposing signatures of the model domain decomposition within the perturbations.

Metric- and Diagnostic-Related Updates

- LWP is now based on AERIoe retrievals using AERI and microwave radiances, augmented with MWRRet1 retrievals.
- Added LCL_Domain metric that provides the domain-averaged LCL value and standard deviation.
- Temperatures for the boundary-layer profiles and mid-boundary layer are now based on a hybrid profile from Raman lidar and AERIoe retrievals.
- Mid-boundary-layer moisture and temperature changed from a constant height range to one that is fixed to a value determined specifically for each case.
- Added cloud-base-height metric based on Doppler lidar data at the Central Facility and boundary facilities.
- Minor bug fixes to metric calculations.

Bundle Browser- and Data Retrieval-Related Updates

- Added a multi-day comparison capability to the Bundle Browser.
- Linked sliders for metric values to the plots in the Bundle Browser so that the plots dynamically redraw when the slider is moved to change the selected metric range.

- Linked the data-ordering process directly to selections in the Bundle Browser.
- Added ability to find and order LASSO data from Data Discovery.
- Can now take advantage of Globus for downloading files.
- Data bundle files grouped into two tar files to simplify retrieval.

Appendix B: Evaluation Data

In the Evaluation Data Section, the surface-based observations of in-cloud LWP, 1D cloud fraction, and mid-boundary-layer moisture and temperature were introduced and are described here in further detail.

In-Cloud Liquid Water Path (LWP)

Retrieval of in-cloud LWP (g m^{-2}) is based on MWR-only retrievals using the 2-channel MWRRet microwave radiometer retrieval (Turner et al. 2007) and retrievals from a new optimal estimation framework (AERIoe) (Turner and Löhnert 2014) that uses spectral infrared radiances measured by the atmospheric emitted radiance interferometer (AERI) and MWR radiances. (The AERIoe algorithm also retrieves thermodynamic profiles for the lower atmosphere that are used below.) The resulting LWP data have excellent sensitivity at low LWP (<40 g m^{-2}) and a full dynamic LWP range (up to 1000 g m^{-2}). Details on the retrievals and how they are combined into a single LWP datum are described next.

The Alpha 2 implementation of AERIoe uses the AERI radiances and the MWR-3C microwave radiometer radiances at 23.834 and 30 GHz. The use of infrared radiances (8–13 µm) provides sensitivity at small LWP (<~50 g m^{-2}), yielding an uncertainty of ~30% for LWP <5 g m^{-2} but manifests signal saturation by 50 g m^{-2}. As retrievals approach this saturation limit of the AERI signal, AERIoe uses more information from the MWR channels until it reaches a point where LWP is based solely on MWR data. AERIoe retrievals have been compared to those from MWRRet1, which uses MWR-2C-measured microwave brightness temperatures (23.8 and 31.4 GHz) that have been bias corrected and yield an LWP uncertainty of 20–30 g m^{-2}. For lower LWP, the two retrievals agree to within the uncertainty of MWRRet1; however, at larger LWP (e.g., > 100 g m^{-2}) AERIoe values tend to be lower than from MWRRet1. Further evaluations are needed to understand the differences between the two retrievals at larger LWP values.

The AERIoe and MWRRet1 retrievals were combined into a single data set in a procedure that favors AERIoe values for LWP < 50 g m^{-2} and MWRRet1 for LWP > 50 g m^{-2}. These preferences are because AERIoe will have superior quality for low LWP where the retrieval relies on AERI radiances, and for larger LWP MWRRet1 has the benefit from a long operating history for both the MWR-2C instrument and the retrieval algorithm. Both retrievals have been regridded and interpolated to 10-s resolution. Times for which data are missing in the native data sets remain missing in the 10-s data sets. For MWRRet1, which has a high-frequency time grid (~30s), such missing points are indicated in the QC information. For AERIoe, which has a more irregular reporting frequency and potentially data gaps, all AERIoe data are used (without QC screening, which is not yet available for AERIoe LWP values) and missing points have been determined by a data gap of 10 minutes or more. The LWP data set starts with the regridded AERIoe LWP data. Missing points use MWRRet1 should a value be available and its LWP > 30 g m^{-2}. For all points where MWRRet1 > 50 g m^{-2}, the MWRRet1 data are used in place of the AERIoe values. The final LWP time series can thus consist of a combination of AERIoe values at any LWP value plus the inclusion of MWRRet1 values starting at LWP > 30 g m^{-2}.

To obtain in-cloud LWP values, cloud screening has been applied to the observations and simulations so that clear-sky values are not used in the time-averaged quantities. For observations, LWP values >1 g m^{-2} are screened as being cloudy values. In the simulations, the summation of the cloud and rainwater mixing ratios within a model grid cell is $>10^{-7}$ kg kg^{-1} to be considered 'cloud' and column integrals are computed of the cloudy cells to yield a 2D LWP field. The average is taken of the LWP columns with values >1 g m^{-2} to produce the simulated in-cloud, domain-averaged LWP. In this procedure, the largest source of uncertainty is the incomplete sampling of broken clouds across the domain in the observations.

Currently, only LWP observations are available from the SGP Central Facility (a single point). However, four new profiling boundary facilities have recently begun operation. In a future LASSO release, when the LWP retrievals from these boundary sites become available, they will be used in a LWP average that will be more representative of the domain.

1D Cloud Fraction (CF)

One-dimensional (1D) cloud fraction is determined from the TSI and ARSCL. The TSI is a hemispheric-viewing camera providing retrievals of fractional sky cover during daytime for 'opaque' and 'thin' clouds. The opaque fractional value is used, which is most relevant to the boundary-layer clouds of interest. Measurements from the 100° field of view are used to minimize cloud fraction (CF) overestimation due to scattering from cloud edges, particularly from clouds on the horizon. Cloud fraction is derived as 15-minute averages that are averaged up to a 1-h value. Cloud fraction from ARSCL is derived as the cloud frequency per time interval as described in Xie et al. (2010), which assumes a horizontally uniform cloud field distribution (i.e., the frozen turbulence assumption). Boundary-layer cloud fractions are 15-minute averages of fractional occurrence, reported at 1-minute intervals, computed from the vertically resolved ARSCL cloud mask for clouds lower than 5 km, which are averaged up to a 1-h value. For ARSCL CF, the largest uncertainties are from the application of the frozen turbulence assumption to broken cloud fields and insect contamination of the radar returns. For the Alpha 2 release, the recommended 1D CF measurement is from the TSI, as cloud above the boundary layer generally has a negligible impact on CF for the cases provided when boundary-layer cloud is present.

In simulations, CF is computed from the simulations in a manner that follows from the computation of in-cloud LWP: a grid cell is identified as cloudy if the sum of the cloud and rainwater mixing ratios is $>10^{-7}$ kg kg^{-1}, column integrals are taken of the cloudy cells to yield a 2D LWP field, and CF is determined as the fraction of the 2D grid with LWP >1 g m^{-2}, which is roughly the lower detectability limit of the measurements. Below this value, retrievals might measure haze or thin cirrus. The 1 g m^{-2} cutoff is somewhat arbitrary but yields similar cloud fractions to when 0.1 g m^{-2} is used. Note, however, that a 0 g m^{-2} cutoff can yield greater cloud fractions by up to 0.2 (Vogelmann et al. 2015).

Boundary-Layer Thermodynamic Profiles

A Raman lidar (Goldsmith et al. 1998) provides high-frequency vertical profiles of the boundary-layer Q_v (Wulfmeyer et al. 2010), temperature (Newsom et al. 2013), and relative humidity (RH) computed from these measurements. The native temporal and vertical resolutions of the measurements are 10 s and 7.5 m, which are averaged and provided at 10-minute and ~60-m resolution using automated processing algorithms (Newsom et al. 2013; Turner et al. 2002). The lidar signal saturates above cloud base. Based on comparisons with sondes, Raman lidar water vapor measurements are used above 0.1 km and temperature measurements are used above 1.25 km. For the temperature profiles, values are used from AERIoe retrievals (see LWP description above) below 1.25 km, Raman lidar values are used above 1.5 km, and a linear weighting with height of the two values is used between these two heights. The high spatial and temporal resolution of the Raman lidar and AERIoe data are valuable for assessing the simulated boundary-layer evolution; however, note that it is a point measurement that cannot represent the total variation across the model-simulated domain.

Appendix C: LASSO Alpha 2 File Contents

This appendix lists the contents of the files contained in data bundles in this release. Most lists take the form of netCDF header dumps with the exception of one text file. The list is organized by the directory structure shown in Figure 7. Files from simulation IDs 1 and 100 from the 10-Jun-2016 case are used as an example. Simulation 1 represents output from WRF, and simulation 100 represents output from SAM. Only one file is included if it is identical for both models.

File: .../20160610/metrics/sgplassostatC1.m1.20160610.000000.nc

Description: Contains metrics for surface meteorology and cloud variables comparing the simulations in the case to observations.

```
netcdf sgplassostatC1.m1.20160610.000000 {
dimensions:
	time = UNLIMITED ; // (1 currently)
	simulation_id = 40 ;
	bound = 2 ;
	variable = 12 ;
	variable_name_length = 100 ;
variables:
	int base_time ;
		base_time:long_name = "Base time in Epoch" ;
		base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
		base_time:ancillary_variables = "time_offset" ;
	double time_offset(time) ;
		time_offset:long_name = "Time offset from base_time" ;
		time_offset:units = "seconds since 2016-06-10 11:30:00 0:00" ;
		time_offset:ancillary_variables = "base_time" ;
	double time(time) ;
		time:long_name = "Time offset from midnight" ;
		time:units = "seconds since 2016-06-10 00:00:00 0:00" ;
		time:calendar = "gregorian" ;
		time:standard_name = "time" ;
		time:bounds = "time_bounds" ;
	double time_bounds(time, bound) ;
		time_bounds:long_name = "Time cell bounds" ;
		time_bounds:units = "unitless" ;
	float simulation_id(simulation_id) ;
		simulation_id:long_name = "Simulation ID" ;
		simulation_id:units = "unitless" ;
	int variable(variable) ;
		variable:long_name = "Variable" ;
		variable:units = "unitless" ;
	char variable_name(variable, variable_name_length) ;
		variable_name:long_name = "Variable name" ;
		variable_name:units = "unitless" ;
	float taylor_skill(time, variable, simulation_id) ;
		taylor_skill:long_name = "Taylor skill score" ;
		taylor_skill:units = "unitless" ;
		taylor_skill:missing_value = -9999.f ;
	float taylor_r0(time, variable, simulation_id) ;
		taylor_r0:long_name = "Taylor_r0" ;
		taylor_r0:units = "unitless" ;
		taylor_r0:missing_value = -9999.f ;
```

```
	float relative_mean_skill(time, variable, simulation_id) ;
		relative_mean_skill:long_name = "Skill of relative mean" ;
		relative_mean_skill:units = "unitless" ;
		relative_mean_skill:missing_value = -9999.f ;
	float net_skill(time, variable, simulation_id) ;
		net_skill:long_name = "Skill of: relative mean skill and taylor skill" ;
		net_skill:units = "unitless" ;
		net_skill:missing_value = -9999.f ;
	float cloud_1d_skill(time, simulation_id) ;
		cloud_1d_skill:long_name = "Skill of: LWP net skill and TSI cloud frac-
tion net skill" ;
		cloud_1d_skill:units = "unitless" ;
		cloud_1d_skill:missing_value = -9999.f ;
	float total_cloud_skill(time, simulation_id) ;
		total_cloud_skill:long_name = "Skill of: cloud mask net skill and 1D
cloud skill (cloud_1d_skill)" ;
		total_cloud_skill:units = "unitless" ;
		total_cloud_skill:missing_value = -9999.f ;
		total_cloud_skill:comments = "Cloud mask net skill found in datastream:
lassostat2d.m1" ;
	float lat ;
		lat:long_name = "North latitude" ;
		lat:units = "degree_N" ;
		lat:valid_min = -90.f ;
		lat:valid_max = 90.f ;
		lat:standard_name = "latitude" ;
	float lon ;
		lon:long_name = "East longitude" ;
		lon:units = "degree_E" ;
		lon:valid_min = -180.f ;
		lon:valid_max = 180.f ;
		lon:standard_name = "longitude" ;
	float alt ;
		alt:long_name = "Altitude above mean sea level" ;
		alt:units = "m" ;
		alt:standard_name = "altitude" ;

// global attributes:
		:process_version = "Alpha 2 Release" ;
		:dod_version = "Southern Great Plains (SGP), Lamont, Oklahoma" ;
		:site_id = "sgp" ;
		:platform_id = "m1" ;
		:facility_id = "C1" ;
		:data_level = "sgplassostat2dC1.m1" ;
		:location_description = "lassostat" ;
		:contacts = "William.Gustafson@pnnl.gov, Vogelmann@bnl.gov" ;
		:doi = "sgplassostatC1.m1" ;
		:history = "Thu Aug 24 21:51:17 2017" ;
}
```

File: .../20160610/metrics/sgplassostat2dC1.m1.20160610.txt

Description: Skill scores for time-height cloud comparisons by simulation within the given case date.

```
netcdf sgplassostat2dC1.m1.20160610.000000 {
dimensions:
	time = UNLIMITED ; // (1 currently)
	simulation_id = 46 ;
	bound = 2 ;
```

```
variables:
	int base_time ;
		base_time:long_name = “Base time in Epoch” ;
		base_time:units = “seconds since 1970-1-1 0:00:00 0:00” ;
		base_time:ancillary_variables = “time_offset” ;
	double time_offset(time) ;
		time_offset:long_name = “Time offset from base_time” ;
		time_offset:units = “seconds since 2016-06-10 12:00:00 0:00” ;
		time_offset:ancillary_variables = “base_time” ;
	double time(time) ;
		time:long_name = “Time offset from midnight” ;
		time:units = “seconds since 2016-06-10 00:00:00 0:00” ;
		time:calendar = “gregorian” ;
		time:standard_name = “time” ;
		time:bounds = “time_bounds” ;
	double time_bounds(time, bound) ;
		time_bounds:long_name = “Time cell bounds” ;
		time_bounds:units = “unitless” ;
	float simulation_id(simulation_id) ;
		simulation_id:long_name = “Simulation ID” ;
		simulation_id:units = “unitless” ;
	float critical_success_index(time, simulation_id) ;
		critical_success_index:long_name = “Critical success index” ;
		critical_success_index:units = “unitless” ;
		critical_success_index:missing_value = -9999.f ;
	float frequency_bias(time, simulation_id) ;
		frequency_bias:long_name = “Frequency bias” ;
		frequency_bias:units = “unitless” ;
		frequency_bias:missing_value = -9999.f ;
	float equitable_threat_score(time, simulation_id) ;
		equitable_threat_score:long_name = “Equitable threat score” ;
		equitable_threat_score:units = “unitless” ;
		equitable_threat_score:missing_value = -9999.f ;
	float ets_skill(time, simulation_id) ;
		ets_skill:long_name = “ETS skill” ;
		ets_skill:units = “unitless” ;
		ets_skill:missing_value = -9999.f ;
	float frequency_bias_skill(time, simulation_id) ;
		frequency_bias_skill:long_name = “Frequency bias skill” ;
		frequency_bias_skill:units = “unitless” ;
		frequency_bias_skill:missing_value = -9999.f ;
	float cloud_mask_2d_net_skill(time, simulation_id) ;
		cloud_mask_2d_net_skill:long_name = “2D Cloud mask net skill” ;
		cloud_mask_2d_net_skill:units = “unitless” ;
		cloud_mask_2d_net_skill:missing_value = -9999.f ;
	float lat ;
		lat:long_name = “North latitude” ;
		lat:units = “degree_N” ;
		lat:valid_min = -90.f ;
		lat:valid_max = 90.f ;
		lat:standard_name = “latitude” ;
	float lon ;
		lon:long_name = “East longitude” ;
		lon:units = “degree_E” ;
		lon:valid_min = -180.f ;
		lon:valid_max = 180.f ;
		lon:standard_name = “longitude” ;
	float alt ;
		alt:long_name = “Altitude above mean sea level” ;
		alt:units = “m” ;
```

```
              alt:standard_name = "altitude" ;

// global attributes:
              :process_version = "Alpha 2 Release" ;
              :dod_version = "lassostat2d.m1-2.0.hdr" ;
              :input_source = "sgplassomod2d*C1.m1.20160610.*.nc" ;
              :site_id = "sgp" ;
              :platform_id = "lassodiagobsmod2d" ;
              :facility_id = "C1" ;
              :data_level = "m1" ;
              :location_description = "Southern Great Plains (SGP), Lamont, Oklahoma" ;
              :contacts = "William.Gustafson@pnnl.gov, Vogelmann@bnl.gov" ;
              :doi = "10.5439/1342961" ;
              :history = "Thu Aug 24 20:20:15 2017" ;
}
```

File: .../20160610/sim0001/obs_model/sgplassodiagobsmod1C1.m1.20160610.000000.nc

Description: Directly comparable model output from sgplassomod1C1.m1.20160610.000000 and observations on coincident hourly time intervals.

```
netcdf sgplassodiagobsmod1C1.m1.20160610.000000 {
dimensions:
       time = UNLIMITED ; // (29 currently)
       source_type = 2 ;
variables:
       int base_time ;
              base_time:long_name = "Base time in Epoch" ;
              base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
              base_time:ancillary_variables = "time_offset" ;
       double time_offset(time) ;
              time_offset:long_name = "Time offset from base_time" ;
              time_offset:units = "seconds since 2016-06-10 00:00:00 0:00" ;
              time_offset:ancillary_variables = "base_time" ;
              time_offset:comments = "time offset corresponds to center of averaging
interval" ;
       double time(time) ;
              time:long_name = "Time offset from midnight" ;
              time:units = "seconds since 2016-06-10 00:00:00 0:00" ;
              time:calendar = "gregorian" ;
              time:standard_name = "time" ;
       int source_type(source_type) ;
              source_type:long_name = "Type of data source" ;
              source_type:units = "unitless" ;
              source_type:description = "This field contains integer values which should
be interpreted as listed." ;           source_type:flag_method = "integer" ;
              source_type:flag_0_description = "Observation" ;
              source_type:flag_1_description = "Model" ;
       float low_cloud_fraction_arscl(time, source_type) ;
              low_cloud_fraction_arscl:long_name = "Frequency of 15-min cloud occur-
rence, as identified by ARSCL, below 5km AGL" ;
              low_cloud_fraction_arscl:units = "1" ;
              low_cloud_fraction_arscl:missing_value = -9999.f ;
              low_cloud_fraction_arscl:valid_min = 0.f ;
              low_cloud_fraction_arscl:valid_max = 1.f ;
              low_cloud_fraction_arscl:cell_methods = "time: mean" ;
              low_cloud_fraction_arscl:ancillary_variables = "qc_low_cloud_fraction_ar-
scl low_cloud_fraction_arscl_goodfraction" ;
       int qc_low_cloud_fraction_arscl(time, source_type) ;
```

```
            qc_low_cloud_fraction_arscl:long_name = "Quality check results on field: 
Frequency of 15-min cloud occurrence, as identified by ARSCL, be" ;
            qc_low_cloud_fraction_arscl:units = "unitless" ;
            qc_low_cloud_fraction_arscl:description = "This field contains bit packed 
integer values, where each bit represents a QC test on the data." ;
            qc_low_cloud_fraction_arscl:flag_method = "bit" ;
            qc_low_cloud_fraction_arscl:bit_1_description = "Calculated with valid 
points between 30% and 50%" ;
            qc_low_cloud_fraction_arscl:bit_2_description = "Calculated with valid 
points less than 30%" ;
            qc_low_cloud_fraction_arscl:bit_2_assessment = "Bad" ;
            qc_low_cloud_fraction_arscl:bit_1_assessment = "Indeterminate" ;
      float low_cloud_fraction_arscl_goodfraction(time, source_type) ;
            low_cloud_fraction_arscl_goodfraction:long_name = "Metric goodfraction 
for Frequency of 15-min cloud occurrence, as identified by ARSCL, below 5km" ;
            low_cloud_fraction_arscl_goodfraction:units = "unitless" ;
            low_cloud_fraction_arscl_goodfraction:missing_value = -9999.f ;
      float low_cloud_fraction_arscl_std(time, source_type) ;
            low_cloud_fraction_arscl_std:long_name = "Metric standard deviation for 
field Frequency of 15-min cloud occurrence, as identified by ARSC" ;
            low_cloud_fraction_arscl_std:units = "m" ;
            low_cloud_fraction_arscl_std:missing_value = -9999.f ;
      float cloud_fraction_tsi(time, source_type) ;
            cloud_fraction_tsi:long_name = "Hemispheric cloud fraction, as identified 
by TSI (opaque)" ;
            cloud_fraction_tsi:units = "1" ;
            cloud_fraction_tsi:missing_value = -9999.f ;
            cloud_fraction_tsi:valid_min = 0.f ;
            cloud_fraction_tsi:valid_max = 1.f ;
            cloud_fraction_tsi:cell_methods = "time: mean" ;
            cloud_fraction_tsi:ancillary_variables = "qc_cloud_fraction_tsi cloud_
fraction_tsi_goodfraction" ;
      int qc_cloud_fraction_tsi(time, source_type) ;
            qc_cloud_fraction_tsi:long_name = "Quality check results on field: Hemi-
spheric cloud fraction, as identified by TSI (opaque)" ;
            qc_cloud_fraction_tsi:units = "unitless" ;
            qc_cloud_fraction_tsi:description = "This field contains bit packed inte-
ger values, where each bit represents a QC test on the data." ;
            qc_cloud_fraction_tsi:flag_method = "bit" ;
            qc_cloud_fraction_tsi:bit_1_description = "Calculated with valid points 
between 30% and 50%" ;
            qc_cloud_fraction_tsi:bit_1_assessment = "Indeterminate" ;
            qc_cloud_fraction_tsi:bit_2_description = "Calculated with valid points 
less than 30%" ;
            qc_cloud_fraction_tsi:bit_2_assessment = "Bad" ;
      float cloud_fraction_tsi_goodfraction(time, source_type) ;
            cloud_fraction_tsi_goodfraction:long_name = "Metric goodfraction for 
Hemispheric cloud fraction, as identified by TSI (opaque)" ;
            cloud_fraction_tsi_goodfraction:units = "unitless" ;
            cloud_fraction_tsi_goodfraction:missing_value = -9999.f ;
      float cloud_fraction_tsi_std(time, source_type) ;
            cloud_fraction_tsi_std:long_name = "Metric standard deviation for field 
Hemispheric cloud fraction, as identified by TSI (opaque)" ;
            cloud_fraction_tsi_std:units = "1" ;
            cloud_fraction_tsi_std:missing_value = -9999.f ;
      float lwp(time, source_type) ;
            lwp:long_name = "Liquid water path" ;
            lwp:units = "g m-2" ;
            lwp:missing_value = -9999.f ;
            lwp:cell_methods = "time: mean" ;
```

```
            lwp:ancillary_variables = "qc_lwp lwp_goodfraction" ;
      int qc_lwp(time, source_type) ;
            qc_lwp:long_name = "Quality check results on field: Liquid water path" ;
            qc_lwp:units = "unitless" ;
            qc_lwp:description = "This field contains bit packed integer values, where
each bit represents a QC test on the data." ;
            qc_lwp:flag_method = "bit" ;
            qc_lwp:bit_1_description = "Calculated with valid points between 30% and
50%" ;
            qc_lwp:bit_1_assessment = "Indeterminate" ;
            qc_lwp:bit_2_description = "Calculated with valid points less than 30%" ;
            qc_lwp:bit_2_assessment = "Bad" ;
      float lwp_goodfraction(time, source_type) ;
            lwp_goodfraction:long_name = "Metric goodfraction for Liquid water path" ;
            lwp_goodfraction:units = "unitless" ;
            lwp_goodfraction:missing_value = -9999.f ;
      float lwp_std(time, source_type) ;
            lwp_std:long_name = "Metric standard deviation for field Liquid water
path" ;
            lwp_std:units = "g m-2" ;
            lwp_std:missing_value = -9999.f ;
      float lcl(time, source_type) ;
            lcl:long_name = "Lifting condensation level" ;
            lcl:units = "m" ;
            lcl:missing_value = -9999.f ;
            lcl:cell_methods = "time: mean" ;
            lcl:ancillary_variables = "qc_lcl lcl_goodfraction" ;
            lcl:standard_name = "atmosphere_lifting_condensation_level_wrt_surface" ;
      int qc_lcl(time, source_type) ;
            qc_lcl:long_name = "Quality check results on field: Lifting condensation
level" ;
            qc_lcl:units = "unitless" ;
            qc_lcl:description = "This field contains bit packed integer values, where
each bit represents a QC test on the data." ;
            qc_lcl:flag_method = "bit" ;
            qc_lcl:bit_1_description = "Calculated with valid points between 30% and
50%" ;
            qc_lcl:bit_1_assessment = "Indeterminate" ;
            qc_lcl:bit_2_description = "Calculated with valid points less than 30%" ;
            qc_lcl:bit_2_assessment = "Bad" ;
      float lcl_goodfraction(time, source_type) ;
            lcl_goodfraction:long_name = "Metric goodfraction for Lifting condensa-
tion level" ;
            lcl_goodfraction:units = "unitless" ;
            lcl_goodfraction:missing_value = -9999.f ;
      float lcl_std(time, source_type) ;
            lcl_std:long_name = "Metric standard deviation for field Lifting condensa-
tion level" ;
            lcl_std:units = "m" ;
            lcl_std:missing_value = -9999.f ;
      float lcl_domain(time, source_type) ;
            lcl_domain:long_name = "Lifting condensation level, domain" ;
            lcl_domain:units = "m" ;
            lcl_domain:missing_value = -9999.f ;
            lcl_domain:cell_methods = "time: mean" ;
            lcl_domain:ancillary_variables = "qc_lcl_domain lcl_domain_goodfraction" ;
            lcl_domain:standard_name = "atmosphere_lifting_condensation_level_wrt_
surface" ;
```

```
      int qc_lcl_domain(time, source_type) ;
            qc_lcl_domain:long_name = "Quality check results on field: Lifting conden-
sation level, domain" ;
            qc_lcl_domain:units = "unitless" ;
            qc_lcl_domain:description = "This field contains bit packed integer val-
ues, where each bit represents a QC test on the data." ;
            qc_lcl_domain:flag_method = "bit" ;
            qc_lcl_domain:bit_1_description = "Calculated with valid points between
30% and 50%" ;
            qc_lcl_domain:bit_1_assessment = "Indeterminate" ;
            qc_lcl_domain:bit_2_description = "Calculated with valid points less than
30%" ;
            qc_lcl_domain:bit_2_assessment = "Bad" ;
      float lcl_domain_goodfraction(time, source_type) ;
            lcl_domain_goodfraction:long_name = "Metric goodfraction for Lifting con-
densation level, domain" ;
            lcl_domain_goodfraction:units = "unitless" ;
            lcl_domain_goodfraction:missing_value = -9999.f ;
      float lcl_domain_std(time, source_type) ;
            lcl_domain_std:long_name = "Metric standard deviation for field Lifting
condensation level, domain" ;
            lcl_domain_std:units = "m" ;
            lcl_domain_std:missing_value = -9999.f ;
      float cloud_base_height(time, source_type) ;
            cloud_base_height:long_name = "Cloud base height" ;
            cloud_base_height:units = "m" ;
            cloud_base_height:missing_value = -9999.f ;
            cloud_base_height:cell_methods = "time: mean" ;
            cloud_base_height:ancillary_variables = "qc_cloud_base_height cloud_base_
height_goodfraction" ;
            cloud_base_height:standard_name = "convective_cloud_base_height" ;
      int qc_cloud_base_height(time, source_type) ;
            qc_cloud_base_height:long_name = "Quality check results on field: Cloud
base height" ;
            qc_cloud_base_height:units = "unitless" ;
            qc_cloud_base_height:description = "This field contains bit packed integer
values, where each bit represents a QC test on the data." ;
            qc_cloud_base_height:flag_method = "bit" ;
            qc_cloud_base_height:bit_1_description = "Calculated with valid points
between 30% and 50%" ;
            qc_cloud_base_height:bit_1_assessment = "Indeterminate" ;
            qc_cloud_base_height:bit_2_description = "Calculated with valid points
less than 30%" ;
            qc_cloud_base_height:bit_2_assessment = "Bad" ;
      float cloud_base_height_goodfraction(time, source_type) ;
            cloud_base_height_goodfraction:long_name = "Metric goodfraction for Cloud
base height" ;
            cloud_base_height_goodfraction:units = "unitless" ;
            cloud_base_height_goodfraction:missing_value = -9999.f ;
      float cloud_base_height_std(time, source_type) ;
            cloud_base_height_std:long_name = "Metric standard deviation for field
Cloud base height" ;
            cloud_base_height_std:units = "m" ;
            cloud_base_height_std:missing_value = -9999.f ;
      float water_vapor_mixing_ratio_surface(time, source_type) ;
            water_vapor_mixing_ratio_surface:long_name = "Water vapor mixing ratio at
surface" ;
            water_vapor_mixing_ratio_surface:units = "g kg-1" ;
            water_vapor_mixing_ratio_surface:missing_value = -9999.f ;
```

```
            water_vapor_mixing_ratio_surface:cell_methods = "time: mean" ;
            water_vapor_mixing_ratio_surface:ancillary_variables = "qc_water_vapor_
mixing_ratio_surface" ;
      int qc_water_vapor_mixing_ratio_surface(time, source_type) ;
            qc_water_vapor_mixing_ratio_surface:long_name = "Quality check results on
field: Water vapor mixing ratio at surface" ;
            qc_water_vapor_mixing_ratio_surface:units = "unitless" ;
            qc_water_vapor_mixing_ratio_surface:description = "This field contains bit
packed integer values, where each bit represents a QC test on the data." ;
            qc_water_vapor_mixing_ratio_surface:flag_method = "bit" ;
            qc_water_vapor_mixing_ratio_surface:bit_1_description = "Calculated with
valid points between 30% and 50%" ;
            qc_water_vapor_mixing_ratio_surface:bit_1_assessment = "Indeterminate" ;
            qc_water_vapor_mixing_ratio_surface:bit_2_description = "Calculated with
valid points less than 30%" ;
            qc_water_vapor_mixing_ratio_surface:bit_2_assessment = "Bad" ;
      float water_vapor_mixing_ratio_surface_std(time, source_type) ;
            water_vapor_mixing_ratio_surface_std:long_name = "Metric standard devia-
tion for field Water vapor mixing ratio at surface" ;
            water_vapor_mixing_ratio_surface_std:units = "g kg-1" ;
            water_vapor_mixing_ratio_surface_std:missing_value = -9999.f ;
      float water_vapor_mixing_ratio_surface_goodfraction(time, source_type) ;
            water_vapor_mixing_ratio_surface_goodfraction:long_name = "Metric good-
fraction for Water vapor mixing ratio at surface" ;
            water_vapor_mixing_ratio_surface_goodfraction:units = "unitless" ;
            water_vapor_mixing_ratio_surface_goodfraction:missing_value = -9999.f ;
      float temperature_surface(time, source_type) ;
            temperature_surface:long_name = "Temperature at surface" ;
            temperature_surface:units = "K" ;
            temperature_surface:missing_value = -9999.f ;
            temperature_surface:cell_methods = "time: mean" ;
            temperature_surface:ancillary_variables = "qc_temperature_surface" ;
            temperature_surface:standard_name = "surface_temperature" ;
      int qc_temperature_surface(time, source_type) ;
            qc_temperature_surface:long_name = "Quality check results on field: Tem-
perature at surface" ;
            qc_temperature_surface:units = "unitless" ;
            qc_temperature_surface:description = "This field contains bit packed inte-
ger values, where each bit represents a QC test on the data." ;
            qc_temperature_surface:flag_method = "bit" ;
            qc_temperature_surface:bit_1_description = "Calculated with valid points
between 30% and 50%" ;
            qc_temperature_surface:bit_1_assessment = "Indeterminate" ;
            qc_temperature_surface:bit_2_description = "Calculated with valid points
less than 30%" ;
            qc_temperature_surface:bit_2_assessment = "Bad" ;
      float temperature_surface_std(time, source_type) ;
            temperature_surface_std:long_name = "Metric standard deviation for field
Temperature at surface" ;
            temperature_surface_std:units = "K" ;
            temperature_surface_std:missing_value = -9999.f ;
      float temperature_surface_goodfraction(time, source_type) ;
            temperature_surface_goodfraction:long_name = "Metric goodfraction for
Temperature at surface" ;
            temperature_surface_goodfraction:units = "unitless" ;
            temperature_surface_goodfraction:missing_value = -9999.f ;
      float rh_surface(time, source_type) ;
            rh_surface:long_name = "Relative humidity" ;
            rh_surface:units = "%" ;
            rh_surface:missing_value = -9999.f ;
```

```
            rh_surface:cell_methods = "time: mean" ;
            rh_surface:ancillary_variables = "qc_rh_surface" ;
      int qc_rh_surface(time, source_type) ;
            qc_rh_surface:long_name = "Quality check results on field: Relative humid-
ity" ;
            qc_rh_surface:units = "unitless" ;
            qc_rh_surface:description = "This field contains bit packed integer val-
ues, where each bit represents a QC test on the data." ;
            qc_rh_surface:flag_method = "bit" ;
            qc_rh_surface:bit_1_description = "Calculated with valid points between
30% and 50%" ;
            qc_rh_surface:bit_1_assessment = "Indeterminate" ;
            qc_rh_surface:bit_2_description = "Calculated with valid points less than
30%" ;
            qc_rh_surface:bit_2_assessment = "Bad" ;
      float rh_surface_std(time, source_type) ;
            rh_surface_std:long_name = "Metric standard deviation for field Relative
humidity at surface" ;
            rh_surface_std:units = "%" ;
            rh_surface_std:missing_value = -9999.f ;
      float rh_surface_goodfraction(time, source_type) ;
            rh_surface_goodfraction:long_name = "Metric goodfraction for Relative hu-
midity at surface" ;
            rh_surface_goodfraction:units = "unitless" ;
            rh_surface_goodfraction:missing_value = -9999.f ;
      float water_vapor_mixing_ratio_boundary_layer(time, source_type) ;
            water_vapor_mixing_ratio_boundary_layer:long_name = "Water vapor mixing
ratio; average of boundary layer" ;
            water_vapor_mixing_ratio_boundary_layer:units = "g kg-1" ;
            water_vapor_mixing_ratio_boundary_layer:missing_value = -9999.f ;
            water_vapor_mixing_ratio_boundary_layer:cell_methods = "time: mean" ;
            water_vapor_mixing_ratio_boundary_layer:ancillary_variables = "qc_water_
vapor_mixing_ratio_boundary_layer" ;
      int qc_water_vapor_mixing_ratio_boundary_layer(time, source_type) ;
            qc_water_vapor_mixing_ratio_boundary_layer:long_name = "Quality check re-
sults on field: Water vapor mixing ratio; average of boundary layer" ;
            qc_water_vapor_mixing_ratio_boundary_layer:units = "unitless" ;
            qc_water_vapor_mixing_ratio_boundary_layer:description = "This field con-
tains bit packed integer values, where each bit represents a QC test on the data." ;
            qc_water_vapor_mixing_ratio_boundary_layer:flag_method = "bit" ;
            qc_water_vapor_mixing_ratio_boundary_layer:bit_1_description = "Calculat-
ed with valid points between 30% and 50%" ;
            qc_water_vapor_mixing_ratio_boundary_layer:bit_1_assessment = "Indetermi-
nate" ;
            qc_water_vapor_mixing_ratio_boundary_layer:bit_2_description = "Calculat-
ed with valid points less than 30%" ;
            qc_water_vapor_mixing_ratio_boundary_layer:bit_2_assessment = "Bad" ;
      float water_vapor_mixing_ratio_boundary_layer_std(time, source_type) ;
            water_vapor_mixing_ratio_boundary_layer_std:long_name = "Metric standard
deviation for field Water vapor mixing ratio: average of boundary layer" ;
            water_vapor_mixing_ratio_boundary_layer_std:units = "g kg-1" ;
            water_vapor_mixing_ratio_boundary_layer_std:missing_value = -9999.f ;
      float water_vapor_mixing_ratio_boundary_layer_goodfraction(time, source_type) ;
            water_vapor_mixing_ratio_boundary_layer_goodfraction:long_name = "Metric
goodfraction for Water vapor mixing ratio: average of boundary layer" ;
            water_vapor_mixing_ratio_boundary_layer_goodfraction:units = "unitless" ;
            water_vapor_mixing_ratio_boundary_layer_goodfraction:missing_value =
-9999.f ;
```

```
	float temperature_boundary_layer(time, source_type) ;
		temperature_boundary_layer:long_name = "Temperature; average of boundary 
layer" ;
		temperature_boundary_layer:units = "K" ;
		temperature_boundary_layer:missing_value = -9999.f ;
		temperature_boundary_layer:cell_methods = "time: mean" ;
		temperature_boundary_layer:ancillary_variables = "qc_temperature_bound-
ary_layer" ;
	int qc_temperature_boundary_layer(time, source_type) ;
		qc_temperature_boundary_layer:long_name = "Quality check results on field: 
Temperature; average of boundary layer" ;
		qc_temperature_boundary_layer:units = "unitless" ;
		qc_temperature_boundary_layer:description = "This field contains bit 
packed integer values, where each bit represents a QC test on the data." ;
		qc_temperature_boundary_layer:flag_method = "bit" ;
		qc_temperature_boundary_layer:bit_1_description = "Calculated with valid 
points between 30% and 50%" ;
		qc_temperature_boundary_layer:bit_1_assessment = "Indeterminate" ;
		qc_temperature_boundary_layer:bit_2_description = "Calculated with valid 
points less than 30%" ;
		qc_temperature_boundary_layer:bit_2_assessment = "Bad" ;
	float temperature_boundary_layer_std(time, source_type) ;
		temperature_boundary_layer_std:long_name = "Metric standard deviation for 
field Temperature: average of boundary layer" ;
		temperature_boundary_layer_std:units = "K" ;
		temperature_boundary_layer_std:missing_value = -9999.f ;
	float temperature_boundary_layer_goodfraction(time, source_type) ;
		temperature_boundary_layer_goodfraction:long_name = "Metric goodfraction 
for Temperature: average of boundary layer" ;
		temperature_boundary_layer_goodfraction:units = "unitless" ;
		temperature_boundary_layer_goodfraction:missing_value = -9999.f ;
	float rh_boundary_layer(time, source_type) ;
		rh_boundary_layer:long_name = "Relative humidity; average of boundary 
layer" ;
		rh_boundary_layer:units = "%" ;
		rh_boundary_layer:missing_value = -9999.f ;
		rh_boundary_layer:cell_methods = "time: mean" ;
		rh_boundary_layer:ancillary_variables = "qc_rh_boundary_layer" ;
	int qc_rh_boundary_layer(time, source_type) ;
		qc_rh_boundary_layer:long_name = "Quality check results on field: Relative 
humidity; average of boundary layer" ;
		qc_rh_boundary_layer:units = "unitless" ;
		qc_rh_boundary_layer:description = "This field contains bit packed integer 
values, where each bit represents a QC test on the data." ;
		qc_rh_boundary_layer:flag_method = "bit" ;
		qc_rh_boundary_layer:bit_1_description = "Calculated with valid points 
between 30% and 50%" ;
		qc_rh_boundary_layer:bit_1_assessment = "Indeterminate" ;
		qc_rh_boundary_layer:bit_2_description = "Calculated with valid points 
less than 30%" ;
		qc_rh_boundary_layer:bit_2_assessment = "Bad" ;
	float rh_boundary_layer_std(time, source_type) ;
		rh_boundary_layer_std:long_name = "Metric standard deviation for field 
Relative humidity: average of boundary layer" ;
		rh_boundary_layer_std:units = "%" ;
		rh_boundary_layer_std:missing_value = -9999.f ;
	float rh_boundary_layer_goodfraction(time, source_type) ;
		rh_boundary_layer_goodfraction:long_name = "Metric goodfraction for Rela-
```

```
tive humidity: average of boundary layer” ;
             rh_boundary_layer_goodfraction:units = “unitless” ;
             rh_boundary_layer_goodfraction:missing_value = -9999.f ;
      float lat ;
             lat:long_name = “North latitude” ;
             lat:units = “degree_N” ;
             lat:valid_min = -90.f ;
             lat:valid_max = 90.f ;
             lat:standard_name = “latitude” ;
      float lon ;
             lon:long_name = “East longitude” ;
             lon:units = “degree_E” ;
             lon:valid_min = -180.f ;
             lon:valid_max = 180.f ;
             lon:standard_name = “longitude” ;
      float alt ;
             alt:long_name = “Altitude above mean sea level” ;
             alt:units = “m” ;
             alt:standard_name = “altitude” ;

// global attributes:
             :process_version = “Alpha 2 Release” ;
             :dod_version = “lassodiagobsmod.m1-2.0.hdr” ;
             :input_source = “sgplassodiagmod1C1.m1.20160610.000000.nc,sgplassodiagob-
sC1.c1.20160610.000000.nc” ;
             :site_id = “sgp” ;
             :platform_id = “lassodiagobsmod” ;
             :facility_id = “C1” ;
             :data_level = “m1” ;
             :location_description = “Southern Great Plains (SGP), Lamont, Oklahoma” ;
             :datastream = “sgplassodiagobsmod1C1.m1” ;
             :simulation_id_number = “1” ;
             :simulation_origin_host = “cumulus-login1” ;
             :simulation_origin_directory = “/gpfs/wolf/cli120/proj-shared/alpha2/to-
process/wrf/alpha2runs/runlas20160610a2mddhd20/” ;
             :simulation_origin_run_date_time = “2017-07-06 16:43:42” ;
             :model_type = “WRF” ;
             :model_version = “3.8.1” ;
             :model_github_hash = “880eb996a632958d7bd91cd03839373a5eeb2a18” ;
             :output_domain_size = “14.4 km” ;
             :output_number_of_levels = “226” ;
             :output_horizontal_grid_spacing = “100 m” ;
             :output_start_datetime = “20160610.120000 UTC” ;
             :config_large_scale_forcing = “ECMWF” ;
             :config_large_scale_forcing_scale = “16 km” ;
             :config_large_scale_forcing_specifics = “ecmwf_lasso_ddh (v20170213)” ;
             :config_surface_treatment = “VARANAL” ;
             :config_surface_treatment_specifics = “sgp60varanarapC1.c1 (v20161129)” ;
             :config_initial_condition = “Sounding” ;
             :config_initial_condition_specifics = “sgpsondewnpnC1” ;
             :config_aerosol = “Constant” ;
             :config_forecast_time = “15 h” ;
             :config_boundary_method = “Periodic” ;
             :config_microphysics = “LASSO Morrison” ;
             :config_physics_suite = “1” ;
             :config_nickname = “runlas20160610a2mddhd20” ;
             :config_comments = “Alpha 2” ;
             :boundary_layer_top = “1321 m AGL” ;
             :boundary_layer_bottom = “1121 m AGL” ;
```

```
             :lcl_domain_sites = "C1, E32, E33, E36, E37, E39, E41, BLAC, BREC, CARL,
MRSH, MEDF, REDR" ;
             :contacts = "William.Gustafson@pnnl.gov, Vogelmann@bnl.gov" ;
             :doi = "10.5439/1342961" ;
             :history = "Thu Aug 24 19:43:34 2017" ;
}
```

File: .../20160610/sim0001/obs_model/sgplassodiagobsmod2d1C1.m1.20160610.000000.nc

Description: Model output and observations for time-height cloud fraction on comparable 10-minute sampling intervals.

```
netcdf sgplassodiagobsmod2d1C1.m1.20160610.000000 {
dimensions:
       time = UNLIMITED ; // (91 currently)
       height = 226 ;
variables:
       int base_time ;
             base_time:long_name = "Base time in Epoch" ;
             base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
             base_time:ancillary_variables = "time_offset" ;
       double time_offset(time) ;
             time_offset:long_name = "Time offset from base_time" ;
             time_offset:units = "seconds since 2016-06-10 12:00:00 0:00" ;
             time_offset:ancillary_variables = "base_time" ;
       double time(time) ;
             time:long_name = "Time offset from midnight" ;
             time:units = "seconds since 2016-06-10 00:00:00 0:00" ;
             time:calendar = "gregorian" ;
             time:standard_name = "time" ;
       float height(time, height) ;
             height:long_name = "Height above ground level" ;
             height:units = "km" ;
             height:standard_name = "height" ;
       float cloud_fraction(time, height) ;
             cloud_fraction:long_name = "Cloud fraction (from simulation)" ;
             cloud_fraction:units = "1" ;            cloud_fraction:missing_value =
-9999.f ;
             cloud_fraction:valid_min = 0.f ;
             cloud_fraction:valid_max = 1.f ;
       float cloud_fraction_mask_arscl(time, height) ;
             cloud_fraction_mask_arscl:long_name = "Cloud fraction mask (from ARSCL)" ;
             cloud_fraction_mask_arscl:units = "unitless" ;
             cloud_fraction_mask_arscl:description = "This field contains integer val-
ues which should be interpreted as listed." ;
             cloud_fraction_mask_arscl:flag_method = "integer" ;
             cloud_fraction_mask_arscl:flag_0_description = "no cloud" ;
             cloud_fraction_mask_arscl:flag_1_description = "cloud" ;
             cloud_fraction_mask_arscl:flag_2_description = "no data available" ;
       float lat ;
             lat:long_name = "North latitude" ;
             lat:units = "degree_N" ;
             lat:valid_min = -90.f ;
             lat:valid_max = 90.f ;
             lat:standard_name = "latitude" ;
       float lon ;
             lon:long_name = "East longitude" ;
             lon:units = "degree_E" ;
             lon:valid_min = -180.f ;
```

```
              lon:valid_max = 180.f ;
              lon:standard_name = "longitude" ;
       float alt ;
              alt:long_name = "Altitude above mean sea level" ;
              alt:units = "m" ;
              alt:standard_name = "altitude" ;

// global attributes:
              :process_version = "Alpha 2 Release" ;
              :dod_version = "lassodiagobsmod2d.m1-2.0.hdr" ;
              :input_source = "sgplassomod2d1C1.m1.20160610.000000.nc,sgplassomod1C1.
m1.20160610.000000.nc,sgpcloudfraction15mC1.c1.20160610.000030.nc,sgpcloudfraction-
15mC1.c1.20160611.001530.nc" ;
              :site_id = "sgp" ;
              :platform_id = "lassodiagobsmod2d" ;
              :facility_id = "C1" ;
              :data_level = "m1" ;
              :location_description = "Southern Great Plains (SGP), Lamont, Oklahoma" ;
              :datastream = "sgplassodiagobsmod2d1C1.m1" ;
              :simulation_id_number = "1" ;
              :simulation_origin_host = "cumulus-login1" ;
              :simulation_origin_directory = "/gpfs/wolf/cli120/proj-shared/alpha2/to-
process/wrf/alpha2runs/runlas20160610a2mddhd20/" ;
              :simulation_origin_run_date_time = "2017-07-06 16:43:42" ;
              :model_type = "WRF" ;
              :model_version = "3.8.1" ;
              :model_github_hash = "880eb996a632958d7bd91cd03839373a5eeb2a18" ;
              :output_domain_size = "14.4 km" ;
              :output_number_of_levels = "226" ;
              :output_horizontal_grid_spacing = "100 m" ;
              :output_start_datetime = "20160610.120000 UTC" ;
              :config_large_scale_forcing = "ECMWF" ;
              :config_large_scale_forcing_scale = "16 km" ;
              :config_large_scale_forcing_specifics = "ecmwf_lasso_ddh (v20170213)" ;
              :config_surface_treatment = "VARANAL" ;
              :config_surface_treatment_specifics = "sgp60varanarapC1.c1 (v20161129)" ;
              :config_initial_condition = "Sounding" ;
              :config_initial_condition_specifics = "sgpsondewnpnC1" ;
              :config_aerosol = "Constant" ;
              :config_forecast_time = "15 h" ;
              :config_boundary_method = "Periodic" ;
              :config_microphysics = "LASSO Morrison" ;
              :config_physics_suite = "1" ;
              :config_nickname = "runlas20160610a2mddhd20" ;
              :config_comments = "Alpha 2" ;
              :contacts = "William.Gustafson@pnnl.gov, Vogelmann@bnl.gov" ;
              :doi = "10.5439/1342961" ;
              :history = "Thu Aug 24 20:10:39 2017" ;
}
```

File: .../20160610/sim0001/obs_model/sgplassomod1C1.m1.20160610.000000.nc

Description: Model output for domain-averaged profiles at 10-minute intervals for variables used in the metrics.

```
netcdf sgplassomod1C1.m1.20160610.000000 {
dimensions:
       time = UNLIMITED ; // (91 currently)
```

```
variables:
        int base_time ;
                base_time:long_name = "Base time in Epoch" ;
                base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
                base_time:ancillary_variables = "time_offset" ;
        double time_offset(time) ;
                time_offset:long_name = "Time offset from base_time" ;
                time_offset:units = "seconds since 2016-06-10 12:00:00 0:00" ;
                time_offset:ancillary_variables = "base_time" ;
        double time(time) ;
                time:long_name = "Time offset from midnight" ;
                time:units = "seconds since 2016-06-10 00:00:00 0:00" ;
                time:calendar = "gregorian" ;
                time:standard_name = "time" ;
        float low_cloud_fraction_arscl(time) ;
                low_cloud_fraction_arscl:long_name = "Frequency of 15-min cloud occur-
rence, as identified by ARSCL, below 5km AGL" ;
                low_cloud_fraction_arscl:units = "1" ;
                low_cloud_fraction_arscl:missing_value = -9999.f ;
                low_cloud_fraction_arscl:valid_min = 0.f ;
                low_cloud_fraction_arscl:valid_max = 1.f ;
        float cloud_fraction_tsi(time) ;
                cloud_fraction_tsi:long_name = "Hemispheric cloud fraction, as identified
by TSI (opaque)" ;
                cloud_fraction_tsi:units = "1" ;
                cloud_fraction_tsi:missing_value = -9999.f ;
                cloud_fraction_tsi:valid_min = 0.f ;
                cloud_fraction_tsi:valid_max = 1.f ;
        float lwp(time) ;
                lwp:long_name = "Liquid water path" ;
                lwp:units = "g m-2" ;
                lwp:missing_value = -9999.f ;
        float lcl(time) ;
                lcl:long_name = "Lifting condensation level" ;
                lcl:units = "m" ;
                lcl:missing_value = -9999.f ;
                lcl:standard_name = "atmosphere_lifting_condensation_level_wrt_surface" ;
        float water_vapor_mixing_ratio_surface(time) ;
                water_vapor_mixing_ratio_surface:long_name = "Water vapor mixing ratio at
surface" ;
                water_vapor_mixing_ratio_surface:units = "g kg-1" ;
                water_vapor_mixing_ratio_surface:missing_value = -9999.f ;
        float temperature_surface(time) ;
                temperature_surface:long_name = "Temperature at surface" ;
                temperature_surface:units = "K" ;
                temperature_surface:missing_value = -9999.f ;
                temperature_surface:standard_name = "surface_temperature" ;
        float pressure_surface(time) ;
                pressure_surface:long_name = "Pressure at surface" ;
                pressure_surface:units = "kPa" ;
                pressure_surface:missing_value = -9999.f ;
                pressure_surface:standard_name = "surface_air_pressure" ;
        float rh_surface(time) ;
                rh_surface:long_name = "Relative humidity" ;
                rh_surface:units = "%" ;
                rh_surface:missing_value = -9999.f ;
        float water_vapor_mixing_ratio_boundary_layer(time) ;
                water_vapor_mixing_ratio_boundary_layer:long_name = "Water vapor mixing
ratio; average of boundary layer" ;
                water_vapor_mixing_ratio_boundary_layer:units = "g kg-1" ;
```

```
             water_vapor_mixing_ratio_boundary_layer:missing_value = -9999.f ;
      float temperature_boundary_layer(time) ;
             temperature_boundary_layer:long_name = "Temperature; average of boundary
layer" ;
             temperature_boundary_layer:units = "K" ;
             temperature_boundary_layer:missing_value = -9999.f ;
      float pressure_boundary_layer(time) ;
             pressure_boundary_layer:long_name = "Pressure; average of boundary layer" ;
             pressure_boundary_layer:units = "kPa" ;
             pressure_boundary_layer:missing_value = -9999.f ;
      float rh_boundary_layer(time) ;
             rh_boundary_layer:long_name = "Relative humidity; average of boundary
layer" ;
             rh_boundary_layer:units = "%" ;
             rh_boundary_layer:missing_value = -9999.f ;
      float lat ;
             lat:long_name = "North latitude" ;
             lat:units = "degree_N" ;
             lat:valid_min = -90.f ;
             lat:valid_max = 90.f ;
             lat:standard_name = "latitude" ;
      float lon ;
             lon:long_name = "East longitude" ;
             lon:units = "degree_E" ;
             lon:valid_min = -180.f ;
             lon:valid_max = 180.f ;
             lon:standard_name = "longitude" ;
      float alt ;
             alt:long_name = "Altitude above mean sea level" ;
             alt:units = "m" ;
             alt:standard_name = "altitude" ;

// global attributes:
             :process_version = "Alpha 2 Release" ;
             :dod_version = "lassomod.m1-2.0.hdr" ;
             :site_id = "sgp" ;
             :platform_id = "lassomod" ;
             :facility_id = "C1" ;
             :data_level = "m1" ;
             :location_description = "Southern Great Plains (SGP), Lamont, Oklahoma" ;
             :datastream = "sgplassomod1C1.m1" ;
             :simulation_id_number = "1" ;
             :simulation_origin_host = "cumulus-login1" ;
             :simulation_origin_directory = "/gpfs/wolf/cli120/proj-shared/alpha2/to-
process/wrf/alpha2runs/runlas20160610a2mddhd20/" ;
             :simulation_origin_run_date_time = "2017-07-06 16:43:42" ;
             :model_type = "WRF" ;
             :model_version = "3.8.1" ;
             :model_github_hash = "880eb996a632958d7bd91cd03839373a5eeb2a18" ;
             :output_domain_size = "14.4 km" ;
             :output_number_of_levels = "226" ;
             :output_horizontal_grid_spacing = "100 m" ;
             :output_start_datetime = "20160610.120000 UTC" ;
             :config_large_scale_forcing = "ECMWF" ;
             :config_large_scale_forcing_scale = "16 km" ;
             :config_large_scale_forcing_specifics = "ecmwf_lasso_ddh (v20170213)" ;
             :config_surface_treatment = "VARANAL" ;
             :config_surface_treatment_specifics = "sgp60varanarapC1.c1 (v20161129)" ;
             :config_initial_condition = "Sounding" ;
             :config_initial_condition_specifics = "sgpsondewnpnC1" ;
```

```
                :config_aerosol = "Constant" ;
                :config_forecast_time = "15 h" ;
                :config_boundary_method = "Periodic" ;
                :config_microphysics = "LASSO Morrison" ;
                :config_physics_suite = "1" ;
                :config_nickname = "runlas20160610a2mddhd20" ;
                :config_comments = "Alpha 2" ;
                :boundary_layer_top = "1321 m AGL" ;
                :boundary_layer_bottom = "1121 m AGL" ;
                :contacts = "William.Gustafson@pnnl.gov, Vogelmann@bnl.gov" ;
                :doi = "10.5439/1342961" ;
                :history = "Sat Jul 29 16:47:10 2017" ;
}
```

File: .../20160610/sim0001/obs_model/ sgplassomod2d1C1.m1.20160610.000000.nc

Description: Time-height series of cloud fraction, potential temperature, water vapor mixing ratio, and pressure from the model for a 10-minute sampling interval.

```
netcdf sgplassomod2d1C1.m1.20160610.000000 {
dimensions:
        time = UNLIMITED ; // (91 currently)
        height = 226 ;
variables:
        int base_time ;
                base_time:long_name = "Base time in Epoch" ;
                base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
                base_time:ancillary_variables = "time_offset" ;
        double time_offset(time) ;
                time_offset:long_name = "Time offset from base_time" ;
                time_offset:units = "seconds since 2016-06-10 12:00:00 0:00" ;
                time_offset:ancillary_variables = "base_time" ;
        double time(time) ;
                time:long_name = "Time offset from midnight" ;
                time:units = "seconds since 2016-06-10 00:00:00 0:00" ;
                time:calendar = "gregorian" ;
                time:standard_name = "time" ;
        float height(time, height) ;
                height:long_name = "Height above ground level" ;
                height:units = "km" ;
                height:standard_name = "height" ;
        float cloud_fraction(time, height) ;
                cloud_fraction:long_name = "Cloud fraction" ;
                cloud_fraction:units = "1" ;
                cloud_fraction:missing_value = -9999.f ;
                cloud_fraction:valid_min = "0.f" ;
                cloud_fraction:valid_max = "1.f" ;
                cloud_fraction:cell_methods = "area: mean time: point height: point" ;
                cloud_fraction:standard_name = "cloud_area_fraction" ;
        float potential_temp(time, height) ;
                potential_temp:long_name = "Potential temperature" ;
                potential_temp:units = "K" ;
                potential_temp:missing_value = -9999.f ;
                potential_temp:cell_methods = "area: mean time: point height: point" ;
                potential_temp:standard_name = "air_potential_temperature" ;
        float water_vapor_mixing_ratio(time, height) ;
                water_vapor_mixing_ratio:long_name = "Water vapor mixing ratio" ;
                water_vapor_mixing_ratio:units = "g kg-1" ;
                water_vapor_mixing_ratio:missing_value = -9999.f ;
                water_vapor_mixing_ratio:cell_methods = "area: mean time: point height:
point" ;
```

```
        float bar_pres(time, height) ;
                bar_pres:long_name = "Barometric pressure" ;
                bar_pres:units = "kPa" ;
                bar_pres:standard_name = "air_pressure" ;
                bar_pres:valid_min = "0.f" ;
                bar_pres:valid_max = "110.f" ;
                bar_pres:missing_value = -9999.f ;
                bar_pres:cell_methods = "area: mean time: point height: point" ;
        float lat ;
                lat:long_name = "North latitude" ;
                lat:units = "degree_N" ;
                lat:valid_min = -90.f ;
                lat:valid_max = 90.f ;
                lat:standard_name = "latitude" ;
        float lon ;
                lon:long_name = "East longitude" ;
                lon:units = "degree_E" ;
                lon:valid_min = -180.f ;
                lon:valid_max = 180.f ;
                lon:standard_name = "longitude" ;
        float alt ;
                alt:long_name = "Altitude above mean sea level" ;
                alt:units = "m" ;
                alt:standard_name = "altitude" ;

// global attributes:
                :process_version = "Alpha 2 Release" ;
                :dod_version = "lassomod2d.m1-2.0.hdr" ;
                :site_id = "sgp" ;
                :platform_id = "lassomod" ;
                :facility_id = "C1" ;
                :data_level = "m1" ;
                :location_description = "Southern Great Plains (SGP), Lamont, Oklahoma" ;
                :datastream = "sgplassomod2d1C1.m1" ;
                :simulation_id_number = "1" ;
                :simulation_origin_host = "cumulus-login1" ;
                :simulation_origin_directory = "/gpfs/wolf/cli120/proj-shared/alpha2/to-
process/wrf/alpha2runs/runlas20160610a2mddhd20/" ;
                :simulation_origin_run_date_time = "2017-07-06 16:43:42" ;
                :model_type = "WRF" ;
                :model_version = "3.8.1" ;
                :model_github_hash = "880eb996a632958d7bd91cd03839373a5eeb2a18" ;
                :output_domain_size = "14.4 km" ;
                :output_number_of_levels = "226" ;
                :output_horizontal_grid_spacing = "100 m" ;
                :output_start_datetime = "20160610.120000 UTC" ;
                :config_large_scale_forcing = "ECMWF" ;
                :config_large_scale_forcing_scale = "16 km" ;
                :config_large_scale_forcing_specifics = "ecmwf_lasso_ddh (v20170213)" ;
                :config_surface_treatment = "VARANAL" ;
                :config_surface_treatment_specifics = "sgp60varanarapC1.c1 (v20161129)" ;
                :config_initial_condition = "Sounding" ;
                :config_initial_condition_specifics = "sgpsondewnpnC1" ;
                :config_aerosol = "Constant" ;
                :config_forecast_time = "15 h" ;
                :config_boundary_method = "Periodic" ;
                :config_microphysics = "LASSO Morrison" ;
                :config_physics_suite = "1" ;
                :config_nickname = "runlas20160610a2mddhd20" ;
                :config_comments = "Alpha 2" ;
```

```
                :boundary_layer_top = "1321 m AGL" ;
                :boundary_layer_bottom = "1121 m AGL" ;
                :contacts = "William.Gustafson@pnnl.gov, Vogelmann@bnl.gov" ;
                :doi = "10.5439/1342961" ;
                :history = "Sat Jul 29 16:47:11 2017" ;
}
```

File: .../20160610/sim0001/obs_model/sgplassomodxy1C1.m1.20160610.000000.nc

Description: Horizontal, two-dimensional time series of cloud base height from the model.

```
netcdf sgplassomodxy1C1.m1.20160610.000000 {
dimensions:
        time = UNLIMITED ; // (91 currently)
        south_north = 144 ;
        west_east = 144 ;
variables:
        int base_time ;
                base_time:long_name = "Base time in Epoch" ;
                base_time:units = "seconds since 1970-1-1 0:00:00 0:00" ;
                base_time:ancillary_variables = "time_offset" ;
        double time_offset(time) ;
                time_offset:long_name = "Time offset from base_time" ;
                time_offset:units = "seconds since 2016-06-10 12:00:00 0:00" ;
                time_offset:ancillary_variables = "base_time" ;
        double time(time) ;
                time:long_name = "Time offset from midnight" ;
                time:calendar = "gregorian" ;
                time:standard_name = "time" ;
        float west_east(west_east) ;
                west_east:long_name = "Coordinate variable for west_east" ;
                west_east:units = "m" ;
        float south_north(south_north) ;
                south_north:long_name = "Coordinate variable for south_north" ;
                south_north:units = "m" ;
        float cloud_base_height(time, south_north, west_east) ;
                cloud_base_height:long_name = "Cloud base height" ;
                cloud_base_height:units = "m" ;
                cloud_base_height:missing_value = -9999.f ;
                cloud_base_height:standard_name = "convective_cloud_base_height" ;
        float lat ;
                lat:long_name = "North latitude" ;
                lat:units = "degree_N" ;
                lat:valid_min = -90.f ;
                lat:valid_max = 90.f ;
                lat:standard_name = "latitude" ;
        float lon ;
                lon:long_name = "East longitude" ;
                lon:units = "degree_E" ;
                lon:valid_min = -180.f ;
                lon:valid_max = 180.f ;
                lon:standard_name = "longitude" ;
        float alt ;
                alt:long_name = "Altitude above mean sea level" ;
                alt:units = "m" ;
                alt:standard_name = "altitude" ;

// global attributes:
                :process_version = "Alpha 2 Release" ;
                :dod_version = "lassomodxy.m1-2.0.hdr" ;
```

```
              :site_id = "sgp" ;
              :platform_id = "lassomod" ;
              :facility_id = "C1" ;
              :data_level = "m1" ;
              :location_description = "Southern Great Plains (SGP), Lamont, Oklahoma" ;
              :datastream = "sgplassomodxy1C1.m1" ;
              :simulation_id_number = "1" ;
              :simulation_origin_host = "cumulus-login1" ;
              :simulation_origin_directory = "/gpfs/wolf/cli120/proj-shared/alpha2/to-
process/wrf/alpha2runs/runlas20160610a2mddhd20/" ;
              :simulation_origin_run_date_time = "2017-07-06 16:43:42" ;
              :model_type = "WRF" ;
              :model_version = "3.8.1" ;
              :model_github_hash = "880eb996a632958d7bd91cd03839373a5eeb2a18" ;
              :output_domain_size = "14.4 km" ;
              :output_number_of_levels = "226" ;
              :output_horizontal_grid_spacing = "100 m" ;
              :output_start_datetime = "20160610.120000 UTC" ;
              :config_large_scale_forcing = "ECMWF" ;
              :config_large_scale_forcing_scale = "16 km" ;
              :config_large_scale_forcing_specifics = "ecmwf_lasso_ddh (v20170213)" ;
              :config_surface_treatment = "VARANAL" ;
              :config_surface_treatment_specifics = "sgp60varanarapC1.c1 (v20161129)" ;
              :config_initial_condition = "Sounding" ;
              :config_initial_condition_specifics = "sgpsondewnpnC1" ;
              :config_aerosol = "Constant" ;
              :config_forecast_time = "15 h" ;
              :config_boundary_method = "Periodic" ;
              :config_microphysics = "LASSO Morrison" ;
              :config_physics_suite = "1" ;
              :config_nickname = "runlas20160610a2mddhd20" ;
              :config_comments = "Alpha 2" ;
              :boundary_layer_top = "1321 m AGL" ;
              :boundary_layer_bottom = "1121 m AGL" ;
              :contacts = "William.Gustafson@pnnl.gov, Vogelmann@bnl.gov" ;
              :doi = "10.5439/1342961" ;
              :history = "Sat Jul 29 16:47:11 2017" ;
}
```

File: .../20160610/sim0001/raw_model/wrfout_d01_2016-06-10_12:00:00.nc

Description: WRF output with instantaneous variables for the full domain. Output is every 10-minutes with six output times per file, i.e., the outputs for one hour.

```
netcdf wrfout_d01_2016-06-10_12\:00\:00 {
dimensions:
       Time = UNLIMITED ; // (6 currently)
       DateStrLen = 19 ;
       west_east = 144 ;
       south_north = 144 ;
       bottom_top = 226 ;
       bottom_top_stag = 227 ;
       soil_layers_stag = 5 ;
       west_east_stag = 145 ;
       south_north_stag = 145 ;
       force_layers = 751 ;
       aer_layers = 321 ;
variables:
       char Times(Time, DateStrLen) ;
```

```
float XLAT(Time, south_north, west_east) ;
       XLAT:FieldType = 104 ;
       XLAT:MemoryOrder = "XY " ;
       XLAT:description = "LATITUDE, SOUTH IS NEGATIVE" ;
       XLAT:units = "degree_north" ;
       XLAT:stagger = "" ;
       XLAT:coordinates = "XLONG XLAT" ;
float XLONG(Time, south_north, west_east) ;
       XLONG:FieldType = 104 ;
       XLONG:MemoryOrder = "XY " ;
       XLONG:description = "LONGITUDE, WEST IS NEGATIVE" ;
       XLONG:units = "degree_east" ;
       XLONG:stagger = "" ;
       XLONG:coordinates = "XLONG XLAT" ;
float LU_INDEX(Time, south_north, west_east) ;
       LU_INDEX:FieldType = 104 ;
       LU_INDEX:MemoryOrder = "XY " ;
       LU_INDEX:description = "LAND USE CATEGORY" ;
       LU_INDEX:units = "" ;
       LU_INDEX:stagger = "" ;
       LU_INDEX:coordinates = "XLONG XLAT XTIME" ;
float ZNU(Time, bottom_top) ;
       ZNU:FieldType = 104 ;
       ZNU:MemoryOrder = "Z  " ;
       ZNU:description = "eta values on half (mass) levels" ;
       ZNU:units = "" ;
       ZNU:stagger = "" ;
float ZNW(Time, bottom_top_stag) ;
       ZNW:FieldType = 104 ;
       ZNW:MemoryOrder = "Z  " ;
       ZNW:description = "eta values on full (w) levels" ;
       ZNW:units = "" ;
       ZNW:stagger = "Z" ;
float ZS(Time, soil_layers_stag) ;
       ZS:FieldType = 104 ;
       ZS:MemoryOrder = "Z  " ;
       ZS:description = "DEPTHS OF CENTERS OF SOIL LAYERS" ;
       ZS:units = "m" ;
       ZS:stagger = "Z" ;
float DZS(Time, soil_layers_stag) ;
       DZS:FieldType = 104 ;
       DZS:MemoryOrder = "Z  " ;
       DZS:description = "THICKNESSES OF SOIL LAYERS" ;
       DZS:units = "m" ;
       DZS:stagger = "Z" ;
float VAR_SSO(Time, south_north, west_east) ;
       VAR_SSO:FieldType = 104 ;
       VAR_SSO:MemoryOrder = "XY " ;
       VAR_SSO:description = "variance of subgrid-scale orography" ;
       VAR_SSO:units = "m2" ;
       VAR_SSO:stagger = "" ;
       VAR_SSO:coordinates = "XLONG XLAT XTIME" ;
float U(Time, bottom_top, south_north, west_east_stag) ;
       U:FieldType = 104 ;
       U:MemoryOrder = "XYZ" ;
       U:description = "x-wind component" ;
       U:units = "m s-1" ;
       U:stagger = "X" ;
       U:coordinates = "XLONG_U XLAT_U XTIME" ;
```

```
float V(Time, bottom_top, south_north_stag, west_east) ;
       V:FieldType = 104 ;
       V:MemoryOrder = "XYZ" ;
       V:description = "y-wind component" ;
       V:units = "m s-1" ;
       V:stagger = "Y" ;
       V:coordinates = "XLONG_V XLAT_V XTIME" ;
float W(Time, bottom_top_stag, south_north, west_east) ;
       W:FieldType = 104 ;
       W:MemoryOrder = "XYZ" ;
       W:description = "z-wind component" ;
       W:units = "m s-1" ;
       W:stagger = "Z" ;
       W:coordinates = "XLONG XLAT XTIME" ;
float PH(Time, bottom_top_stag, south_north, west_east) ;
       PH:FieldType = 104 ;
       PH:MemoryOrder = "XYZ" ;
       PH:description = "perturbation geopotential" ;
       PH:units = "m2 s-2" ;
       PH:stagger = "Z" ;
       PH:coordinates = "XLONG XLAT XTIME" ;
float PHB(Time, bottom_top_stag, south_north, west_east) ;
       PHB:FieldType = 104 ;
       PHB:MemoryOrder = "XYZ" ;
       PHB:description = "base-state geopotential" ;
       PHB:units = "m2 s-2" ;
       PHB:stagger = "Z" ;
       PHB:coordinates = "XLONG XLAT XTIME" ;
float T(Time, bottom_top, south_north, west_east) ;
       T:FieldType = 104 ;
       T:MemoryOrder = "XYZ" ;
       T:description = "perturbation potential temperature (theta-t0)" ;
       T:units = "K" ;
       T:stagger = "" ;
       T:coordinates = "XLONG XLAT XTIME" ;
float HFX_FORCE(Time) ;
       HFX_FORCE:FieldType = 104 ;
       HFX_FORCE:MemoryOrder = "0  " ;
       HFX_FORCE:description = "SCM ideal surface sensible heat flux" ;
       HFX_FORCE:units = "W m-2" ;
       HFX_FORCE:stagger = "" ;
float LH_FORCE(Time) ;
       LH_FORCE:FieldType = 104 ;
       LH_FORCE:MemoryOrder = "0  " ;
       LH_FORCE:description = "SCM ideal surface latent heat flux" ;
       LH_FORCE:units = "W m-2" ;
       LH_FORCE:stagger = "" ;
float TSK_FORCE(Time) ;
       TSK_FORCE:FieldType = 104 ;
       TSK_FORCE:MemoryOrder = "0  " ;
       TSK_FORCE:description = "SCM ideal surface skin temperature" ;
       TSK_FORCE:units = "W m-2" ;
       TSK_FORCE:stagger = "" ;
float HFX_FORCE_TEND(Time) ;
       HFX_FORCE_TEND:FieldType = 104 ;
       HFX_FORCE_TEND:MemoryOrder = "0  " ;
       HFX_FORCE_TEND:description = "SCM ideal surface sensible heat flux tenden-
cy" ;
       HFX_FORCE_TEND:units = "W m-2 s-1" ;
       HFX_FORCE_TEND:stagger = "" ;
```

```
	float LH_FORCE_TEND(Time) ;
		LH_FORCE_TEND:FieldType = 104 ;
		LH_FORCE_TEND:MemoryOrder = “0  “ ;
		LH_FORCE_TEND:description = “SCM ideal surface latent heat flux tendency” 
;
		LH_FORCE_TEND:units = “W m-2 s-1” ;
		LH_FORCE_TEND:stagger = “” ;
	float TSK_FORCE_TEND(Time) ;
		TSK_FORCE_TEND:FieldType = 104 ;
		TSK_FORCE_TEND:MemoryOrder = “0  “ ;
		TSK_FORCE_TEND:description = “SCM ideal surface skin temperature tenden-
cy” ;
		TSK_FORCE_TEND:units = “W m-2 s-1” ;
		TSK_FORCE_TEND:stagger = “” ;
	float MU(Time, south_north, west_east) ;
		MU:FieldType = 104 ;
		MU:MemoryOrder = “XY “ ;
		MU:description = “perturbation dry air mass in column” ;
		MU:units = “Pa” ;
		MU:stagger = “” ;
		MU:coordinates = “XLONG XLAT XTIME” ;
	float MUB(Time, south_north, west_east) ;
		MUB:FieldType = 104 ;
		MUB:MemoryOrder = “XY “ ;
		MUB:description = “base state dry air mass in column” ;
		MUB:units = “Pa” ;
		MUB:stagger = “” ;
		MUB:coordinates = “XLONG XLAT XTIME” ;
	float NEST_POS(Time, south_north, west_east) ;
		NEST_POS:FieldType = 104 ;
		NEST_POS:MemoryOrder = “XY “ ;
		NEST_POS:description = “-” ;
		NEST_POS:units = “-” ;
		NEST_POS:stagger = “” ;
		NEST_POS:coordinates = “XLONG XLAT XTIME” ;
	float TKE(Time, bottom_top, south_north, west_east) ;
		TKE:FieldType = 104 ;
		TKE:MemoryOrder = “XYZ” ;
		TKE:description = “TURBULENCE KINETIC ENERGY” ;
		TKE:units = “m2 s-2” ;
		TKE:stagger = “” ;
		TKE:coordinates = “XLONG XLAT XTIME” ;
	float P(Time, bottom_top, south_north, west_east) ;
		P:FieldType = 104 ;
		P:MemoryOrder = “XYZ” ;
		P:description = “perturbation pressure” ;
		P:units = “Pa” ;
		P:stagger = “” ;
		P:coordinates = “XLONG XLAT XTIME” ;
	float ALT(Time, bottom_top, south_north, west_east) ;
		ALT:FieldType = 104 ;
		ALT:MemoryOrder = “XYZ” ;
		ALT:description = “inverse density” ;
		ALT:units = “m3 kg-1” ;
		ALT:stagger = “” ;
		ALT:coordinates = “XLONG XLAT XTIME” ;
	float PB(Time, bottom_top, south_north, west_east) ;
		PB:FieldType = 104 ;
		PB:MemoryOrder = “XYZ” ;
		PB:description = “BASE STATE PRESSURE” ;
```

```
		PB:units = "Pa" ;
		PB:stagger = "" ;
		PB:coordinates = "XLONG XLAT XTIME" ;
	float FNM(Time, bottom_top) ;
		FNM:FieldType = 104 ;
		FNM:MemoryOrder = "Z  " ;
		FNM:description = "upper weight for vertical stretching" ;
		FNM:units = "" ;
		FNM:stagger = "" ;
	float FNP(Time, bottom_top) ;
		FNP:FieldType = 104 ;
		FNP:MemoryOrder = "Z  " ;
		FNP:description = "lower weight for vertical stretching" ;
		FNP:units = "" ;
		FNP:stagger = "" ;
	float RDNW(Time, bottom_top) ;
		RDNW:FieldType = 104 ;
		RDNW:MemoryOrder = "Z  " ;
		RDNW:description = "inverse d(eta) values between full (w) levels" ;
		RDNW:units = "" ;
		RDNW:stagger = "" ;
	float RDN(Time, bottom_top) ;
		RDN:FieldType = 104 ;
		RDN:MemoryOrder = "Z  " ;
		RDN:description = "inverse d(eta) values between half (mass) levels" ;
		RDN:units = "" ;
		RDN:stagger = "" ;
	float DNW(Time, bottom_top) ;
		DNW:FieldType = 104 ;
		DNW:MemoryOrder = "Z  " ;
		DNW:description = "d(eta) values between full (w) levels" ;
		DNW:units = "" ;
		DNW:stagger = "" ;
	float DN(Time, bottom_top) ;
		DN:FieldType = 104 ;
		DN:MemoryOrder = "Z  " ;
		DN:description = "d(eta) values between half (mass) levels" ;
		DN:units = "" ;
		DN:stagger = "" ;
	float CFN(Time) ;
		CFN:FieldType = 104 ;
		CFN:MemoryOrder = "0  " ;
		CFN:description = "extrapolation constant" ;
		CFN:units = "" ;
		CFN:stagger = "" ;
	float CFN1(Time) ;
		CFN1:FieldType = 104 ;
		CFN1:MemoryOrder = "0  " ;
		CFN1:description = "extrapolation constant" ;
		CFN1:units = "" ;
		CFN1:stagger = "" ;
	int THIS_IS_AN_IDEAL_RUN(Time) ;
		THIS_IS_AN_IDEAL_RUN:FieldType = 106 ;
		THIS_IS_AN_IDEAL_RUN:MemoryOrder = "0  " ;
		THIS_IS_AN_IDEAL_RUN:description = "T/F flag: this is an ARW ideal simula-
tion" ;
		THIS_IS_AN_IDEAL_RUN:units = "-" ;
		THIS_IS_AN_IDEAL_RUN:stagger = "" ;
	float P_HYD(Time, bottom_top, south_north, west_east) ;
		P_HYD:FieldType = 104 ;
```

```
        P_HYD:MemoryOrder = “XYZ” ;
        P_HYD:description = “hydrostatic pressure” ;
        P_HYD:units = “Pa” ;
        P_HYD:stagger = “” ;
        P_HYD:coordinates = “XLONG XLAT XTIME” ;
float Q2(Time, south_north, west_east) ;
        Q2:FieldType = 104 ;
        Q2:MemoryOrder = “XY “ ;
        Q2:description = “QV at 2 M” ;
        Q2:units = “kg kg-1” ;
        Q2:stagger = “” ;
        Q2:coordinates = “XLONG XLAT XTIME” ;
float T2(Time, south_north, west_east) ;
        T2:FieldType = 104 ;
        T2:MemoryOrder = “XY “ ;
        T2:description = “TEMP at 2 M” ;
        T2:units = “K” ;
        T2:stagger = “” ;
        T2:coordinates = “XLONG XLAT XTIME” ;
float TH2(Time, south_north, west_east) ;
        TH2:FieldType = 104 ;
        TH2:MemoryOrder = “XY “ ;
        TH2:description = “POT TEMP at 2 M” ;
        TH2:units = “K” ;
        TH2:stagger = “” ;
        TH2:coordinates = “XLONG XLAT XTIME” ;
float PSFC(Time, south_north, west_east) ;
        PSFC:FieldType = 104 ;
        PSFC:MemoryOrder = “XY “ ;
        PSFC:description = “SFC PRESSURE” ;
        PSFC:units = “Pa” ;
        PSFC:stagger = “” ;
        PSFC:coordinates = “XLONG XLAT XTIME” ;
float U10(Time, south_north, west_east) ;
        U10:FieldType = 104 ;
        U10:MemoryOrder = “XY “ ;
        U10:description = “U at 10 M” ;
        U10:units = “m s-1” ;
        U10:stagger = “” ;
        U10:coordinates = “XLONG XLAT XTIME” ;
float V10(Time, south_north, west_east) ;
        V10:FieldType = 104 ;
        V10:MemoryOrder = “XY “ ;
        V10:description = “V at 10 M” ;
        V10:units = “m s-1” ;
        V10:stagger = “” ;
        V10:coordinates = “XLONG XLAT XTIME” ;
float RDX(Time) ;
        RDX:FieldType = 104 ;
        RDX:MemoryOrder = “0  “ ;
        RDX:description = “INVERSE X GRID LENGTH” ;
        RDX:units = “” ;
        RDX:stagger = “” ;
float RDY(Time) ;
        RDY:FieldType = 104 ;
        RDY:MemoryOrder = “0  “ ;
        RDY:description = “INVERSE Y GRID LENGTH” ;
        RDY:units = “” ;
        RDY:stagger = “” ;
```

```
float RESM(Time) ;
       RESM:FieldType = 104 ;
       RESM:MemoryOrder = "0  " ;
       RESM:description = "TIME WEIGHT CONSTANT FOR SMALL STEPS" ;
       RESM:units = "" ;
       RESM:stagger = "" ;
float ZETATOP(Time) ;
       ZETATOP:FieldType = 104 ;
       ZETATOP:MemoryOrder = "0  " ;
       ZETATOP:description = "ZETA AT MODEL TOP" ;
       ZETATOP:units = "" ;
       ZETATOP:stagger = "" ;
float CF1(Time) ;
       CF1:FieldType = 104 ;
       CF1:MemoryOrder = "0  " ;
       CF1:description = "2nd order extrapolation constant" ;
       CF1:units = "" ;
       CF1:stagger = "" ;
float CF2(Time) ;
       CF2:FieldType = 104 ;
       CF2:MemoryOrder = "0  " ;
       CF2:description = "2nd order extrapolation constant" ;
       CF2:units = "" ;
       CF2:stagger = "" ;
float CF3(Time) ;
       CF3:FieldType = 104 ;
       CF3:MemoryOrder = "0  " ;
       CF3:description = "2nd order extrapolation constant" ;
       CF3:units = "" ;
       CF3:stagger = "" ;
int ITIMESTEP(Time) ;
       ITIMESTEP:FieldType = 106 ;
       ITIMESTEP:MemoryOrder = "0  " ;
       ITIMESTEP:description = "" ;
       ITIMESTEP:units = "" ;
       ITIMESTEP:stagger = "" ;
float XTIME(Time) ;
       XTIME:FieldType = 104 ;
       XTIME:MemoryOrder = "0  " ;
       XTIME:description = "minutes since 2016-06-10 12:00:00" ;
       XTIME:units = "minutes since 2016-06-10 12:00:00" ;
       XTIME:stagger = "" ;
float QVAPOR(Time, bottom_top, south_north, west_east) ;
       QVAPOR:FieldType = 104 ;
       QVAPOR:MemoryOrder = "XYZ" ;
       QVAPOR:description = "Water vapor mixing ratio" ;
       QVAPOR:units = "kg kg-1" ;
       QVAPOR:stagger = "" ;
       QVAPOR:coordinates = "XLONG XLAT XTIME" ;
float QCLOUD(Time, bottom_top, south_north, west_east) ;
       QCLOUD:FieldType = 104 ;
       QCLOUD:MemoryOrder = "XYZ" ;
       QCLOUD:description = "Cloud water mixing ratio" ;
       QCLOUD:units = "kg kg-1" ;
       QCLOUD:stagger = "" ;
       QCLOUD:coordinates = "XLONG XLAT XTIME" ;
float QRAIN(Time, bottom_top, south_north, west_east) ;
       QRAIN:FieldType = 104 ;
       QRAIN:MemoryOrder = "XYZ" ;
       QRAIN:description = "Rain water mixing ratio" ;
```

```
        QRAIN:units = "kg kg-1" ;
        QRAIN:stagger = "" ;
        QRAIN:coordinates = "XLONG XLAT XTIME" ;
float QICE(Time, bottom_top, south_north, west_east) ;
        QICE:FieldType = 104 ;
        QICE:MemoryOrder = "XYZ" ;
        QICE:description = "Ice mixing ratio" ;
        QICE:units = "kg kg-1" ;
        QICE:stagger = "" ;
        QICE:coordinates = "XLONG XLAT XTIME" ;
float QSNOW(Time, bottom_top, south_north, west_east) ;
        QSNOW:FieldType = 104 ;
        QSNOW:MemoryOrder = "XYZ" ;
        QSNOW:description = "Snow mixing ratio" ;
        QSNOW:units = "kg kg-1" ;
        QSNOW:stagger = "" ;
        QSNOW:coordinates = "XLONG XLAT XTIME" ;
float QGRAUP(Time, bottom_top, south_north, west_east) ;
        QGRAUP:FieldType = 104 ;
        QGRAUP:MemoryOrder = "XYZ" ;
        QGRAUP:description = "Graupel mixing ratio" ;
        QGRAUP:units = "kg kg-1" ;
        QGRAUP:stagger = "" ;
        QGRAUP:coordinates = "XLONG XLAT XTIME" ;
float QNCLOUD(Time, bottom_top, south_north, west_east) ;
        QNCLOUD:FieldType = 104 ;
        QNCLOUD:MemoryOrder = "XYZ" ;
        QNCLOUD:description = "cloud water Number concentration" ;
        QNCLOUD:units = "  kg(-1)" ;
        QNCLOUD:stagger = "" ;
        QNCLOUD:coordinates = "XLONG XLAT XTIME" ;
float QNICE(Time, bottom_top, south_north, west_east) ;
        QNICE:FieldType = 104 ;
        QNICE:MemoryOrder = "XYZ" ;
        QNICE:description = "Ice Number concentration" ;
        QNICE:units = "  kg-1" ;
        QNICE:stagger = "" ;
        QNICE:coordinates = "XLONG XLAT XTIME" ;
float QNSNOW(Time, bottom_top, south_north, west_east) ;
        QNSNOW:FieldType = 104 ;
        QNSNOW:MemoryOrder = "XYZ" ;
        QNSNOW:description = "Snow Number concentration" ;
        QNSNOW:units = "  kg(-1)" ;
        QNSNOW:stagger = "" ;
        QNSNOW:coordinates = "XLONG XLAT XTIME" ;
float QNRAIN(Time, bottom_top, south_north, west_east) ;
        QNRAIN:FieldType = 104 ;
        QNRAIN:MemoryOrder = "XYZ" ;
        QNRAIN:description = "Rain Number concentration" ;
        QNRAIN:units = "  kg(-1)" ;
        QNRAIN:stagger = "" ;
        QNRAIN:coordinates = "XLONG XLAT XTIME" ;
float QNGRAUPEL(Time, bottom_top, south_north, west_east) ;
        QNGRAUPEL:FieldType = 104 ;
        QNGRAUPEL:MemoryOrder = "XYZ" ;
        QNGRAUPEL:description = "Graupel Number concentration" ;
        QNGRAUPEL:units = "  kg(-1)" ;
        QNGRAUPEL:stagger = "" ;
        QNGRAUPEL:coordinates = "XLONG XLAT XTIME" ;
```

```
float SHDMAX(Time, south_north, west_east) ;
       SHDMAX:FieldType = 104 ;
       SHDMAX:MemoryOrder = "XY " ;
       SHDMAX:description = "ANNUAL MAX VEG FRACTION" ;
       SHDMAX:units = "" ;
       SHDMAX:stagger = "" ;
       SHDMAX:coordinates = "XLONG XLAT XTIME" ;
float SHDMIN(Time, south_north, west_east) ;
       SHDMIN:FieldType = 104 ;
       SHDMIN:MemoryOrder = "XY " ;
       SHDMIN:description = "ANNUAL MIN VEG FRACTION" ;
       SHDMIN:units = "" ;
       SHDMIN:stagger = "" ;
       SHDMIN:coordinates = "XLONG XLAT XTIME" ;
float SNOALB(Time, south_north, west_east) ;
       SNOALB:FieldType = 104 ;
       SNOALB:MemoryOrder = "XY " ;
       SNOALB:description = "ANNUAL MAX SNOW ALBEDO IN FRACTION" ;
       SNOALB:units = "" ;
       SNOALB:stagger = "" ;
       SNOALB:coordinates = "XLONG XLAT XTIME" ;
float TSLB(Time, soil_layers_stag, south_north, west_east) ;
       TSLB:FieldType = 104 ;
       TSLB:MemoryOrder = "XYZ" ;
       TSLB:description = "SOIL TEMPERATURE" ;
       TSLB:units = "K" ;
       TSLB:stagger = "Z" ;
       TSLB:coordinates = "XLONG XLAT XTIME" ;
float SMOIS(Time, soil_layers_stag, south_north, west_east) ;
       SMOIS:FieldType = 104 ;
       SMOIS:MemoryOrder = "XYZ" ;
       SMOIS:description = "SOIL MOISTURE" ;
       SMOIS:units = "m3 m-3" ;
       SMOIS:stagger = "Z" ;
       SMOIS:coordinates = "XLONG XLAT XTIME" ;
float SH2O(Time, soil_layers_stag, south_north, west_east) ;
       SH2O:FieldType = 104 ;
       SH2O:MemoryOrder = "XYZ" ;
       SH2O:description = "SOIL LIQUID WATER" ;
       SH2O:units = "m3 m-3" ;
       SH2O:stagger = "Z" ;
       SH2O:coordinates = "XLONG XLAT XTIME" ;
float SEAICE(Time, south_north, west_east) ;
       SEAICE:FieldType = 104 ;
       SEAICE:MemoryOrder = "XY " ;
       SEAICE:description = "SEA ICE FLAG" ;
       SEAICE:units = "" ;
       SEAICE:stagger = "" ;
       SEAICE:coordinates = "XLONG XLAT XTIME" ;
float XICEM(Time, south_north, west_east) ;
       XICEM:FieldType = 104 ;
       XICEM:MemoryOrder = "XY " ;
       XICEM:description = "SEA ICE FLAG (PREVIOUS STEP)" ;
       XICEM:units = "" ;
       XICEM:stagger = "" ;
       XICEM:coordinates = "XLONG XLAT XTIME" ;
float SFROFF(Time, south_north, west_east) ;
       SFROFF:FieldType = 104 ;
       SFROFF:MemoryOrder = "XY " ;
       SFROFF:description = "SURFACE RUNOFF" ;
```

```
        SFROFF:units = “mm” ;
        SFROFF:stagger = “” ;
        SFROFF:coordinates = “XLONG XLAT XTIME” ;
float UDROFF(Time, south_north, west_east) ;
        UDROFF:FieldType = 104 ;
        UDROFF:MemoryOrder = “XY “ ;
        UDROFF:description = “UNDERGROUND RUNOFF” ;
        UDROFF:units = “mm” ;
        UDROFF:stagger = “” ;
        UDROFF:coordinates = “XLONG XLAT XTIME” ;
int IVGTYP(Time, south_north, west_east) ;
        IVGTYP:FieldType = 106 ;
        IVGTYP:MemoryOrder = “XY “ ;
        IVGTYP:description = “DOMINANT VEGETATION CATEGORY” ;
        IVGTYP:units = “” ;
        IVGTYP:stagger = “” ;
        IVGTYP:coordinates = “XLONG XLAT XTIME” ;
int ISLTYP(Time, south_north, west_east) ;
        ISLTYP:FieldType = 106 ;
        ISLTYP:MemoryOrder = “XY “ ;
        ISLTYP:description = “DOMINANT SOIL CATEGORY” ;
        ISLTYP:units = “” ;
        ISLTYP:stagger = “” ;
        ISLTYP:coordinates = “XLONG XLAT XTIME” ;
float VEGFRA(Time, south_north, west_east) ;
        VEGFRA:FieldType = 104 ;
        VEGFRA:MemoryOrder = “XY “ ;
        VEGFRA:description = “VEGETATION FRACTION” ;
        VEGFRA:units = “” ;
        VEGFRA:stagger = “” ;
        VEGFRA:coordinates = “XLONG XLAT XTIME” ;
float GRDFLX(Time, south_north, west_east) ;
        GRDFLX:FieldType = 104 ;
        GRDFLX:MemoryOrder = “XY “ ;
        GRDFLX:description = “GROUND HEAT FLUX” ;
        GRDFLX:units = “W m-2” ;
        GRDFLX:stagger = “” ;
        GRDFLX:coordinates = “XLONG XLAT XTIME” ;
float ACGRDFLX(Time, south_north, west_east) ;
        ACGRDFLX:FieldType = 104 ;
        ACGRDFLX:MemoryOrder = “XY “ ;
        ACGRDFLX:description = “ACCUMULATED GROUND HEAT FLUX” ;
        ACGRDFLX:units = “J m-2” ;
        ACGRDFLX:stagger = “” ;
        ACGRDFLX:coordinates = “XLONG XLAT XTIME” ;
float ACSNOM(Time, south_north, west_east) ;
        ACSNOM:FieldType = 104 ;
        ACSNOM:MemoryOrder = “XY “ ;
        ACSNOM:description = “ACCUMULATED MELTED SNOW” ;
        ACSNOM:units = “kg m-2” ;
        ACSNOM:stagger = “” ;
        ACSNOM:coordinates = “XLONG XLAT XTIME” ;
float SNOW(Time, south_north, west_east) ;
        SNOW:FieldType = 104 ;
        SNOW:MemoryOrder = “XY “ ;
        SNOW:description = “SNOW WATER EQUIVALENT” ;
        SNOW:units = “kg m-2” ;
        SNOW:stagger = “” ;
        SNOW:coordinates = “XLONG XLAT XTIME” ;
```

```
float SNOWH(Time, south_north, west_east) ;
       SNOWH:FieldType = 104 ;
       SNOWH:MemoryOrder = "XY " ;
       SNOWH:description = "PHYSICAL SNOW DEPTH" ;
       SNOWH:units = "m" ;
       SNOWH:stagger = "" ;
       SNOWH:coordinates = "XLONG XLAT XTIME" ;
float CANWAT(Time, south_north, west_east) ;
       CANWAT:FieldType = 104 ;
       CANWAT:MemoryOrder = "XY " ;
       CANWAT:description = "CANOPY WATER" ;
       CANWAT:units = "kg m-2" ;
       CANWAT:stagger = "" ;
       CANWAT:coordinates = "XLONG XLAT XTIME" ;
float SSTSK(Time, south_north, west_east) ;
       SSTSK:FieldType = 104 ;
       SSTSK:MemoryOrder = "XY " ;
       SSTSK:description = "SKIN SEA SURFACE TEMPERATURE" ;
       SSTSK:units = "K" ;
       SSTSK:stagger = "" ;
       SSTSK:coordinates = "XLONG XLAT XTIME" ;
float COSZEN(Time, south_north, west_east) ;
       COSZEN:FieldType = 104 ;
       COSZEN:MemoryOrder = "XY " ;
       COSZEN:description = "COS of SOLAR ZENITH ANGLE" ;
       COSZEN:units = "dimensionless" ;
       COSZEN:stagger = "" ;
       COSZEN:coordinates = "XLONG XLAT XTIME" ;
float LAI(Time, south_north, west_east) ;
       LAI:FieldType = 104 ;
       LAI:MemoryOrder = "XY " ;
       LAI:description = "LEAF AREA INDEX" ;
       LAI:units = "m-2/m-2" ;
       LAI:stagger = "" ;
       LAI:coordinates = "XLONG XLAT XTIME" ;
float VAR(Time, south_north, west_east) ;
       VAR:FieldType = 104 ;
       VAR:MemoryOrder = "XY " ;
       VAR:description = "OROGRAPHIC VARIANCE" ;
       VAR:units = "" ;
       VAR:stagger = "" ;
       VAR:coordinates = "XLONG XLAT XTIME" ;
float MAPFAC_M(Time, south_north, west_east) ;
       MAPFAC_M:FieldType = 104 ;
       MAPFAC_M:MemoryOrder = "XY " ;
       MAPFAC_M:description = "Map scale factor on mass grid" ;
       MAPFAC_M:units = "" ;
       MAPFAC_M:stagger = "" ;
       MAPFAC_M:coordinates = "XLONG XLAT XTIME" ;
float MAPFAC_U(Time, south_north, west_east_stag) ;
       MAPFAC_U:FieldType = 104 ;
       MAPFAC_U:MemoryOrder = "XY " ;
       MAPFAC_U:description = "Map scale factor on u-grid" ;
       MAPFAC_U:units = "" ;
       MAPFAC_U:stagger = "X" ;
       MAPFAC_U:coordinates = "XLONG_U XLAT_U XTIME" ;
float MAPFAC_V(Time, south_north_stag, west_east) ;
       MAPFAC_V:FieldType = 104 ;
       MAPFAC_V:MemoryOrder = "XY " ;
       MAPFAC_V:description = "Map scale factor on v-grid" ;
```

```
        MAPFAC_V:units = "" ;
        MAPFAC_V:stagger = "Y" ;
        MAPFAC_V:coordinates = "XLONG_V XLAT_V XTIME" ;
float MAPFAC_MX(Time, south_north, west_east) ;
        MAPFAC_MX:FieldType = 104 ;
        MAPFAC_MX:MemoryOrder = "XY " ;
        MAPFAC_MX:description = "Map scale factor on mass grid, x direction" ;
        MAPFAC_MX:units = "" ;
        MAPFAC_MX:stagger = "" ;
        MAPFAC_MX:coordinates = "XLONG XLAT XTIME" ;
float MAPFAC_MY(Time, south_north, west_east) ;
        MAPFAC_MY:FieldType = 104 ;
        MAPFAC_MY:MemoryOrder = "XY " ;
        MAPFAC_MY:description = "Map scale factor on mass grid, y direction" ;
        MAPFAC_MY:units = "" ;
        MAPFAC_MY:stagger = "" ;
        MAPFAC_MY:coordinates = "XLONG XLAT XTIME" ;
float MAPFAC_UX(Time, south_north, west_east_stag) ;
        MAPFAC_UX:FieldType = 104 ;
        MAPFAC_UX:MemoryOrder = "XY " ;
        MAPFAC_UX:description = "Map scale factor on u-grid, x direction" ;
        MAPFAC_UX:units = "" ;
        MAPFAC_UX:stagger = "X" ;
        MAPFAC_UX:coordinates = "XLONG_U XLAT_U XTIME" ;
float MAPFAC_UY(Time, south_north, west_east_stag) ;
        MAPFAC_UY:FieldType = 104 ;
        MAPFAC_UY:MemoryOrder = "XY " ;
        MAPFAC_UY:description = "Map scale factor on u-grid, y direction" ;
        MAPFAC_UY:units = "" ;
        MAPFAC_UY:stagger = "X" ;
        MAPFAC_UY:coordinates = "XLONG_U XLAT_U XTIME" ;
float MAPFAC_VX(Time, south_north_stag, west_east) ;
        MAPFAC_VX:FieldType = 104 ;
        MAPFAC_VX:MemoryOrder = "XY " ;
        MAPFAC_VX:description = "Map scale factor on v-grid, x direction" ;
        MAPFAC_VX:units = "" ;
        MAPFAC_VX:stagger = "Y" ;
        MAPFAC_VX:coordinates = "XLONG_V XLAT_V XTIME" ;
float MF_VX_INV(Time, south_north_stag, west_east) ;
        MF_VX_INV:FieldType = 104 ;
        MF_VX_INV:MemoryOrder = "XY " ;
        MF_VX_INV:description = "Inverse map scale factor on v-grid, x direction" ;
        MF_VX_INV:units = "" ;
        MF_VX_INV:stagger = "Y" ;
        MF_VX_INV:coordinates = "XLONG_V XLAT_V XTIME" ;
float MAPFAC_VY(Time, south_north_stag, west_east) ;
        MAPFAC_VY:FieldType = 104 ;
        MAPFAC_VY:MemoryOrder = "XY " ;
        MAPFAC_VY:description = "Map scale factor on v-grid, y direction" ;
        MAPFAC_VY:units = "" ;
        MAPFAC_VY:stagger = "Y" ;
        MAPFAC_VY:coordinates = "XLONG_V XLAT_V XTIME" ;
float F(Time, south_north, west_east) ;
        F:FieldType = 104 ;
        F:MemoryOrder = "XY " ;
        F:description = "Coriolis sine latitude term" ;
        F:units = "s-1" ;
        F:stagger = "" ;
        F:coordinates = "XLONG XLAT XTIME" ;
```

```
float E(Time, south_north, west_east) ;
       E:FieldType = 104 ;
       E:MemoryOrder = "XY " ;
       E:description = "Coriolis cosine latitude term" ;
       E:units = "s-1" ;
       E:stagger = "" ;
       E:coordinates = "XLONG XLAT XTIME" ;
float SINALPHA(Time, south_north, west_east) ;
       SINALPHA:FieldType = 104 ;
       SINALPHA:MemoryOrder = "XY " ;
       SINALPHA:description = "Local sine of map rotation" ;
       SINALPHA:units = "" ;
       SINALPHA:stagger = "" ;
       SINALPHA:coordinates = "XLONG XLAT XTIME" ;
float COSALPHA(Time, south_north, west_east) ;
       COSALPHA:FieldType = 104 ;
       COSALPHA:MemoryOrder = "XY " ;
       COSALPHA:description = "Local cosine of map rotation" ;
       COSALPHA:units = "" ;
       COSALPHA:stagger = "" ;
       COSALPHA:coordinates = "XLONG XLAT XTIME" ;
float HGT(Time, south_north, west_east) ;
       HGT:FieldType = 104 ;
       HGT:MemoryOrder = "XY " ;
       HGT:description = "Terrain Height" ;
       HGT:units = "m" ;
       HGT:stagger = "" ;
       HGT:coordinates = "XLONG XLAT XTIME" ;
float TSK(Time, south_north, west_east) ;
       TSK:FieldType = 104 ;
       TSK:MemoryOrder = "XY " ;
       TSK:description = "SURFACE SKIN TEMPERATURE" ;
       TSK:units = "K" ;
       TSK:stagger = "" ;
       TSK:coordinates = "XLONG XLAT XTIME" ;
float P_TOP(Time) ;
       P_TOP:FieldType = 104 ;
       P_TOP:MemoryOrder = "0  " ;
       P_TOP:description = "PRESSURE TOP OF THE MODEL" ;
       P_TOP:units = "Pa" ;
       P_TOP:stagger = "" ;
float T00(Time) ;
       T00:FieldType = 104 ;
       T00:MemoryOrder = "0  " ;
       T00:description = "BASE STATE TEMPERATURE" ;
       T00:units = "K" ;
       T00:stagger = "" ;
float P00(Time) ;
       P00:FieldType = 104 ;
       P00:MemoryOrder = "0  " ;
       P00:description = "BASE STATE PRESURE" ;
       P00:units = "Pa" ;
       P00:stagger = "" ;
float TLP(Time) ;
       TLP:FieldType = 104 ;
       TLP:MemoryOrder = "0  " ;
       TLP:description = "BASE STATE LAPSE RATE" ;
       TLP:units = "" ;
       TLP:stagger = "" ;
```

```
        float TISO(Time) ;
                TISO:FieldType = 104 ;
                TISO:MemoryOrder = "0  " ;
                TISO:description = "TEMP AT WHICH THE BASE T TURNS CONST" ;
                TISO:units = "K" ;
                TISO:stagger = "" ;
        float TLP_STRAT(Time) ;
                TLP_STRAT:FieldType = 104 ;
                TLP_STRAT:MemoryOrder = "0  " ;
                TLP_STRAT:description = "BASE STATE LAPSE RATE (DT/D(LN(P)) IN STRATO-
SPHERE" ;
                TLP_STRAT:units = "K" ;
                TLP_STRAT:stagger = "" ;
        float P_STRAT(Time) ;
                P_STRAT:FieldType = 104 ;
                P_STRAT:MemoryOrder = "0  " ;
                P_STRAT:description = "BASE STATE PRESSURE AT BOTTOM OF STRATOSPHERE" ;
                P_STRAT:units = "Pa" ;
                P_STRAT:stagger = "" ;
        float MAX_MSTFX(Time) ;
                MAX_MSTFX:FieldType = 104 ;
                MAX_MSTFX:MemoryOrder = "0  " ;
                MAX_MSTFX:description = "Max map factor in domain" ;
                MAX_MSTFX:units = "" ;
                MAX_MSTFX:stagger = "" ;
        float MAX_MSTFY(Time) ;
                MAX_MSTFY:FieldType = 104 ;
                MAX_MSTFY:MemoryOrder = "0  " ;
                MAX_MSTFY:description = "Max map factor in domain" ;
                MAX_MSTFY:units = "" ;
                MAX_MSTFY:stagger = "" ;
        float RAINC(Time, south_north, west_east) ;
                RAINC:FieldType = 104 ;
                RAINC:MemoryOrder = "XY " ;
                RAINC:description = "ACCUMULATED TOTAL CUMULUS PRECIPITATION" ;
                RAINC:units = "mm" ;
                RAINC:stagger = "" ;
                RAINC:coordinates = "XLONG XLAT XTIME" ;
        float RAINSH(Time, south_north, west_east) ;
                RAINSH:FieldType = 104 ;
                RAINSH:MemoryOrder = "XY " ;
                RAINSH:description = "ACCUMULATED SHALLOW CUMULUS PRECIPITATION" ;
                RAINSH:units = "mm" ;
                RAINSH:stagger = "" ;
                RAINSH:coordinates = "XLONG XLAT XTIME" ;
        float RAINNC(Time, south_north, west_east) ;
                RAINNC:FieldType = 104 ;
                RAINNC:MemoryOrder = "XY " ;
                RAINNC:description = "ACCUMULATED TOTAL GRID SCALE PRECIPITATION" ;
                RAINNC:units = "mm" ;
                RAINNC:stagger = "" ;
                RAINNC:coordinates = "XLONG XLAT XTIME" ;
        float SNOWNC(Time, south_north, west_east) ;
                SNOWNC:FieldType = 104 ;
                SNOWNC:MemoryOrder = "XY " ;
                SNOWNC:description = "ACCUMULATED TOTAL GRID SCALE SNOW AND ICE" ;
                SNOWNC:units = "mm" ;
                SNOWNC:stagger = "" ;
                SNOWNC:coordinates = "XLONG XLAT XTIME" ;
```

```
	float GRAUPELNC(Time, south_north, west_east) ;
		GRAUPELNC:FieldType = 104 ;
		GRAUPELNC:MemoryOrder = "XY " ;
		GRAUPELNC:description = "ACCUMULATED TOTAL GRID SCALE GRAUPEL" ;
		GRAUPELNC:units = "mm" ;
		GRAUPELNC:stagger = "" ;
		GRAUPELNC:coordinates = "XLONG XLAT XTIME" ;
	float HAILNC(Time, south_north, west_east) ;
		HAILNC:FieldType = 104 ;
		HAILNC:MemoryOrder = "XY " ;
		HAILNC:description = "ACCUMULATED TOTAL GRID SCALE HAIL" ;
		HAILNC:units = "mm" ;
		HAILNC:stagger = "" ;
		HAILNC:coordinates = "XLONG XLAT XTIME" ;
	float REFL_10CM(Time, bottom_top, south_north, west_east) ;
		REFL_10CM:FieldType = 104 ;
		REFL_10CM:MemoryOrder = "XYZ" ;
		REFL_10CM:description = "Radar reflectivity (lamda = 10 cm)" ;
		REFL_10CM:units = "dBZ" ;
		REFL_10CM:stagger = "" ;
		REFL_10CM:coordinates = "XLONG XLAT XTIME" ;
	float CLDFRA(Time, bottom_top, south_north, west_east) ;
		CLDFRA:FieldType = 104 ;
		CLDFRA:MemoryOrder = "XYZ" ;
		CLDFRA:description = "CLOUD FRACTION" ;
		CLDFRA:units = "" ;
		CLDFRA:stagger = "" ;
		CLDFRA:coordinates = "XLONG XLAT XTIME" ;
	float SWDOWN(Time, south_north, west_east) ;
		SWDOWN:FieldType = 104 ;
		SWDOWN:MemoryOrder = "XY " ;
		SWDOWN:description = "DOWNWARD SHORT WAVE FLUX AT GROUND SURFACE" ;
		SWDOWN:units = "W m-2" ;
		SWDOWN:stagger = "" ;
		SWDOWN:coordinates = "XLONG XLAT XTIME" ;
	float GLW(Time, south_north, west_east) ;
		GLW:FieldType = 104 ;
		GLW:MemoryOrder = "XY " ;
		GLW:description = "DOWNWARD LONG WAVE FLUX AT GROUND SURFACE" ;
		GLW:units = "W m-2" ;
		GLW:stagger = "" ;
		GLW:coordinates = "XLONG XLAT XTIME" ;
	float SWNORM(Time, south_north, west_east) ;
		SWNORM:FieldType = 104 ;
		SWNORM:MemoryOrder = "XY " ;
		SWNORM:description = "NORMAL SHORT WAVE FLUX AT GROUND SURFACE (SLOPE-DE-
PENDENT)" ;
		SWNORM:units = "W m-2" ;
		SWNORM:stagger = "" ;
		SWNORM:coordinates = "XLONG XLAT XTIME" ;
	float ACSWUPT(Time, south_north, west_east) ;
		ACSWUPT:FieldType = 104 ;
		ACSWUPT:MemoryOrder = "XY " ;
		ACSWUPT:description = "ACCUMULATED UPWELLING SHORTWAVE FLUX AT TOP" ;
		ACSWUPT:units = "J m-2" ;
		ACSWUPT:stagger = "" ;
		ACSWUPT:coordinates = "XLONG XLAT XTIME" ;
	float ACSWUPTC(Time, south_north, west_east) ;
		ACSWUPTC:FieldType = 104 ;
		ACSWUPTC:MemoryOrder = "XY " ;
```

```
             ACSWUPTC:description = "ACCUMULATED UPWELLING CLEAR SKY SHORTWAVE FLUX AT
TOP" ;
             ACSWUPTC:units = "J m-2" ;
             ACSWUPTC:stagger = "" ;
             ACSWUPTC:coordinates = "XLONG XLAT XTIME" ;
      float ACSWDNT(Time, south_north, west_east) ;
             ACSWDNT:FieldType = 104 ;
             ACSWDNT:MemoryOrder = "XY " ;
             ACSWDNT:description = "ACCUMULATED DOWNWELLING SHORTWAVE FLUX AT TOP" ;
             ACSWDNT:units = "J m-2" ;
             ACSWDNT:stagger = "" ;
             ACSWDNT:coordinates = "XLONG XLAT XTIME" ;
      float ACSWDNTC(Time, south_north, west_east) ;
             ACSWDNTC:FieldType = 104 ;
             ACSWDNTC:MemoryOrder = "XY " ;
             ACSWDNTC:description = "ACCUMULATED DOWNWELLING CLEAR SKY SHORTWAVE FLUX
AT TOP" ;
             ACSWDNTC:units = "J m-2" ;
             ACSWDNTC:stagger = "" ;
             ACSWDNTC:coordinates = "XLONG XLAT XTIME" ;
      float ACSWUPB(Time, south_north, west_east) ;
             ACSWUPB:FieldType = 104 ;
             ACSWUPB:MemoryOrder = "XY " ;
             ACSWUPB:description = "ACCUMULATED UPWELLING SHORTWAVE FLUX AT BOTTOM" ;
             ACSWUPB:units = "J m-2" ;
             ACSWUPB:stagger = "" ;
             ACSWUPB:coordinates = "XLONG XLAT XTIME" ;
      float ACSWUPBC(Time, south_north, west_east) ;
             ACSWUPBC:FieldType = 104 ;
             ACSWUPBC:MemoryOrder = "XY " ;
             ACSWUPBC:description = "ACCUMULATED UPWELLING CLEAR SKY SHORTWAVE FLUX AT
BOTTOM" ;
             ACSWUPBC:units = "J m-2" ;
             ACSWUPBC:stagger = "" ;
             ACSWUPBC:coordinates = "XLONG XLAT XTIME" ;
      float ACSWDNB(Time, south_north, west_east) ;
             ACSWDNB:FieldType = 104 ;
             ACSWDNB:MemoryOrder = "XY " ;
             ACSWDNB:description = "ACCUMULATED DOWNWELLING SHORTWAVE FLUX AT BOTTOM" ;
             ACSWDNB:units = "J m-2" ;
             ACSWDNB:stagger = "" ;
             ACSWDNB:coordinates = "XLONG XLAT XTIME" ;
      float ACSWDNBC(Time, south_north, west_east) ;
             ACSWDNBC:FieldType = 104 ;
             ACSWDNBC:MemoryOrder = "XY " ;
             ACSWDNBC:description = "ACCUMULATED DOWNWELLING CLEAR SKY SHORTWAVE FLUX
AT BOTTOM" ;
             ACSWDNBC:units = "J m-2" ;
             ACSWDNBC:stagger = "" ;
             ACSWDNBC:coordinates = "XLONG XLAT XTIME" ;
      float ACLWUPT(Time, south_north, west_east) ;
             ACLWUPT:FieldType = 104 ;
             ACLWUPT:MemoryOrder = "XY " ;
             ACLWUPT:description = "ACCUMULATED UPWELLING LONGWAVE FLUX AT TOP" ;
             ACLWUPT:units = "J m-2" ;
             ACLWUPT:stagger = "" ;
             ACLWUPT:coordinates = "XLONG XLAT XTIME" ;
      float ACLWUPTC(Time, south_north, west_east) ;
             ACLWUPTC:FieldType = 104 ;
             ACLWUPTC:MemoryOrder = "XY " ;
```

```
            ACLWUPTC:description = "ACCUMULATED UPWELLING CLEAR SKY LONGWAVE FLUX AT
TOP" ;
            ACLWUPTC:units = "J m-2" ;
            ACLWUPTC:stagger = "" ;
            ACLWUPTC:coordinates = "XLONG XLAT XTIME" ;    float ACLWDNT(Time, south_
north, west_east) ;
            ACLWDNT:FieldType = 104 ;
            ACLWDNT:MemoryOrder = "XY " ;
            ACLWDNT:description = "ACCUMULATED DOWNWELLING LONGWAVE FLUX AT TOP" ;
            ACLWDNT:units = "J m-2" ;
            ACLWDNT:stagger = "" ;
            ACLWDNT:coordinates = "XLONG XLAT XTIME" ;
      float ACLWDNTC(Time, south_north, west_east) ;
            ACLWDNTC:FieldType = 104 ;
            ACLWDNTC:MemoryOrder = "XY " ;
            ACLWDNTC:description = "ACCUMULATED DOWNWELLING CLEAR SKY LONGWAVE FLUX
AT TOP" ;
            ACLWDNTC:units = "J m-2" ;
            ACLWDNTC:stagger = "" ;
            ACLWDNTC:coordinates = "XLONG XLAT XTIME" ;
      float ACLWUPB(Time, south_north, west_east) ;
            ACLWUPB:FieldType = 104 ;
            ACLWUPB:MemoryOrder = "XY " ;
            ACLWUPB:description = "ACCUMULATED UPWELLING LONGWAVE FLUX AT BOTTOM" ;
            ACLWUPB:units = "J m-2" ;
            ACLWUPB:stagger = "" ;
            ACLWUPB:coordinates = "XLONG XLAT XTIME" ;
      float ACLWUPBC(Time, south_north, west_east) ;
            ACLWUPBC:FieldType = 104 ;
            ACLWUPBC:MemoryOrder = "XY " ;
            ACLWUPBC:description = "ACCUMULATED UPWELLING CLEAR SKY LONGWAVE FLUX AT
BOTTOM" ;
            ACLWUPBC:units = "J m-2" ;
            ACLWUPBC:stagger = "" ;
            ACLWUPBC:coordinates = "XLONG XLAT XTIME" ;
      float ACLWDNB(Time, south_north, west_east) ;
            ACLWDNB:FieldType = 104 ;
            ACLWDNB:MemoryOrder = "XY " ;
            ACLWDNB:description = "ACCUMULATED DOWNWELLING LONGWAVE FLUX AT BOTTOM" ;
            ACLWDNB:units = "J m-2" ;
            ACLWDNB:stagger = "" ;
            ACLWDNB:coordinates = "XLONG XLAT XTIME" ;
      float ACLWDNBC(Time, south_north, west_east) ;
            ACLWDNBC:FieldType = 104 ;
            ACLWDNBC:MemoryOrder = "XY " ;
            ACLWDNBC:description = "ACCUMULATED DOWNWELLING CLEAR SKY LONGWAVE FLUX
AT BOTTOM" ;
            ACLWDNBC:units = "J m-2" ;
            ACLWDNBC:stagger = "" ;
            ACLWDNBC:coordinates = "XLONG XLAT XTIME" ;
      int I_ACSWUPT(Time, south_north, west_east) ;
            I_ACSWUPT:FieldType = 106 ;
            I_ACSWUPT:MemoryOrder = "XY " ;
            I_ACSWUPT:description = "BUCKET FOR UPWELLING SHORTWAVE FLUX AT TOP" ;
            I_ACSWUPT:units = "J m-2" ;
            I_ACSWUPT:stagger = "" ;
            I_ACSWUPT:coordinates = "XLONG XLAT XTIME" ;
      int I_ACSWUPTC(Time, south_north, west_east) ;
            I_ACSWUPTC:FieldType = 106 ;
            I_ACSWUPTC:MemoryOrder = "XY " ;
```

```
              I_ACSWUPTC:description = "BUCKET FOR UPWELLING CLEAR SKY SHORTWAVE FLUX
AT TOP" ;
              I_ACSWUPTC:units = "J m-2" ;
              I_ACSWUPTC:stagger = "" ;
              I_ACSWUPTC:coordinates = "XLONG XLAT XTIME" ;
       int I_ACSWDNT(Time, south_north, west_east) ;
              I_ACSWDNT:FieldType = 106 ;
              I_ACSWDNT:MemoryOrder = "XY " ;
              I_ACSWDNT:description = "BUCKET FOR DOWNWELLING SHORTWAVE FLUX AT TOP" ;
              I_ACSWDNT:units = "J m-2" ;
              I_ACSWDNT:stagger = "" ;
              I_ACSWDNT:coordinates = "XLONG XLAT XTIME" ;
       int I_ACSWDNTC(Time, south_north, west_east) ;
              I_ACSWDNTC:FieldType = 106 ;
              I_ACSWDNTC:MemoryOrder = "XY " ;
              I_ACSWDNTC:description = "BUCKET FOR DOWNWELLING CLEAR SKY SHORTWAVE FLUX
AT TOP" ;
              I_ACSWDNTC:units = "J m-2" ;
              I_ACSWDNTC:stagger = "" ;
              I_ACSWDNTC:coordinates = "XLONG XLAT XTIME" ;
       int I_ACSWUPB(Time, south_north, west_east) ;
              I_ACSWUPB:FieldType = 106 ;
              I_ACSWUPB:MemoryOrder = "XY " ;
              I_ACSWUPB:description = "BUCKET FOR UPWELLING SHORTWAVE FLUX AT BOTTOM" ;
              I_ACSWUPB:units = "J m-2" ;
              I_ACSWUPB:stagger = "" ;
              I_ACSWUPB:coordinates = "XLONG XLAT XTIME" ;
       int I_ACSWUPBC(Time, south_north, west_east) ;
              I_ACSWUPBC:FieldType = 106 ;
              I_ACSWUPBC:MemoryOrder = "XY " ;
              I_ACSWUPBC:description = "BUCKET FOR UPWELLING CLEAR SKY SHORTWAVE FLUX
AT BOTTOM" ;
              I_ACSWUPBC:units = "J m-2" ;
              I_ACSWUPBC:stagger = "" ;
              I_ACSWUPBC:coordinates = "XLONG XLAT XTIME" ;
       int I_ACSWDNB(Time, south_north, west_east) ;
              I_ACSWDNB:FieldType = 106 ;
              I_ACSWDNB:MemoryOrder = "XY " ;
              I_ACSWDNB:description = "BUCKET FOR DOWNWELLING SHORTWAVE FLUX AT BOTTOM" ;
              I_ACSWDNB:units = "J m-2" ;
              I_ACSWDNB:stagger = "" ;
              I_ACSWDNB:coordinates = "XLONG XLAT XTIME" ;
       int I_ACSWDNBC(Time, south_north, west_east) ;
              I_ACSWDNBC:FieldType = 106 ;
              I_ACSWDNBC:MemoryOrder = "XY " ;
              I_ACSWDNBC:description = "BUCKET FOR DOWNWELLING CLEAR SKY SHORTWAVE FLUX
AT BOTTOM" ;
              I_ACSWDNBC:units = "J m-2" ;
              I_ACSWDNBC:stagger = "" ;
              I_ACSWDNBC:coordinates = "XLONG XLAT XTIME" ;
       int I_ACLWUPT(Time, south_north, west_east) ;
              I_ACLWUPT:FieldType = 106 ;
              I_ACLWUPT:MemoryOrder = "XY " ;
              I_ACLWUPT:description = "BUCKET FOR UPWELLING LONGWAVE FLUX AT TOP" ;
              I_ACLWUPT:units = "J m-2" ;
              I_ACLWUPT:stagger = "" ;
              I_ACLWUPT:coordinates = "XLONG XLAT XTIME" ;
       int I_ACLWUPTC(Time, south_north, west_east) ;
              I_ACLWUPTC:FieldType = 106 ;
              I_ACLWUPTC:MemoryOrder = "XY " ;
```

```
             I_ACLWUPTC:description = "BUCKET FOR UPWELLING CLEAR SKY LONGWAVE FLUX AT
TOP" ;
             I_ACLWUPTC:units = "J m-2" ;
             I_ACLWUPTC:stagger = "" ;
             I_ACLWUPTC:coordinates = "XLONG XLAT XTIME" ;
       int I_ACLWDNT(Time, south_north, west_east) ;
             I_ACLWDNT:FieldType = 106 ;
             I_ACLWDNT:MemoryOrder = "XY " ;
             I_ACLWDNT:description = "BUCKET FOR DOWNWELLING LONGWAVE FLUX AT TOP" ;
             I_ACLWDNT:units = "J m-2" ;
             I_ACLWDNT:stagger = "" ;
             I_ACLWDNT:coordinates = "XLONG XLAT XTIME" ;
       int I_ACLWDNTC(Time, south_north, west_east) ;
             I_ACLWDNTC:FieldType = 106 ;
             I_ACLWDNTC:MemoryOrder = "XY " ;
             I_ACLWDNTC:description = "BUCKET FOR DOWNWELLING CLEAR SKY LONGWAVE FLUX
AT TOP" ;
             I_ACLWDNTC:units = "J m-2" ;
             I_ACLWDNTC:stagger = "" ;
             I_ACLWDNTC:coordinates = "XLONG XLAT XTIME" ;
       int I_ACLWUPB(Time, south_north, west_east) ;
             I_ACLWUPB:FieldType = 106 ;
             I_ACLWUPB:MemoryOrder = "XY " ;
             I_ACLWUPB:description = "BUCKET FOR UPWELLING LONGWAVE FLUX AT BOTTOM" ;
             I_ACLWUPB:units = "J m-2" ;
             I_ACLWUPB:stagger = "" ;
             I_ACLWUPB:coordinates = "XLONG XLAT XTIME" ;
       int I_ACLWUPBC(Time, south_north, west_east) ;
             I_ACLWUPBC:FieldType = 106 ;
             I_ACLWUPBC:MemoryOrder = "XY " ;
             I_ACLWUPBC:description = "BUCKET FOR UPWELLING CLEAR SKY LONGWAVE FLUX AT
BOTTOM" ;
             I_ACLWUPBC:units = "J m-2" ;
             I_ACLWUPBC:stagger = "" ;
             I_ACLWUPBC:coordinates = "XLONG XLAT XTIME" ;
       int I_ACLWDNB(Time, south_north, west_east) ;
             I_ACLWDNB:FieldType = 106 ;
             I_ACLWDNB:MemoryOrder = "XY " ;
             I_ACLWDNB:description = "BUCKET FOR DOWNWELLING LONGWAVE FLUX AT BOTTOM" ;
             I_ACLWDNB:units = "J m-2" ;
             I_ACLWDNB:stagger = "" ;
             I_ACLWDNB:coordinates = "XLONG XLAT XTIME" ;
       int I_ACLWDNBC(Time, south_north, west_east) ;
             I_ACLWDNBC:FieldType = 106 ;
             I_ACLWDNBC:MemoryOrder = "XY " ;
             I_ACLWDNBC:description = "BUCKET FOR DOWNWELLING CLEAR SKY LONGWAVE FLUX
AT BOTTOM" ;
             I_ACLWDNBC:units = "J m-2" ;
             I_ACLWDNBC:stagger = "" ;
             I_ACLWDNBC:coordinates = "XLONG XLAT XTIME" ;
       float SWUPT(Time, south_north, west_east) ;
             SWUPT:FieldType = 104 ;
             SWUPT:MemoryOrder = "XY " ;
             SWUPT:description = "INSTANTANEOUS UPWELLING SHORTWAVE FLUX AT TOP" ;
             SWUPT:units = "W m-2" ;
             SWUPT:stagger = "" ;
             SWUPT:coordinates = "XLONG XLAT XTIME" ;
       float SWUPTC(Time, south_north, west_east) ;
             SWUPTC:FieldType = 104 ;
             SWUPTC:MemoryOrder = "XY " ;
```

```
            SWUPTC:description = "INSTANTANEOUS UPWELLING CLEAR SKY SHORTWAVE FLUX AT
TOP" ;
            SWUPTC:units = "W m-2" ;
            SWUPTC:stagger = "" ;
            SWUPTC:coordinates = "XLONG XLAT XTIME" ;
      float SWDNT(Time, south_north, west_east) ;
            SWDNT:FieldType = 104 ;
            SWDNT:MemoryOrder = "XY " ;
            SWDNT:description = "INSTANTANEOUS DOWNWELLING SHORTWAVE FLUX AT TOP" ;
            SWDNT:units = "W m-2" ;
            SWDNT:stagger = "" ;
            SWDNT:coordinates = "XLONG XLAT XTIME" ;
      float SWDNTC(Time, south_north, west_east) ;
            SWDNTC:FieldType = 104 ;
            SWDNTC:MemoryOrder = "XY " ;
            SWDNTC:description = "INSTANTANEOUS DOWNWELLING CLEAR SKY SHORTWAVE FLUX
AT TOP" ;
            SWDNTC:units = "W m-2" ;
            SWDNTC:stagger = "" ;
            SWDNTC:coordinates = "XLONG XLAT XTIME" ;
      float SWUPB(Time, south_north, west_east) ;
            SWUPB:FieldType = 104 ;
            SWUPB:MemoryOrder = "XY " ;
            SWUPB:description = "INSTANTANEOUS UPWELLING SHORTWAVE FLUX AT BOTTOM" ;
            SWUPB:units = "W m-2" ;
            SWUPB:stagger = "" ;
            SWUPB:coordinates = "XLONG XLAT XTIME" ;
      float SWUPBC(Time, south_north, west_east) ;
            SWUPBC:FieldType = 104 ;
            SWUPBC:MemoryOrder = "XY " ;
            SWUPBC:description = "INSTANTANEOUS UPWELLING CLEAR SKY SHORTWAVE FLUX AT
BOTTOM" ;
            SWUPBC:units = "W m-2" ;
            SWUPBC:stagger = "" ;
            SWUPBC:coordinates = "XLONG XLAT XTIME" ;
      float SWDNB(Time, south_north, west_east) ;
            SWDNB:FieldType = 104 ;
            SWDNB:MemoryOrder = "XY " ;
            SWDNB:description = "INSTANTANEOUS DOWNWELLING SHORTWAVE FLUX AT BOTTOM" ;
            SWDNB:units = "W m-2" ;
            SWDNB:stagger = "" ;
            SWDNB:coordinates = "XLONG XLAT XTIME" ;
      float SWDNBC(Time, south_north, west_east) ;
            SWDNBC:FieldType = 104 ;
            SWDNBC:MemoryOrder = "XY " ;
            SWDNBC:description = "INSTANTANEOUS DOWNWELLING CLEAR SKY SHORTWAVE FLUX
AT BOTTOM" ;
            SWDNBC:units = "W m-2" ;
            SWDNBC:stagger = "" ;
            SWDNBC:coordinates = "XLONG XLAT XTIME" ;
      float LWUPT(Time, south_north, west_east) ;
            LWUPT:FieldType = 104 ;
            LWUPT:MemoryOrder = "XY " ;
            LWUPT:description = "INSTANTANEOUS UPWELLING LONGWAVE FLUX AT TOP" ;
            LWUPT:units = "W m-2" ;
            LWUPT:stagger = "" ;
            LWUPT:coordinates = "XLONG XLAT XTIME" ;
      float LWUPTC(Time, south_north, west_east) ;
            LWUPTC:FieldType = 104 ;
            LWUPTC:MemoryOrder = "XY " ;
```

```
            LWUPTC:description = “INSTANTANEOUS UPWELLING CLEAR SKY LONGWAVE FLUX AT
TOP” ;
            LWUPTC:units = “W m-2” ;
            LWUPTC:stagger = “” ;
            LWUPTC:coordinates = “XLONG XLAT XTIME” ;
      float LWDNT(Time, south_north, west_east) ;
            LWDNT:FieldType = 104 ;
            LWDNT:MemoryOrder = “XY “ ;
            LWDNT:description = “INSTANTANEOUS DOWNWELLING LONGWAVE FLUX AT TOP” ;
            LWDNT:units = “W m-2” ;
            LWDNT:stagger = “” ;
            LWDNT:coordinates = “XLONG XLAT XTIME” ;
      float LWDNTC(Time, south_north, west_east) ;
            LWDNTC:FieldType = 104 ;
            LWDNTC:MemoryOrder = “XY “ ;
            LWDNTC:description = “INSTANTANEOUS DOWNWELLING CLEAR SKY LONGWAVE FLUX
AT TOP” ;
            LWDNTC:units = “W m-2” ;
            LWDNTC:stagger = “” ;
            LWDNTC:coordinates = “XLONG XLAT XTIME” ;
      float LWUPB(Time, south_north, west_east) ;
            LWUPB:FieldType = 104 ;
            LWUPB:MemoryOrder = “XY “ ;
            LWUPB:description = “INSTANTANEOUS UPWELLING LONGWAVE FLUX AT BOTTOM” ;
            LWUPB:units = “W m-2” ;
            LWUPB:stagger = “” ;
            LWUPB:coordinates = “XLONG XLAT XTIME” ;
      float LWUPBC(Time, south_north, west_east) ;
            LWUPBC:FieldType = 104 ;
            LWUPBC:MemoryOrder = “XY “ ;
            LWUPBC:description = “INSTANTANEOUS UPWELLING CLEAR SKY LONGWAVE FLUX AT
BOTTOM” ;
            LWUPBC:units = “W m-2” ;
            LWUPBC:stagger = “” ;
            LWUPBC:coordinates = “XLONG XLAT XTIME” ;
      float LWDNB(Time, south_north, west_east) ;
            LWDNB:FieldType = 104 ;
            LWDNB:MemoryOrder = “XY “ ;
            LWDNB:description = “INSTANTANEOUS DOWNWELLING LONGWAVE FLUX AT BOTTOM” ;
            LWDNB:units = “W m-2” ;
            LWDNB:stagger = “” ;
            LWDNB:coordinates = “XLONG XLAT XTIME” ;
      float LWDNBC(Time, south_north, west_east) ;
            LWDNBC:FieldType = 104 ;
            LWDNBC:MemoryOrder = “XY “ ;
            LWDNBC:description = “INSTANTANEOUS DOWNWELLING CLEAR SKY LONGWAVE FLUX
AT BOTTOM” ;
            LWDNBC:units = “W m-2” ;
            LWDNBC:stagger = “” ;
            LWDNBC:coordinates = “XLONG XLAT XTIME” ;
      float OLR(Time, south_north, west_east) ;
            OLR:FieldType = 104 ;
            OLR:MemoryOrder = “XY “ ;
            OLR:description = “TOA OUTGOING LONG WAVE” ;
            OLR:units = “W m-2” ;
            OLR:stagger = “” ;
            OLR:coordinates = “XLONG XLAT XTIME” ;
      float XLAT_U(Time, south_north, west_east_stag) ;
            XLAT_U:FieldType = 104 ;
            XLAT_U:MemoryOrder = “XY “ ;
```

```
        XLAT_U:description = "LATITUDE, SOUTH IS NEGATIVE" ;
        XLAT_U:units = "degree_north" ;
        XLAT_U:stagger = "X" ;
        XLAT_U:coordinates = "XLONG_U XLAT_U" ;
float XLONG_U(Time, south_north, west_east_stag) ;
        XLONG_U:FieldType = 104 ;
        XLONG_U:MemoryOrder = "XY " ;
        XLONG_U:description = "LONGITUDE, WEST IS NEGATIVE" ;
        XLONG_U:units = "degree_east" ;
        XLONG_U:stagger = "X" ;
        XLONG_U:coordinates = "XLONG_U XLAT_U" ;
float XLAT_V(Time, south_north_stag, west_east) ;
        XLAT_V:FieldType = 104 ;
        XLAT_V:MemoryOrder = "XY " ;
        XLAT_V:description = "LATITUDE, SOUTH IS NEGATIVE" ;
        XLAT_V:units = "degree_north" ;
        XLAT_V:stagger = "Y" ;
        XLAT_V:coordinates = "XLONG_V XLAT_V" ;
float XLONG_V(Time, south_north_stag, west_east) ;
        XLONG_V:FieldType = 104 ;
        XLONG_V:MemoryOrder = "XY " ;
        XLONG_V:description = "LONGITUDE, WEST IS NEGATIVE" ;
        XLONG_V:units = "degree_east" ;
        XLONG_V:stagger = "Y" ;
        XLONG_V:coordinates = "XLONG_V XLAT_V" ;
float ALBEDO(Time, south_north, west_east) ;
        ALBEDO:FieldType = 104 ;
        ALBEDO:MemoryOrder = "XY " ;
        ALBEDO:description = "ALBEDO" ;
        ALBEDO:units = "-" ;
        ALBEDO:stagger = "" ;
        ALBEDO:coordinates = "XLONG XLAT XTIME" ;
float CLAT(Time, south_north, west_east) ;
        CLAT:FieldType = 104 ;
        CLAT:MemoryOrder = "XY " ;
        CLAT:description = "COMPUTATIONAL GRID LATITUDE, SOUTH IS NEGATIVE" ;
        CLAT:units = "degree_north" ;
        CLAT:stagger = "" ;
        CLAT:coordinates = "XLONG XLAT XTIME" ;
float ALBBCK(Time, south_north, west_east) ;
        ALBBCK:FieldType = 104 ;
        ALBBCK:MemoryOrder = "XY " ;
        ALBBCK:description = "BACKGROUND ALBEDO" ;
        ALBBCK:units = "" ;
        ALBBCK:stagger = "" ;
        ALBBCK:coordinates = "XLONG XLAT XTIME" ;
float EMISS(Time, south_north, west_east) ;
        EMISS:FieldType = 104 ;
        EMISS:MemoryOrder = "XY " ;
        EMISS:description = "SURFACE EMISSIVITY" ;
        EMISS:units = "" ;
        EMISS:stagger = "" ;
        EMISS:coordinates = "XLONG XLAT XTIME" ;
float NOAHRES(Time, south_north, west_east) ;
        NOAHRES:FieldType = 104 ;
        NOAHRES:MemoryOrder = "XY " ;
        NOAHRES:description = "RESIDUAL OF THE NOAH SURFACE ENERGY BUDGET" ;
        NOAHRES:units = "W m{-2}" ;
        NOAHRES:stagger = "" ;
        NOAHRES:coordinates = "XLONG XLAT XTIME" ;
```

```
float TMN(Time, south_north, west_east) ;
       TMN:FieldType = 104 ;
       TMN:MemoryOrder = "XY " ;
       TMN:description = "SOIL TEMPERATURE AT LOWER BOUNDARY" ;
       TMN:units = "K" ;
       TMN:stagger = "" ;
       TMN:coordinates = "XLONG XLAT XTIME" ;
float XLAND(Time, south_north, west_east) ;
       XLAND:FieldType = 104 ;
       XLAND:MemoryOrder = "XY " ;
       XLAND:description = "LAND MASK (1 FOR LAND, 2 FOR WATER)" ;
       XLAND:units = "" ;
       XLAND:stagger = "" ;
       XLAND:coordinates = "XLONG XLAT XTIME" ;
float UST(Time, south_north, west_east) ;
       UST:FieldType = 104 ;
       UST:MemoryOrder = "XY " ;
       UST:description = "U* IN SIMILARITY THEORY" ;
       UST:units = "m s-1" ;
       UST:stagger = "" ;
       UST:coordinates = "XLONG XLAT XTIME" ;
float PBLH(Time, south_north, west_east) ;
       PBLH:FieldType = 104 ;
       PBLH:MemoryOrder = "XY " ;
       PBLH:description = "PBL HEIGHT" ;
       PBLH:units = "m" ;
       PBLH:stagger = "" ;
       PBLH:coordinates = "XLONG XLAT XTIME" ;
float HFX(Time, south_north, west_east) ;
       HFX:FieldType = 104 ;
       HFX:MemoryOrder = "XY " ;
       HFX:description = "UPWARD HEAT FLUX AT THE SURFACE" ;
       HFX:units = "W m-2" ;
       HFX:stagger = "" ;
       HFX:coordinates = "XLONG XLAT XTIME" ;
float QFX(Time, south_north, west_east) ;
       QFX:FieldType = 104 ;
       QFX:MemoryOrder = "XY " ;
       QFX:description = "UPWARD MOISTURE FLUX AT THE SURFACE" ;
       QFX:units = "kg m-2 s-1" ;
       QFX:stagger = "" ;
       QFX:coordinates = "XLONG XLAT XTIME" ;
float LH(Time, south_north, west_east) ;
       LH:FieldType = 104 ;
       LH:MemoryOrder = "XY " ;
       LH:description = "LATENT HEAT FLUX AT THE SURFACE" ;
       LH:units = "W m-2" ;
       LH:stagger = "" ;
       LH:coordinates = "XLONG XLAT XTIME" ;
float ACHFX(Time, south_north, west_east) ;
       ACHFX:FieldType = 104 ;
       ACHFX:MemoryOrder = "XY " ;
       ACHFX:description = "ACCUMULATED UPWARD HEAT FLUX AT THE SURFACE" ;
       ACHFX:units = "J m-2" ;
       ACHFX:stagger = "" ;
       ACHFX:coordinates = "XLONG XLAT XTIME" ;
float ACLHF(Time, south_north, west_east) ;
       ACLHF:FieldType = 104 ;
       ACLHF:MemoryOrder = "XY " ;
       ACLHF:description = "ACCUMULATED UPWARD LATENT HEAT FLUX AT THE SURFACE" ;
```

```
            ACLHF:units = “J m-2” ;
            ACLHF:stagger = “” ;
            ACLHF:coordinates = “XLONG XLAT XTIME” ;
      float SNOWC(Time, south_north, west_east) ;
            SNOWC:FieldType = 104 ;
            SNOWC:MemoryOrder = “XY “ ;
            SNOWC:description = “FLAG INDICATING SNOW COVERAGE (1 FOR SNOW COVER)” ;
            SNOWC:units = “” ;
            SNOWC:stagger = “” ;
            SNOWC:coordinates = “XLONG XLAT XTIME” ;
      float SR(Time, south_north, west_east) ;
            SR:FieldType = 104 ;
            SR:MemoryOrder = “XY “ ;
            SR:description = “fraction of frozen precipitation” ;
            SR:units = “-” ;
            SR:stagger = “” ;
            SR:coordinates = “XLONG XLAT XTIME” ;
      int SAVE_TOPO_FROM_REAL(Time) ;
            SAVE_TOPO_FROM_REAL:FieldType = 106 ;
            SAVE_TOPO_FROM_REAL:MemoryOrder = “0  “ ;
            SAVE_TOPO_FROM_REAL:description = “1=original topo from real/0=topo modi-
fied by WRF” ;
            SAVE_TOPO_FROM_REAL:units = “flag” ;
            SAVE_TOPO_FROM_REAL:stagger = “” ;
      int ISEEDARR_RAND_PERTURB(Time, bottom_top) ;
            ISEEDARR_RAND_PERTURB:FieldType = 106 ;
            ISEEDARR_RAND_PERTURB:MemoryOrder = “Z  “ ;
            ISEEDARR_RAND_PERTURB:description = “Array to hold seed for restart,
RAND_PERT” ;
            ISEEDARR_RAND_PERTURB:units = “” ;
            ISEEDARR_RAND_PERTURB:stagger = “” ;
      int ISEEDARR_SPPT(Time, bottom_top) ;
            ISEEDARR_SPPT:FieldType = 106 ;
            ISEEDARR_SPPT:MemoryOrder = “Z  “ ;
            ISEEDARR_SPPT:description = “Array to hold seed for restart, SPPT” ;
            ISEEDARR_SPPT:units = “” ;
            ISEEDARR_SPPT:stagger = “” ;
      int ISEEDARR_SKEBS(Time, bottom_top) ;
            ISEEDARR_SKEBS:FieldType = 106 ;
            ISEEDARR_SKEBS:MemoryOrder = “Z  “ ;
            ISEEDARR_SKEBS:description = “Array to hold seed for restart, SKEBS” ;
            ISEEDARR_SKEBS:units = “” ;
            ISEEDARR_SKEBS:stagger = “” ;
      float m11(Time, bottom_top, south_north, west_east) ;
            m11:FieldType = 104 ;
            m11:MemoryOrder = “XYZ” ;
            m11:description = “11 component of NBA subgrid stress tensor” ;
            m11:units = “m2 s-2” ;
            m11:stagger = “” ;
            m11:coordinates = “XLONG XLAT XTIME” ;
      float m22(Time, bottom_top, south_north, west_east) ;
            m22:FieldType = 104 ;
            m22:MemoryOrder = “XYZ” ;
            m22:description = “22 component of NBA subgrid stress tensor” ;
            m22:units = “m2 s-2” ;
            m22:stagger = “” ;
            m22:coordinates = “XLONG XLAT XTIME” ;
      float m33(Time, bottom_top, south_north, west_east) ;
            m33:FieldType = 104 ;
            m33:MemoryOrder = “XYZ” ;
```

```
        m33:description = "33 component of NBA subgrid stress tensor" ;
        m33:units = "m2 s-2" ;
        m33:stagger = "" ;
        m33:coordinates = "XLONG XLAT XTIME" ;
float m12(Time, bottom_top, south_north, west_east) ;
        m12:FieldType = 104 ;
        m12:MemoryOrder = "XYZ" ;
        m12:description = "12 component of NBA subgrid stress tensor" ;
        m12:units = "m2 s-2" ;
        m12:stagger = "" ;
        m12:coordinates = "XLONG XLAT XTIME" ;
float m13(Time, bottom_top, south_north, west_east) ;
        m13:FieldType = 104 ;
        m13:MemoryOrder = "XYZ" ;
        m13:description = "13 component of NBA subgrid stress tensor" ;
        m13:units = "m2 s-2" ;
        m13:stagger = "" ;
        m13:coordinates = "XLONG XLAT XTIME" ;
float m23(Time, bottom_top, south_north, west_east) ;
        m23:FieldType = 104 ;
        m23:MemoryOrder = "XYZ" ;
        m23:description = "23 component of NBA subgrid stress tensor" ;
        m23:units = "m2 s-2" ;
        m23:stagger = "" ;
        m23:coordinates = "XLONG XLAT XTIME" ;
float U_LS(Time, force_layers) ;
        U_LS:FieldType = 104 ;
        U_LS:MemoryOrder = "Z  " ;
        U_LS:description = "large-scale zonal wind velocity" ;
        U_LS:units = "m s-1" ;
        U_LS:stagger = "" ;
float U_LS_TEND(Time, force_layers) ;
        U_LS_TEND:FieldType = 104 ;
        U_LS_TEND:MemoryOrder = "Z  " ;
        U_LS_TEND:description = "tendency large-scale zonal wind velocity" ;
        U_LS_TEND:units = "m s-2" ;
        U_LS_TEND:stagger = "" ;
float V_LS(Time, force_layers) ;
        V_LS:FieldType = 104 ;
        V_LS:MemoryOrder = "Z  " ;
        V_LS:description = "large-scale meridional wind velocity" ;
        V_LS:units = "m s-1" ;
        V_LS:stagger = "" ;
float V_LS_TEND(Time, force_layers) ;
        V_LS_TEND:FieldType = 104 ;
        V_LS_TEND:MemoryOrder = "Z  " ;
        V_LS_TEND:description = "tendency large-scale meridional wind velocity" ;
        V_LS_TEND:units = "m s-2" ;
        V_LS_TEND:stagger = "" ;
float W_LS(Time, force_layers) ;
        W_LS:FieldType = 104 ;
        W_LS:MemoryOrder = "Z  " ;
        W_LS:description = "large-scale vertical velocity" ;
        W_LS:units = "m s-1" ;
        W_LS:stagger = "" ;
float W_LS_TEND(Time, force_layers) ;
        W_LS_TEND:FieldType = 104 ;
        W_LS_TEND:MemoryOrder = "Z  " ;
        W_LS_TEND:description = "tendency large-scale vertical velocity" ;
        W_LS_TEND:units = "m s-2" ;
```

```
        W_LS_TEND:stagger = “” ;
    float INV_TAU_S(Time, force_layers) ;
        INV_TAU_S:FieldType = 104 ;
        INV_TAU_S:MemoryOrder = “Z  “ ;
        INV_TAU_S:description = “inverse relaxation time for scalars” ;
        INV_TAU_S:units = “s-1” ;
        INV_TAU_S:stagger = “” ;
    float INV_TAU_M(Time, force_layers) ;
        INV_TAU_M:FieldType = 104 ;
        INV_TAU_M:MemoryOrder = “Z  “ ;
        INV_TAU_M:description = “inverse relaxation time for scalars” ;
        INV_TAU_M:units = “s-1” ;
        INV_TAU_M:stagger = “” ;
    float TH_ADV(Time, force_layers) ;
        TH_ADV:FieldType = 104 ;
        TH_ADV:MemoryOrder = “Z  “ ;
        TH_ADV:description = “tendency theta advection” ;
        TH_ADV:units = “K s-1” ;
        TH_ADV:stagger = “” ;
    float TH_ADV_TEND(Time, force_layers) ;
        TH_ADV_TEND:FieldType = 104 ;
        TH_ADV_TEND:MemoryOrder = “Z  “ ;
        TH_ADV_TEND:description = “tendency of tendency theta advection” ;
        TH_ADV_TEND:units = “K s-2” ;
        TH_ADV_TEND:stagger = “” ;
    float TH_RLX(Time, force_layers) ;
        TH_RLX:FieldType = 104 ;
        TH_RLX:MemoryOrder = “Z  “ ;
        TH_RLX:description = “theta relaxation” ;
        TH_RLX:units = “K” ;
        TH_RLX:stagger = “” ;
    float TH_RLX_TEND(Time, force_layers) ;
        TH_RLX_TEND:FieldType = 104 ;
        TH_RLX_TEND:MemoryOrder = “Z  “ ;
        TH_RLX_TEND:description = “tendency theta relaxation” ;
        TH_RLX_TEND:units = “K s-1” ;
        TH_RLX_TEND:stagger = “” ;
    float QV_ADV(Time, force_layers) ;
        QV_ADV:FieldType = 104 ;
        QV_ADV:MemoryOrder = “Z  “ ;
        QV_ADV:description = “tendency qv advection” ;
        QV_ADV:units = “kg kg-1 s-1” ;
        QV_ADV:stagger = “” ;
    float QV_ADV_TEND(Time, force_layers) ;
        QV_ADV_TEND:FieldType = 104 ;
        QV_ADV_TEND:MemoryOrder = “Z  “ ;
        QV_ADV_TEND:description = “tendency of tendency qv advection” ;
        QV_ADV_TEND:units = “kg kg-1 s-2” ;
        QV_ADV_TEND:stagger = “” ;
    float QV_RLX(Time, force_layers) ;
        QV_RLX:FieldType = 104 ;
        QV_RLX:MemoryOrder = “Z  “ ;
        QV_RLX:description = “qv relaxation” ;
        QV_RLX:units = “kg kg-1” ;
        QV_RLX:stagger = “” ;
    float QV_RLX_TEND(Time, force_layers) ;
        QV_RLX_TEND:FieldType = 104 ;
        QV_RLX_TEND:MemoryOrder = “Z  “ ;
        QV_RLX_TEND:description = “tendency qv relaxation” ;
```

```
        QV_RLX_TEND:units = “kg kg-1 s-1” ;
        QV_RLX_TEND:stagger = “” ;
float Z_LS(Time, force_layers) ;
        Z_LS:FieldType = 104 ;
        Z_LS:MemoryOrder = “Z  “ ;
        Z_LS:description = “z of large-scale forcings” ;
        Z_LS:units = “m” ;
        Z_LS:stagger = “” ;
float Z_LS_TEND(Time, force_layers) ;
        Z_LS_TEND:FieldType = 104 ;
        Z_LS_TEND:MemoryOrder = “Z  “ ;
        Z_LS_TEND:description = “tendency of z of large-scale forcings” ;
        Z_LS_TEND:units = “m s-1” ;
        Z_LS_TEND:stagger = “” ;
float PRE_SH_FLX(Time) ;
        PRE_SH_FLX:FieldType = 104 ;
        PRE_SH_FLX:MemoryOrder = “0  “ ;
        PRE_SH_FLX:description = “prescribed sensible heat flux” ;
        PRE_SH_FLX:units = “W m-2” ;
        PRE_SH_FLX:stagger = “” ;
float PRE_SH_FLX_TEND(Time) ;
        PRE_SH_FLX_TEND:FieldType = 104 ;
        PRE_SH_FLX_TEND:MemoryOrder = “0  “ ;
        PRE_SH_FLX_TEND:description = “tendency prescribed sensible heat flux” ;
        PRE_SH_FLX_TEND:units = “W m-2 s-1” ;
        PRE_SH_FLX_TEND:stagger = “” ;
float PRE_LH_FLX(Time) ;
        PRE_LH_FLX:FieldType = 104 ;
        PRE_LH_FLX:MemoryOrder = “0  “ ;
        PRE_LH_FLX:description = “prescribed latent heat flux” ;
        PRE_LH_FLX:units = “W m-2” ;
        PRE_LH_FLX:stagger = “” ;
float PRE_LH_FLX_TEND(Time) ;
        PRE_LH_FLX_TEND:FieldType = 104 ;
        PRE_LH_FLX_TEND:MemoryOrder = “0  “ ;
        PRE_LH_FLX_TEND:description = “tendency prescribed latent heat flux” ;
        PRE_LH_FLX_TEND:units = “W m-2 s-1” ;
        PRE_LH_FLX_TEND:stagger = “” ;
float PRE_ALBEDO(Time) ;
        PRE_ALBEDO:FieldType = 104 ;
        PRE_ALBEDO:MemoryOrder = “0  “ ;
        PRE_ALBEDO:description = “prescribed albedo” ;
        PRE_ALBEDO:units = “” ;
        PRE_ALBEDO:stagger = “” ;
float PRE_ALBEDO_TEND(Time) ;
        PRE_ALBEDO_TEND:FieldType = 104 ;
        PRE_ALBEDO_TEND:MemoryOrder = “0  “ ;
        PRE_ALBEDO_TEND:description = “tendency prescribed albedo” ;
        PRE_ALBEDO_TEND:units = “s-1” ;
        PRE_ALBEDO_TEND:stagger = “” ;
float PRE_TSK(Time) ;
        PRE_TSK:FieldType = 104 ;
        PRE_TSK:MemoryOrder = “0  “ ;
        PRE_TSK:description = “prescribed skin temperature” ;
        PRE_TSK:units = “K” ;
        PRE_TSK:stagger = “” ;
float PRE_TSK_TEND(Time) ;
        PRE_TSK_TEND:FieldType = 104 ;
        PRE_TSK_TEND:MemoryOrder = “0  “ ;
        PRE_TSK_TEND:description = “tendency prescribed skin temperature” ;
```

```
        PRE_TSK_TEND:units = "K s-1" ;
        PRE_TSK_TEND:stagger = "" ;
int NUM_MODES_AER(Time) ;
        NUM_MODES_AER:FieldType = 106 ;
        NUM_MODES_AER:MemoryOrder = "0  " ;
        NUM_MODES_AER:description = "Number of aerosol distribution modes" ;
        NUM_MODES_AER:units = "" ;
        NUM_MODES_AER:stagger = "" ;
float RM_AER1(Time) ;
        RM_AER1:FieldType = 104 ;
        RM_AER1:MemoryOrder = "0  " ;
        RM_AER1:description = "mode 1 aerosol geometric mean" ;
        RM_AER1:units = "um" ;
        RM_AER1:stagger = "" ;
float RM_AER1_TEND(Time) ;
        RM_AER1_TEND:FieldType = 104 ;
        RM_AER1_TEND:MemoryOrder = "0  " ;
        RM_AER1_TEND:description = "mode 1 tendency aerosol geometric mean" ;
        RM_AER1_TEND:units = "um s-1" ;
        RM_AER1_TEND:stagger = "" ;
float RM_AER2(Time) ;
        RM_AER2:FieldType = 104 ;
        RM_AER2:MemoryOrder = "0  " ;
        RM_AER2:description = "mode 2 aerosol geometric mean" ;
        RM_AER2:units = "um" ;
        RM_AER2:stagger = "" ;
float RM_AER2_TEND(Time) ;
        RM_AER2_TEND:FieldType = 104 ;
        RM_AER2_TEND:MemoryOrder = "0  " ;
        RM_AER2_TEND:description = "mode 2 tendency aerosol geometric mean" ;
        RM_AER2_TEND:units = "um s-1" ;
        RM_AER2_TEND:stagger = "" ;
float RM_AER3(Time) ;
        RM_AER3:FieldType = 104 ;
        RM_AER3:MemoryOrder = "0  " ;
        RM_AER3:description = "mode 3 aerosol geometric mean" ;
        RM_AER3:units = "um" ;
        RM_AER3:stagger = "" ;
float RM_AER3_TEND(Time) ;
        RM_AER3_TEND:FieldType = 104 ;
        RM_AER3_TEND:MemoryOrder = "0  " ;
        RM_AER3_TEND:description = "mode 3 tendency aerosol geometric mean" ;
        RM_AER3_TEND:units = "um s-1" ;
        RM_AER3_TEND:stagger = "" ;
float RM_AER4(Time) ;
        RM_AER4:FieldType = 104 ;
        RM_AER4:MemoryOrder = "0  " ;
        RM_AER4:description = "mode 4 aerosol geometric mean" ;
        RM_AER4:units = "um" ;
        RM_AER4:stagger = "" ;
float RM_AER4_TEND(Time) ;
        RM_AER4_TEND:FieldType = 104 ;
        RM_AER4_TEND:MemoryOrder = "0  " ;
        RM_AER4_TEND:description = "mode 4 tendency aerosol geometric mean" ;
        RM_AER4_TEND:units = "um s-1" ;
        RM_AER4_TEND:stagger = "" ;
float SIG_AER1(Time) ;
        SIG_AER1:FieldType = 104 ;
        SIG_AER1:MemoryOrder = "0  " ;
        SIG_AER1:description = "mode 1 aerosol standard deviation" ;
```

```
        SIG_AER1:units = “” ;
        SIG_AER1:stagger = “” ;
    float SIG_AER1_TEND(Time) ;
        SIG_AER1_TEND:FieldType = 104 ;
        SIG_AER1_TEND:MemoryOrder = “0  “ ;
        SIG_AER1_TEND:description = “mode 1 tendency aerosol standard deviation” ;
        SIG_AER1_TEND:units = “s-1” ;
        SIG_AER1_TEND:stagger = “” ;
    float SIG_AER2(Time) ;
        SIG_AER2:FieldType = 104 ;
        SIG_AER2:MemoryOrder = “0  “ ;
        SIG_AER2:description = “mode 2 aerosol standard deviation” ;
        SIG_AER2:units = “” ;
        SIG_AER2:stagger = “” ;
    float SIG_AER2_TEND(Time) ;
        SIG_AER2_TEND:FieldType = 104 ;
        SIG_AER2_TEND:MemoryOrder = “0  “ ;
        SIG_AER2_TEND:description = “mode 2 tendency aerosol standard deviation” ;
        SIG_AER2_TEND:units = “s-1” ;
        SIG_AER2_TEND:stagger = “” ;
    float SIG_AER3(Time) ;
        SIG_AER3:FieldType = 104 ;
        SIG_AER3:MemoryOrder = “0  “ ;
        SIG_AER3:description = “mode 3 aerosol standard deviation” ;
        SIG_AER3:units = “” ;
        SIG_AER3:stagger = “” ;
    float SIG_AER3_TEND(Time) ;
        SIG_AER3_TEND:FieldType = 104 ;
        SIG_AER3_TEND:MemoryOrder = “0  “ ;
        SIG_AER3_TEND:description = “mode 3 tendency aerosol standard deviation” ;
        SIG_AER3_TEND:units = “s-1” ;
        SIG_AER3_TEND:stagger = “” ;
    float SIG_AER4(Time) ;
        SIG_AER4:FieldType = 104 ;
        SIG_AER4:MemoryOrder = “0  “ ;
        SIG_AER4:description = “mode 4 aerosol standard deviation” ;
        SIG_AER4:units = “” ;
        SIG_AER4:stagger = “” ;
    float SIG_AER4_TEND(Time) ;
        SIG_AER4_TEND:FieldType = 104 ;
        SIG_AER4_TEND:MemoryOrder = “0  “ ;
        SIG_AER4_TEND:description = “mode 4 tendency aerosol standard deviation” ;
        SIG_AER4_TEND:units = “s-1” ;
        SIG_AER4_TEND:stagger = “” ;
    float NA_AER1(Time, aer_layers) ;
        NA_AER1:FieldType = 104 ;
        NA_AER1:MemoryOrder = “Z  “ ;
        NA_AER1:description = “mode 1 aerosol number concentration” ;
        NA_AER1:units = “cm-3” ;
        NA_AER1:stagger = “” ;
    float NA_AER1_TEND(Time, aer_layers) ;
        NA_AER1_TEND:FieldType = 104 ;
        NA_AER1_TEND:MemoryOrder = “Z  “ ;
        NA_AER1_TEND:description = “mode 1 tendency aerosol number concentration” ;
        NA_AER1_TEND:units = “cm-3 s-1” ;
        NA_AER1_TEND:stagger = “” ;
    float NA_AER2(Time, aer_layers) ;
        NA_AER2:FieldType = 104 ;
        NA_AER2:MemoryOrder = “Z  “ ;
```

```
        NA_AER2:description = "mode 2 aerosol number concentration" ;
        NA_AER2:units = "cm-3" ;
        NA_AER2:stagger = "" ;
    float NA_AER2_TEND(Time, aer_layers) ;
        NA_AER2_TEND:FieldType = 104 ;
        NA_AER2_TEND:MemoryOrder = "Z  " ;
        NA_AER2_TEND:description = "mode 2 tendency aerosol number concentration" ;
        NA_AER2_TEND:units = "cm-3 s-1" ;
        NA_AER2_TEND:stagger = "" ;
    float NA_AER3(Time, aer_layers) ;
        NA_AER3:FieldType = 104 ;
        NA_AER3:MemoryOrder = "Z  " ;
        NA_AER3:description = "mode 3 aerosol number concentration" ;
        NA_AER3:units = "cm-3" ;
        NA_AER3:stagger = "" ;
    float NA_AER3_TEND(Time, aer_layers) ;
        NA_AER3_TEND:FieldType = 104 ;
        NA_AER3_TEND:MemoryOrder = "Z  " ;
        NA_AER3_TEND:description = "mode 3 tendency aerosol number concentration" ;
        NA_AER3_TEND:units = "cm-3 s-1" ;
        NA_AER3_TEND:stagger = "" ;
    float NA_AER4(Time, aer_layers) ;
        NA_AER4:FieldType = 104 ;
        NA_AER4:MemoryOrder = "Z  " ;
        NA_AER4:description = "mode 4 aerosol number concentration" ;
        NA_AER4:units = "cm-3" ;
        NA_AER4:stagger = "" ;
    float NA_AER4_TEND(Time, aer_layers) ;
        NA_AER4_TEND:FieldType = 104 ;
        NA_AER4_TEND:MemoryOrder = "Z  " ;
        NA_AER4_TEND:description = "mode 4 tendency aerosol number concentration" ;
        NA_AER4_TEND:units = "cm-3 s-1" ;
        NA_AER4_TEND:stagger = "" ;
    float Z_AER(Time, aer_layers) ;
        Z_AER:FieldType = 104 ;
        Z_AER:MemoryOrder = "Z  " ;
        Z_AER:description = "z of aerosol data" ;
        Z_AER:units = "m" ;
        Z_AER:stagger = "" ;
    float Z_AER_TEND(Time, aer_layers) ;
        Z_AER_TEND:FieldType = 104 ;
        Z_AER_TEND:MemoryOrder = "Z  " ;
        Z_AER_TEND:description = "tendency z of aerosol data" ;
        Z_AER_TEND:units = "m s-1" ;
        Z_AER_TEND:stagger = "" ;
    float HYGRO_AER1(Time) ;
        HYGRO_AER1:FieldType = 104 ;
        HYGRO_AER1:MemoryOrder = "0  " ;
        HYGRO_AER1:description = "mode 1 hygroscopicity for aerosol" ;
        HYGRO_AER1:units = "" ;
        HYGRO_AER1:stagger = "" ;
    float HYGRO_AER1_TEND(Time) ;
        HYGRO_AER1_TEND:FieldType = 104 ;
        HYGRO_AER1_TEND:MemoryOrder = "0  " ;
        HYGRO_AER1_TEND:description = "mode 1 tendency hygroscopicity for aero-
sol" ;
        HYGRO_AER1_TEND:units = "" ;
        HYGRO_AER1_TEND:stagger = "" ;
    float HYGRO_AER2(Time) ;
        HYGRO_AER2:FieldType = 104 ;
```

```
            HYGRO_AER2:MemoryOrder = "0  " ;
            HYGRO_AER2:description = "mode 2 hygroscopicity for aerosol" ;
            HYGRO_AER2:units = "" ;
            HYGRO_AER2:stagger = "" ;
      float HYGRO_AER2_TEND(Time) ;
            HYGRO_AER2_TEND:FieldType = 104 ;
            HYGRO_AER2_TEND:MemoryOrder = "0  " ;
            HYGRO_AER2_TEND:description = "mode 2 tendency hygroscopicity for aero-
sol" ;
            HYGRO_AER2_TEND:units = "" ;
            HYGRO_AER2_TEND:stagger = "" ;
      float HYGRO_AER3(Time) ;
            HYGRO_AER3:FieldType = 104 ;
            HYGRO_AER3:MemoryOrder = "0  " ;
            HYGRO_AER3:description = "mode 3 hygroscopicity for aerosol" ;
            HYGRO_AER3:units = "" ;
            HYGRO_AER3:stagger = "" ;
      float HYGRO_AER3_TEND(Time) ;
            HYGRO_AER3_TEND:FieldType = 104 ;
            HYGRO_AER3_TEND:MemoryOrder = "0  " ;
            HYGRO_AER3_TEND:description = "mode 3 tendency hygroscopicity for aero-
sol" ;
            HYGRO_AER3_TEND:units = "" ;
            HYGRO_AER3_TEND:stagger = "" ;
      float HYGRO_AER4(Time) ;
            HYGRO_AER4:FieldType = 104 ;
            HYGRO_AER4:MemoryOrder = "0  " ;
            HYGRO_AER4:description = "mode 4 hygroscopicity for aerosol" ;
            HYGRO_AER4:units = "" ;
            HYGRO_AER4:stagger = "" ;
      float HYGRO_AER4_TEND(Time) ;
            HYGRO_AER4_TEND:FieldType = 104 ;
            HYGRO_AER4_TEND:MemoryOrder = "0  " ;
            HYGRO_AER4_TEND:description = "mode 4 tendency hygroscopicity for aero-
sol" ;
            HYGRO_AER4_TEND:units = "" ;
            HYGRO_AER4_TEND:stagger = "" ;
      float NA_AER1_GRD(Time, bottom_top) ;
            NA_AER1_GRD:FieldType = 104 ;
            NA_AER1_GRD:MemoryOrder = "Z  " ;
            NA_AER1_GRD:description = "mode 1 aerosol number concentration at half
level" ;
            NA_AER1_GRD:units = "cm-3" ;
            NA_AER1_GRD:stagger = "" ;
      float NA_AER2_GRD(Time, bottom_top) ;
            NA_AER2_GRD:FieldType = 104 ;
            NA_AER2_GRD:MemoryOrder = "Z  " ;
            NA_AER2_GRD:description = "mode 2 aerosol number concentration at half
level" ;
            NA_AER2_GRD:units = "cm-3" ;
            NA_AER2_GRD:stagger = "" ;
      float NA_AER3_GRD(Time, bottom_top) ;
            NA_AER3_GRD:FieldType = 104 ;
            NA_AER3_GRD:MemoryOrder = "Z  " ;
            NA_AER3_GRD:description = "mode 3 aerosol number concentration at half
level" ;
            NA_AER3_GRD:units = "cm-3" ;
            NA_AER3_GRD:stagger = "" ;
```

```
	float NA_AER4_GRD(Time, bottom_top) ;
		NA_AER4_GRD:FieldType = 104 ;
		NA_AER4_GRD:MemoryOrder = "Z  " ;
		NA_AER4_GRD:description = "mode 4 aerosol number concentration at half 
level" ;
		NA_AER4_GRD:units = "cm-3" ;
		NA_AER4_GRD:stagger = "" ;
	float RULSTEN(Time, bottom_top, south_north, west_east) ;
		RULSTEN:FieldType = 104 ;
		RULSTEN:MemoryOrder = "XYZ" ;
		RULSTEN:description = "coupled u tendency due to LS forcing" ;
		RULSTEN:units = "Pa m s-2" ;
		RULSTEN:stagger = "" ;
		RULSTEN:coordinates = "XLONG XLAT XTIME" ;
	float RVLSTEN(Time, bottom_top, south_north, west_east) ;
		RVLSTEN:FieldType = 104 ;
		RVLSTEN:MemoryOrder = "XYZ" ;
		RVLSTEN:description = "coupled v tendency due to LS forcing" ;
		RVLSTEN:units = "Pa m s-2" ;
		RVLSTEN:stagger = "" ;
		RVLSTEN:coordinates = "XLONG XLAT XTIME" ;
	float RTHLSTEN(Time, bottom_top, south_north, west_east) ;
		RTHLSTEN:FieldType = 104 ;
		RTHLSTEN:MemoryOrder = "XYZ" ;
		RTHLSTEN:description = "coupled theta tendency due to LS forcing" ;
		RTHLSTEN:units = "Pa K s-1" ;
		RTHLSTEN:stagger = "" ;
		RTHLSTEN:coordinates = "XLONG XLAT XTIME" ;
	float RQVLSTEN(Time, bottom_top, south_north, west_east) ;
		RQVLSTEN:FieldType = 104 ;
		RQVLSTEN:MemoryOrder = "XYZ" ;
		RQVLSTEN:description = "coupled Q_V tendency due to LS forcing" ;
		RQVLSTEN:units = "Pa kg kg-1 s-1" ;
		RQVLSTEN:stagger = "" ;
		RQVLSTEN:coordinates = "XLONG XLAT XTIME" ;
	float W_DTHDZ(Time, bottom_top, south_north, west_east) ;
		W_DTHDZ:FieldType = 104 ;
		W_DTHDZ:MemoryOrder = "XYZ" ;
		W_DTHDZ:description = "th tendency due to LS vertical adv" ;
		W_DTHDZ:units = "K s-1" ;
		W_DTHDZ:stagger = "" ;
		W_DTHDZ:coordinates = "XLONG XLAT XTIME" ;
	float W_DQVDZ(Time, bottom_top, south_north, west_east) ;
		W_DQVDZ:FieldType = 104 ;
		W_DQVDZ:MemoryOrder = "XYZ" ;
		W_DQVDZ:description = "qv tendency due to LS vertical adv" ;
		W_DQVDZ:units = "kg kg-1 s-1" ;
		W_DQVDZ:stagger = "" ;
		W_DQVDZ:coordinates = "XLONG XLAT XTIME" ;
	float W_DUDZ(Time, bottom_top, south_north, west_east) ;
		W_DUDZ:FieldType = 104 ;
		W_DUDZ:MemoryOrder = "XYZ" ;
		W_DUDZ:description = "u tendency due to LS vertical adv" ;
		W_DUDZ:units = "m s-2" ;
		W_DUDZ:stagger = "" ;
		W_DUDZ:coordinates = "XLONG XLAT XTIME" ;
	float W_DVDZ(Time, bottom_top, south_north, west_east) ;
		W_DVDZ:FieldType = 104 ;
		W_DVDZ:MemoryOrder = "XYZ" ;
```

```
        W_DVDZ:description = “v tendency due to LS vertical adv” ;
        W_DVDZ:units = “m s-2” ;
        W_DVDZ:stagger = “” ;
        W_DVDZ:coordinates = “XLONG XLAT XTIME” ;
float THDT_LSHOR(Time, bottom_top) ;
        THDT_LSHOR:FieldType = 104 ;
        THDT_LSHOR:MemoryOrder = “Z  “ ;
        THDT_LSHOR:description = “th tendency due to LS horizontal adv” ;
        THDT_LSHOR:units = “K s-1” ;
        THDT_LSHOR:stagger = “” ;
float QVDT_LSHOR(Time, bottom_top) ;
        QVDT_LSHOR:FieldType = 104 ;
        QVDT_LSHOR:MemoryOrder = “Z  “ ;
        QVDT_LSHOR:description = “qv tendency due to LS horizontal adv” ;
        QVDT_LSHOR:units = “kg kg-1 s-1” ;
        QVDT_LSHOR:stagger = “” ;
float THDT_LSRLX(Time, bottom_top) ;
        THDT_LSRLX:FieldType = 104 ;
        THDT_LSRLX:MemoryOrder = “Z  “ ;
        THDT_LSRLX:description = “th tendency due to relaxation to LS” ;
        THDT_LSRLX:units = “K s-1” ;
        THDT_LSRLX:stagger = “” ;
float QVDT_LSRLX(Time, bottom_top) ;
        QVDT_LSRLX:FieldType = 104 ;
        QVDT_LSRLX:MemoryOrder = “Z  “ ;
        QVDT_LSRLX:description = “qv tendency due to relaxation to LS” ;
        QVDT_LSRLX:units = “kg kg-1 s-1” ;
        QVDT_LSRLX:stagger = “” ;
float UDT_LSRLX(Time, bottom_top) ;
        UDT_LSRLX:FieldType = 104 ;
        UDT_LSRLX:MemoryOrder = “Z  “ ;
        UDT_LSRLX:description = “u tendency due to relaxation to LS” ;
        UDT_LSRLX:units = “m s-2” ;
        UDT_LSRLX:stagger = “” ;
float VDT_LSRLX(Time, bottom_top) ;
        VDT_LSRLX:FieldType = 104 ;
        VDT_LSRLX:MemoryOrder = “Z  “ ;
        VDT_LSRLX:description = “v tendency due to relaxation to LS” ;
        VDT_LSRLX:units = “m s-2” ;
        VDT_LSRLX:stagger = “” ;
float EFFCS(Time, bottom_top, south_north, west_east) ;
        EFFCS:FieldType = 104 ;
        EFFCS:MemoryOrder = “XYZ” ;
        EFFCS:description = “CLOUD DROPLET EFFECTIVE RADIUS” ;
        EFFCS:units = “micron” ;
        EFFCS:stagger = “” ;
        EFFCS:coordinates = “XLONG XLAT XTIME” ;
float LWF0(Time, bottom_top, south_north, west_east) ;
        LWF0:FieldType = 104 ;
        LWF0:MemoryOrder = “XYZ” ;
        LWF0:description = “Net LW radiative flux” ;
        LWF0:units = “W m-2” ;
        LWF0:stagger = “” ;
        LWF0:coordinates = “XLONG XLAT XTIME” ;
float LWF1(Time, bottom_top, south_north, west_east) ;
        LWF1:FieldType = 104 ;
        LWF1:MemoryOrder = “XYZ” ;
        LWF1:description = “Net LW radiative flux, term1” ;
        LWF1:units = “W m-2” ;
```

```
		LWF1:stagger = “” ;
		LWF1:coordinates = “XLONG XLAT XTIME” ;
	float LWF2(Time, bottom_top, south_north, west_east) ;
		LWF2:FieldType = 104 ;
		LWF2:MemoryOrder = “XYZ” ;
		LWF2:description = “Net LW radiative flux, term2” ;
		LWF2:units = “W m-2” ;
		LWF2:stagger = “” ;
		LWF2:coordinates = “XLONG XLAT XTIME” ;
	float LWF3(Time, bottom_top, south_north, west_east) ;
		LWF3:FieldType = 104 ;
		LWF3:MemoryOrder = “XYZ” ;
		LWF3:description = “Net LW radiative flux, term3” ;
		LWF3:units = “W m-2” ;
		LWF3:stagger = “” ;
		LWF3:coordinates = “XLONG XLAT XTIME” ;
	float ZI_QT8(Time, south_north, west_east) ;
		ZI_QT8:FieldType = 104 ;
		ZI_QT8:MemoryOrder = “XY “ ;
		ZI_QT8:description = “zi defined by qt” ;
		ZI_QT8:units = “m” ;
		ZI_QT8:stagger = “” ;
		ZI_QT8:coordinates = “XLONG XLAT XTIME” ;
	float SEDFQC(Time, bottom_top, south_north, west_east) ;
		SEDFQC:FieldType = 104 ;
		SEDFQC:MemoryOrder = “XYZ” ;
		SEDFQC:description = “sedimentation flux of cloud water” ;
		SEDFQC:units = “kg m-2 s-1” ;
		SEDFQC:stagger = “” ;
		SEDFQC:coordinates = “XLONG XLAT XTIME” ;
	float SEDFQR(Time, bottom_top, south_north, west_east) ;
		SEDFQR:FieldType = 104 ;
		SEDFQR:MemoryOrder = “XYZ” ;
		SEDFQR:description = “sedimentation flux of rain water” ;
		SEDFQR:units = “kg m-2 s-1” ;
		SEDFQR:stagger = “” ;
		SEDFQR:coordinates = “XLONG XLAT XTIME” ;
	float QVDT_PR(Time, bottom_top, south_north, west_east) ;
		QVDT_PR:FieldType = 104 ;
		QVDT_PR:MemoryOrder = “XYZ” ;
		QVDT_PR:description = “Production rate of vapor by conversion to rain” ;
		QVDT_PR:units = “” ;
		QVDT_PR:stagger = “” ;
		QVDT_PR:coordinates = “XLONG XLAT XTIME” ;
	float QVDT_COND(Time, bottom_top, south_north, west_east) ;
		QVDT_COND:FieldType = 104 ;
		QVDT_COND:MemoryOrder = “XYZ” ;
		QVDT_COND:description = “Production rate of vapor by conversion to rain” ;
		QVDT_COND:units = “” ;
		QVDT_COND:stagger = “” ;
		QVDT_COND:coordinates = “XLONG XLAT XTIME” ;
	float QCDT_PR(Time, bottom_top, south_north, west_east) ;
		QCDT_PR:FieldType = 104 ;
		QCDT_PR:MemoryOrder = “XYZ” ;
		QCDT_PR:description = “Production rate of vapor by conversion to rain” ;
		QCDT_PR:units = “” ;
		QCDT_PR:stagger = “” ;
		QCDT_PR:coordinates = “XLONG XLAT XTIME” ;
```

```
        float QCDT_SED(Time, bottom_top, south_north, west_east) ;
                QCDT_SED:FieldType = 104 ;
                QCDT_SED:MemoryOrder = "XYZ" ;
                QCDT_SED:description = "Tendency of cloud water due to sedimentation" ;
                QCDT_SED:units = "" ;
                QCDT_SED:stagger = "" ;
                QCDT_SED:coordinates = "XLONG XLAT XTIME" ;
        float QRDT_SED(Time, bottom_top, south_north, west_east) ;
                QRDT_SED:FieldType = 104 ;
                QRDT_SED:MemoryOrder = "XYZ" ;
                QRDT_SED:description = "Tendency of rain water due to sedimentation" ;
                QRDT_SED:units = "" ;
                QRDT_SED:stagger = "" ;
                QRDT_SED:coordinates = "XLONG XLAT XTIME" ;
        float SMAXACT(Time, bottom_top, south_north, west_east) ;
                SMAXACT:FieldType = 104 ;
                SMAXACT:MemoryOrder = "XYZ" ;
                SMAXACT:description = "Maximum supersaturation in Morrison microphysics" ;
                SMAXACT:units = "" ;
                SMAXACT:stagger = "" ;
                SMAXACT:coordinates = "XLONG XLAT XTIME" ;
        float RMINACT(Time, bottom_top, south_north, west_east) ;
                RMINACT:FieldType = 104 ;
                RMINACT:MemoryOrder = "XYZ" ;
                RMINACT:description = "Minimum activated radius in Morrison microphysics" ;
                RMINACT:units = "" ;
                RMINACT:stagger = "" ;
                RMINACT:coordinates = "XLONG XLAT XTIME" ;
        float LANDMASK(Time, south_north, west_east) ;
                LANDMASK:FieldType = 104 ;
                LANDMASK:MemoryOrder = "XY " ;
                LANDMASK:description = "LAND MASK (1 FOR LAND, 0 FOR WATER)" ;
                LANDMASK:units = "" ;
                LANDMASK:stagger = "" ;
                LANDMASK:coordinates = "XLONG XLAT XTIME" ;
        float LAKEMASK(Time, south_north, west_east) ;
                LAKEMASK:FieldType = 104 ;
                LAKEMASK:MemoryOrder = "XY " ;
                LAKEMASK:description = "LAKE MASK (1 FOR LAKE, 0 FOR NON-LAKE)" ;
                LAKEMASK:units = "" ;
                LAKEMASK:stagger = "" ;
                LAKEMASK:coordinates = "XLONG XLAT XTIME" ;
        float SST(Time, south_north, west_east) ;
                SST:FieldType = 104 ;
                SST:MemoryOrder = "XY " ;
                SST:description = "SEA SURFACE TEMPERATURE" ;
                SST:units = "K" ;
                SST:stagger = "" ;
                SST:coordinates = "XLONG XLAT XTIME" ;
        float SST_INPUT(Time, south_north, west_east) ;
                SST_INPUT:FieldType = 104 ;
                SST_INPUT:MemoryOrder = "XY " ;
                SST_INPUT:description = "SEA SURFACE TEMPERATURE FROM WRFLOWINPUT FILE" ;
                SST_INPUT:units = "K" ;
                SST_INPUT:stagger = "" ;
                SST_INPUT:coordinates = "XLONG XLAT XTIME" ;

// global attributes:
                :TITLE = " OUTPUT FROM WRF V3.8.1 MODEL" ;
                :START_DATE = "2016-06-10_12:00:00" ;
```

```
:SIMULATION_START_DATE = “2016-06-10_12:00:00” ;
:WEST-EAST_GRID_DIMENSION = 145 ;
:SOUTH-NORTH_GRID_DIMENSION = 145 ;
:BOTTOM-TOP_GRID_DIMENSION = 227 ;
:DX = 100.f ;
:DY = 100.f ;
:SKEBS_ON = 0 ;
:SPEC_BDY_FINAL_MU = 1 ;
:USE_Q_DIABATIC = 0 ;
:GRIDTYPE = “C” ;
:DIFF_OPT = 2 ;
:KM_OPT = 2 ;
:DAMP_OPT = 3 ;
:DAMPCOEF = 0.2f ;
:KHDIF = 1.f ;
:KVDIF = 1.f ;
:MP_PHYSICS = 50 ;
:RA_LW_PHYSICS = 4 ;
:RA_SW_PHYSICS = 4 ;
:SF_SFCLAY_PHYSICS = 1 ;
:SF_SURFACE_PHYSICS = 1 ;
:BL_PBL_PHYSICS = 0 ;
:CU_PHYSICS = 0 ;
:SF_LAKE_PHYSICS = 0 ;
:SURFACE_INPUT_SOURCE = 3 ;
:SST_UPDATE = 0 ;
:GRID_FDDA = 0 ;
:GFDDA_INTERVAL_M = 0 ;
:GFDDA_END_H = 0 ;
:GRID_SFDDA = 0 ;
:SGFDDA_INTERVAL_M = 0 ;
:SGFDDA_END_H = 0 ;
:HYPSOMETRIC_OPT = 1 ;
:USE_THETA_M = 1 ;
:SF_URBAN_PHYSICS = 0 ;
:SHCU_PHYSICS = 0 ;
:MFSHCONV = 0 ;
:FEEDBACK = 0 ;
:SMOOTH_OPTION = 0 ;
:SWRAD_SCAT = 1.f ;
:W_DAMPING = 0 ;
:RADT = 1.f ;
:BLDT = 0.f ;
:CUDT = 0.f ;
:AER_OPT = 0 ;
:SWINT_OPT = 0 ;
:AER_TYPE = 1 ;
:AER_AOD550_OPT = 1 ;
:AER_ANGEXP_OPT = 1 ;
:AER_SSA_OPT = 1 ;
:AER_ASY_OPT = 1 ;
:AER_AOD550_VAL = 0.12f ;
:AER_ANGEXP_VAL = 1.3f ;
:AER_SSA_VAL = 1.401298e-45f ;
:AER_ASY_VAL = 1.401298e-45f ;
:MOIST_ADV_OPT = 2 ;
:SCALAR_ADV_OPT = 2 ;
:TKE_ADV_OPT = 2 ;
:DIFF_6TH_OPT = 0 ;
:DIFF_6TH_FACTOR = 0.12f ;
```

```
        :OBS_NUDGE_OPT = 0 ;
        :BUCKET_MM = -1.f ;
        :BUCKET_J = -1.f ;
        :PREC_ACC_DT = 0.f ;
        :SF_OCEAN_PHYSICS = 0 ;
        :ISFTCFLX = 0 ;
        :ISHALLOW = 0 ;
        :ISFFLX = 11 ;
        :ICLOUD = 1 ;
        :ICLOUD_CU = 0 ;
        :TRACER_PBLMIX = 1 ;
        :SCALAR_PBLMIX = 0 ;
        :YSU_TOPDOWN_PBLMIX = 0 ;
        :GRAV_SETTLING = 0 ;
        :DFI_OPT = 0 ;
        :SIMULATION_INITIALIZATION_TYPE = "IDEALIZED DATA" ;
        :WEST-EAST_PATCH_START_UNSTAG = 1 ;
        :WEST-EAST_PATCH_END_UNSTAG = 144 ;
        :WEST-EAST_PATCH_START_STAG = 1 ;
        :WEST-EAST_PATCH_END_STAG = 145 ;
        :SOUTH-NORTH_PATCH_START_UNSTAG = 1 ;
        :SOUTH-NORTH_PATCH_END_UNSTAG = 144 ;
        :SOUTH-NORTH_PATCH_START_STAG = 1 ;
        :SOUTH-NORTH_PATCH_END_STAG = 145 ;
        :BOTTOM-TOP_PATCH_START_UNSTAG = 1 ;
        :BOTTOM-TOP_PATCH_END_UNSTAG = 226 ;
        :BOTTOM-TOP_PATCH_START_STAG = 1 ;
        :BOTTOM-TOP_PATCH_END_STAG = 227 ;
        :GRID_ID = 1 ;
        :PARENT_ID = 0 ;
        :I_PARENT_START = 0 ;
        :J_PARENT_START = 0 ;
        :PARENT_GRID_RATIO = 1 ;
        :DT = 0.5f ;
        :CEN_LAT = 0.f ;
        :CEN_LON = 0.f ;
        :TRUELAT1 = 0.f ;
        :TRUELAT2 = 0.f ;
        :MOAD_CEN_LAT = 0.f ;
        :STAND_LON = 0.f ;
        :POLE_LAT = 0.f ;
        :POLE_LON = 0.f ;
        :GMT = 0.f ;
        :JULYR = 0 ;
        :JULDAY = 1 ;
        :MAP_PROJ = 0 ;
        :MAP_PROJ_CHAR = "Cartesian" ;
        :MMINLU = "" ;
        :NUM_LAND_CAT = 21 ;
        :ISWATER = 16 ;
        :ISLAKE = 0 ;
        :ISICE = 0 ;
        :ISURBAN = 0 ;
        :ISOILWATER = 0 ;
}
```

File: .../20160610/sim0001/raw_model/wrfstat_d01_2016-06-10_12:00:00.nc

Description: WRF LES statistical output for domain-wide profiles. Values are time averaged over 10-minute periods with 1-minute sampling and the time label is at the end of the averaging period. All output times from the entire simulation are included in one file. Variable names with the suffixes CSP, CST, CSV, and CSS represent horizontally averaged profiles, time series of horizontally averaged surface or column integrated quantities, cell-specific volume quantities only averaged in time, and time averaged slab quantities, respectively. CSV values represent the full 3D volume and are related to CSP values through horizontal averaging. Likewise, CSS values are horizontal slabs, e.g., of surface values, which relate to CST values through horizontal averaging.

```
netcdf wrfstat_d01_2016-06-10_12\:00\:00 {
dimensions:
	Time = UNLIMITED ; // (91 currently)
	DateStrLen = 19 ;
	bottom_top = 226 ;
	bottom_top_stag = 227 ;
	west_east = 144 ;
	south_north = 144 ;
	west_east_stag = 145 ;
	south_north_stag = 145 ;
variables:
	char Times(Time, DateStrLen) ;
	float XTIME(Time) ;
		XTIME:FieldType = 104 ;
		XTIME:MemoryOrder = "0  " ;
		XTIME:description = "minutes since 2016-06-10 12:00:00" ;
		XTIME:stagger = "" ;
		XTIME:units = "minutes since 2016-06-10 12:00:00" ;
	float CST_PRECT(Time) ;
		CST_PRECT:FieldType = 104 ;
		CST_PRECT:MemoryOrder = "0  " ;
		CST_PRECT:description = "Total precipitation at surface" ;
		CST_PRECT:stagger = "" ;
		CST_PRECT:units = "mm/sec" ;
	float CST_CLDLOW(Time) ;
		CST_CLDLOW:FieldType = 104 ;
		CST_CLDLOW:MemoryOrder = "0  " ;
		CST_CLDLOW:description = "Fractional low-cloud cover (<5 km)" ;
		CST_CLDLOW:units = "(0-1)" ;
		CST_CLDLOW:stagger = "" ;
	float CST_CLDTOT(Time) ;
		CST_CLDTOT:FieldType = 104 ;
		CST_CLDTOT:MemoryOrder = "0  " ;
		CST_CLDTOT:description = "Fractional cloud cover" ;
		CST_CLDTOT:units = "(0-1)" ;
		CST_CLDTOT:stagger = "" ;
	float CST_LWP(Time) ;
		CST_LWP:FieldType = 104 ;
		CST_LWP:MemoryOrder = "0  " ;
		CST_LWP:description = "Vertical integrated liquid water path (based on
ql)" ;
		CST_LWP:units = "kg/m^2" ;
		CST_LWP:stagger = "" ;
	float CST_IWP(Time) ;
		CST_IWP:FieldType = 104 ;
		CST_IWP:MemoryOrder = "0  " ;
```

```
        CST_IWP:description = "Vertical integrated ice water path (based on qf)" ;
        CST_IWP:units = "kg/m^2" ;
        CST_IWP:stagger = "" ;
    float CST_PRECW(Time) ;
        CST_PRECW:FieldType = 104 ;
        CST_PRECW:MemoryOrder = "0  " ;
        CST_PRECW:description = "Vertical integrated water vapor" ;
        CST_PRECW:units = "kg/m^2" ;
        CST_PRECW:stagger = "" ;
    float CST_TKE(Time) ;
        CST_TKE:FieldType = 104 ;
        CST_TKE:MemoryOrder = "0  " ;
        CST_TKE:description = "Vertical integrated TKE" ;
        CST_TKE:units = "kg/s^2" ;
        CST_TKE:stagger = "" ;
    float CST_TSAIR(Time) ;
        CST_TSAIR:FieldType = 104 ;
        CST_TSAIR:MemoryOrder = "0  " ;
        CST_TSAIR:description = "Surface air temperature" ;
        CST_TSAIR:units = "K" ;
        CST_TSAIR:stagger = "" ;
    float CST_PS(Time) ;
        CST_PS:FieldType = 104 ;
        CST_PS:MemoryOrder = "0  " ;
        CST_PS:description = "Surface pressure" ;
        CST_PS:units = "Pa" ;
        CST_PS:stagger = "" ;
    float CST_SH(Time) ;
        CST_SH:FieldType = 104 ;
        CST_SH:MemoryOrder = "0  " ;
        CST_SH:units = "W/m^2" ;
        CST_SH:stagger = "" ;
        CST_SH:description = "Surface sensible heat flux" ;
    float CST_LH(Time) ;
        CST_LH:FieldType = 104 ;
        CST_LH:MemoryOrder = "0  " ;
        CST_LH:units = "W/m^2" ;
        CST_LH:stagger = "" ;
        CST_LH:description = "Surface latent heat flux" ;
    float CST_FSNTC(Time) ;
        CST_FSNTC:FieldType = 104 ;
        CST_FSNTC:MemoryOrder = "0  " ;
        CST_FSNTC:description = "TOA SW net upward clear-sky radiation" ;
        CST_FSNTC:units = "W/m^2" ;
        CST_FSNTC:stagger = "" ;
    float CST_FSNT(Time) ;
        CST_FSNT:FieldType = 104 ;
        CST_FSNT:MemoryOrder = "0  " ;
        CST_FSNT:description = "TOA SW net upward total-sky radiation" ;
        CST_FSNT:units = "W/m^2" ;
        CST_FSNT:stagger = "" ;
    float CST_FLNTC(Time) ;
        CST_FLNTC:FieldType = 104 ;
        CST_FLNTC:MemoryOrder = "0  " ;
        CST_FLNTC:description = "TOA LW (net) upward clear-sky radiation" ;
        CST_FLNTC:units = "W/m^2" ;
        CST_FLNTC:stagger = "" ;
    float CST_FLNT(Time) ;
        CST_FLNT:FieldType = 104 ;
        CST_FLNT:MemoryOrder = "0  " ;
```

```
        CST_FLNT:description = "TOA LW (net) upward total-sky radiation" ;
        CST_FLNT:units = "W/m^2" ;
        CST_FLNT:stagger = "" ;
float CST_FSNSC(Time) ;
        CST_FSNSC:FieldType = 104 ;
        CST_FSNSC:MemoryOrder = "0  " ;
        CST_FSNSC:description = "Surface SW net upward clear-sky radiation" ;
        CST_FSNSC:units = "W/m^2" ;
        CST_FSNSC:stagger = "" ;
float CST_FSNS(Time) ;
        CST_FSNS:FieldType = 104 ;
        CST_FSNS:MemoryOrder = "0  " ;
        CST_FSNS:description = "Surface SW net upward total-sky radiation" ;
        CST_FSNS:units = "W/m^2" ;
        CST_FSNS:stagger = "" ;
float CST_FLNSC(Time) ;
        CST_FLNSC:FieldType = 104 ;
        CST_FLNSC:MemoryOrder = "0  " ;
        CST_FLNSC:description = "Surface LW net upward clear-sky radiation" ;
        CST_FLNSC:units = "W/m^2" ;
        CST_FLNSC:stagger = "" ;
float CST_FLNS(Time) ;
        CST_FLNS:FieldType = 104 ;
        CST_FLNS:MemoryOrder = "0  " ;
        CST_FLNS:description = "Surface LW net upward total-sky radiation" ;
        CST_FLNS:units = "W/m^2" ;
        CST_FLNS:stagger = "" ;
float CST_SWINC(Time) ;
        CST_SWINC:FieldType = 104 ;
        CST_SWINC:MemoryOrder = "0  " ;
        CST_SWINC:description = "TOA solar insolation" ;
        CST_SWINC:units = "W/m^2" ;
        CST_SWINC:stagger = "" ;
float CST_TSK(Time) ;
        CST_TSK:FieldType = 104 ;
        CST_TSK:MemoryOrder = "0  " ;
        CST_TSK:description = "Surface skin temperature" ;
        CST_TSK:units = "K" ;
        CST_TSK:stagger = "" ;
float CST_UST(Time) ;
        CST_UST:FieldType = 104 ;
        CST_UST:MemoryOrder = "0  " ;
        CST_UST:description = "Surface friction velocity" ;
        CST_UST:units = "m/s" ;
        CST_UST:stagger = "" ;
float CSP_Z(Time, bottom_top) ;
        CSP_Z:FieldType = 104 ;
        CSP_Z:MemoryOrder = "Z  " ;
        CSP_Z:description = "Half level height" ;
        CSP_Z:units = "m" ;
        CSP_Z:stagger = "" ;
float CSP_Z8W(Time, bottom_top_stag) ;
        CSP_Z8W:FieldType = 104 ;
        CSP_Z8W:MemoryOrder = "Z  " ;
        CSP_Z8W:description = "Full level height" ;
        CSP_Z8W:units = "m" ;
        CSP_Z8W:stagger = "Z" ;
float CSP_DZ8W(Time, bottom_top) ;
        CSP_DZ8W:FieldType = 104 ;
        CSP_DZ8W:MemoryOrder = "Z  " ;
```

```
        CSP_DZ8W:description = "dz at full level" ;
        CSP_DZ8W:units = "m" ;
        CSP_DZ8W:stagger = "" ;
float CSP_U(Time, bottom_top) ;
        CSP_U:FieldType = 104 ;
        CSP_U:MemoryOrder = "Z  " ;
        CSP_U:description = "Zonal wind" ;
        CSP_U:units = "m/s" ;
        CSP_U:stagger = "" ;
float CSP_V(Time, bottom_top) ;
        CSP_V:FieldType = 104 ;
        CSP_V:MemoryOrder = "Z  " ;
        CSP_V:description = "Meridional wind" ;
        CSP_V:units = "m/s" ;
        CSP_V:stagger = "" ;
float CSP_W(Time, bottom_top_stag) ;
        CSP_W:FieldType = 104 ;
        CSP_W:MemoryOrder = "Z  " ;
        CSP_W:description = "Vertical motion" ;
        CSP_W:units = "m/s" ;
        CSP_W:stagger = "Z" ;
float CSP_P(Time, bottom_top) ;
        CSP_P:FieldType = 104 ;
        CSP_P:MemoryOrder = "Z  " ;
        CSP_P:description = "Pressure" ;
        CSP_P:units = "Pa" ;
        CSP_P:stagger = "" ;
float CSP_TH(Time, bottom_top) ;
        CSP_TH:FieldType = 104 ;
        CSP_TH:MemoryOrder = "Z  " ;
        CSP_TH:description = "Potential temperature" ;
        CSP_TH:units = "K" ;
        CSP_TH:stagger = "" ;
float CSP_THV(Time, bottom_top) ;
        CSP_THV:FieldType = 104 ;
        CSP_THV:MemoryOrder = "Z  " ;
        CSP_THV:description = "Virtual potential temperature" ;
        CSP_THV:units = "K" ;
        CSP_THV:stagger = "" ;
float CSP_THL(Time, bottom_top) ;
        CSP_THL:FieldType = 104 ;
        CSP_THL:MemoryOrder = "Z  " ;
        CSP_THL:description = "Liquid water potential temperature" ;
        CSP_THL:units = "K" ;
        CSP_THL:stagger = "" ;
float CSP_QV(Time, bottom_top) ;
        CSP_QV:FieldType = 104 ;
        CSP_QV:MemoryOrder = "Z  " ;
        CSP_QV:description = "Water vapor mixing ratio" ;
        CSP_QV:units = "kg/kg" ;
        CSP_QV:stagger = "" ;
float CSP_QC(Time, bottom_top) ;
        CSP_QC:FieldType = 104 ;
        CSP_QC:MemoryOrder = "Z  " ;
        CSP_QC:description = "Cloud water mixing ratio" ;
        CSP_QC:units = "kg/kg" ;
        CSP_QC:stagger = "" ;
float CSP_QI(Time, bottom_top) ;
        CSP_QI:FieldType = 104 ;
        CSP_QI:MemoryOrder = "Z  " ;
```

```
        CSP_QI:description = "Ice crystal (cloud ice) mixing ratio" ;
        CSP_QI:units = "kg/kg" ;
        CSP_QI:stagger = "" ;
float CSP_QL(Time, bottom_top) ;
        CSP_QL:FieldType = 104 ;
        CSP_QL:MemoryOrder = "Z  " ;
        CSP_QL:description = "Liquid water mixing ratio" ;
        CSP_QL:units = "kg/kg" ;
        CSP_QL:stagger = "" ;
float CSP_QF(Time, bottom_top) ;
        CSP_QF:FieldType = 104 ;
        CSP_QF:MemoryOrder = "Z  " ;
        CSP_QF:description = "Frozen water mixing ratio" ;
        CSP_QF:units = "kg/kg" ;
        CSP_QF:stagger = "" ;
float CSP_QT(Time, bottom_top) ;
        CSP_QT:FieldType = 104 ;
        CSP_QT:MemoryOrder = "Z  " ;
        CSP_QT:description = "Total (vapor+liquid+frozen) water mixing ratio" ;
        CSP_QT:units = "kg/kg" ;
        CSP_QT:stagger = "" ;
float CSP_LWC(Time, bottom_top) ;
        CSP_LWC:FieldType = 104 ;
        CSP_LWC:MemoryOrder = "Z  " ;
        CSP_LWC:description = "Liquid water content (based on ql)" ;
        CSP_LWC:units = "kg/m^3" ;
        CSP_LWC:stagger = "" ;
float CSP_IWC(Time, bottom_top) ;
        CSP_IWC:FieldType = 104 ;
        CSP_IWC:MemoryOrder = "Z  " ;
        CSP_IWC:description = "Ice water content (based on qf)" ;
        CSP_IWC:units = "kg/m^3" ;
        CSP_IWC:stagger = "" ;
float CSP_SPEQV(Time, bottom_top) ;
        CSP_SPEQV:FieldType = 104 ;
        CSP_SPEQV:MemoryOrder = "Z  " ;
        CSP_SPEQV:description = "Specific humidity" ;
        CSP_SPEQV:units = "kg/kg" ;
        CSP_SPEQV:stagger = "" ;
float CSP_A_CL(Time, bottom_top) ;
        CSP_A_CL:FieldType = 104 ;
        CSP_A_CL:MemoryOrder = "Z  " ;
        CSP_A_CL:description = "Fraction of cloudy grid points" ;
        CSP_A_CL:units = "(0-1)" ;
        CSP_A_CL:stagger = "" ;
float CSP_RHO(Time, bottom_top) ;
        CSP_RHO:FieldType = 104 ;
        CSP_RHO:MemoryOrder = "Z  " ;
        CSP_RHO:description = "Density" ;
        CSP_RHO:units = "kg/m^3" ;
        CSP_RHO:stagger = "" ;
float CSP_U2(Time, bottom_top) ;
        CSP_U2:FieldType = 104 ;
        CSP_U2:MemoryOrder = "Z  " ;
        CSP_U2:description = "u_p^2" ;
        CSP_U2:units = "m^2/s^2" ;
        CSP_U2:stagger = "" ;
float CSP_V2(Time, bottom_top) ;
        CSP_V2:FieldType = 104 ;
        CSP_V2:MemoryOrder = "Z  " ;
```

```
        CSP_V2:description = “v_p^2” ;
        CSP_V2:units = “m^2/s^2” ;
        CSP_V2:stagger = “” ;
    float CSP_U2V2(Time, bottom_top) ;
        CSP_U2V2:FieldType = 104 ;
        CSP_U2V2:MemoryOrder = “Z  “ ;
        CSP_U2V2:description = “u_p^2+v_p^2” ;
        CSP_U2V2:units = “m^2/s^2” ;
        CSP_U2V2:stagger = “” ;
    float CSP_W2(Time, bottom_top_stag) ;
        CSP_W2:FieldType = 104 ;
        CSP_W2:MemoryOrder = “Z  “ ;
        CSP_W2:description = “w_p^2” ;
        CSP_W2:units = “m^2/s^2” ;
        CSP_W2:stagger = “Z” ;
    float CSP_W3(Time, bottom_top_stag) ;
        CSP_W3:FieldType = 104 ;
        CSP_W3:MemoryOrder = “Z  “ ;
        CSP_W3:description = “w_p^3” ;
        CSP_W3:units = “m^3/s^3” ;
        CSP_W3:stagger = “Z” ;
    float CSP_WSKEW(Time, bottom_top_stag) ;
        CSP_WSKEW:FieldType = 104 ;
        CSP_WSKEW:MemoryOrder = “Z  “ ;
        CSP_WSKEW:description = “Skewness <w3>/<w2>^(3/2)” ;
        CSP_WSKEW:units = “” ;
        CSP_WSKEW:stagger = “Z” ;
    float CSP_UW(Time, bottom_top) ;
        CSP_UW:FieldType = 104 ;
        CSP_UW:MemoryOrder = “Z  “ ;
        CSP_UW:description = “x-momentum flux uw (rs+sgs)” ;
        CSP_UW:units = “m^2/s^2” ;
        CSP_UW:stagger = “” ;
    float CSP_VW(Time, bottom_top) ;
        CSP_VW:FieldType = 104 ;
        CSP_VW:MemoryOrder = “Z  “ ;
        CSP_VW:description = “y-momentum flux vw (rs+sgs)” ;
        CSP_VW:units = “m^2/s^2” ;
        CSP_VW:stagger = “” ;
    float CSP_WTH(Time, bottom_top) ;
        CSP_WTH:FieldType = 104 ;
        CSP_WTH:MemoryOrder = “Z  “ ;
        CSP_WTH:description = “Potential temperature flux (rs+sgs)” ;
        CSP_WTH:units = “K m/s” ;
        CSP_WTH:stagger = “” ;
    float CSP_WTHV(Time, bottom_top) ;
        CSP_WTHV:FieldType = 104 ;
        CSP_WTHV:MemoryOrder = “Z  “ ;
        CSP_WTHV:description = “Virtual potential temperature flux (rs+sgs)” ;
        CSP_WTHV:units = “K m/s” ;
        CSP_WTHV:stagger = “” ;
    float CSP_WTHL(Time, bottom_top) ;
        CSP_WTHL:FieldType = 104 ;
        CSP_WTHL:MemoryOrder = “Z  “ ;
        CSP_WTHL:description = “Liquid water potential temperature flux (rs+sgs)” ;
        CSP_WTHL:units = “K m/s” ;
        CSP_WTHL:stagger = “” ;
    float CSP_WQV(Time, bottom_top) ;
        CSP_WQV:FieldType = 104 ;
        CSP_WQV:MemoryOrder = “Z  “ ;
```

```
        CSP_WQV:description = “Water vapor flux (rs+sgs)” ;
        CSP_WQV:units = “kg/kg m/s” ;
        CSP_WQV:stagger = “” ;
    float CSP_WQC(Time, bottom_top) ;
        CSP_WQC:FieldType = 104 ;
        CSP_WQC:MemoryOrder = “Z  “ ;
        CSP_WQC:description = “Cloud water flux (rs+sgs)” ;
        CSP_WQC:units = “kg/kg m/s” ;
        CSP_WQC:stagger = “” ;
    float CSP_WQI(Time, bottom_top) ;
        CSP_WQI:FieldType = 104 ;
        CSP_WQI:MemoryOrder = “Z  “ ;
        CSP_WQI:description = “Ice crystal (cloud ice) flux (rs+sgs)” ;
        CSP_WQI:units = “kg/kg m/s” ;
        CSP_WQI:stagger = “” ;
    float CSP_WQL(Time, bottom_top) ;
        CSP_WQL:FieldType = 104 ;
        CSP_WQL:MemoryOrder = “Z  “ ;
        CSP_WQL:description = “Liquid water flux (rs+sgs)” ;
        CSP_WQL:units = “kg/kg m/s” ;
        CSP_WQL:stagger = “” ;
    float CSP_WQF(Time, bottom_top) ;
        CSP_WQF:FieldType = 104 ;
        CSP_WQF:MemoryOrder = “Z  “ ;
        CSP_WQF:description = “Frozen water flux (rs+sgs)” ;
        CSP_WQF:units = “kg/kg m/s” ;
        CSP_WQF:stagger = “” ;
    float CSP_WQT(Time, bottom_top) ;
        CSP_WQT:FieldType = 104 ;
        CSP_WQT:MemoryOrder = “Z  “ ;
        CSP_WQT:description = “Total water flux (rs+sgs)” ;
        CSP_WQT:units = “kg/kg m/s” ;
        CSP_WQT:stagger = “” ;
    float CSP_UW_SGS(Time, bottom_top) ;
        CSP_UW_SGS:FieldType = 104 ;
        CSP_UW_SGS:MemoryOrder = “Z  “ ;
        CSP_UW_SGS:description = “x-momentum flux uw (sgs)” ;
        CSP_UW_SGS:units = “m^2/s^2” ;
        CSP_UW_SGS:stagger = “” ;
    float CSP_VW_SGS(Time, bottom_top) ;
        CSP_VW_SGS:FieldType = 104 ;
        CSP_VW_SGS:MemoryOrder = “Z  “ ;
        CSP_VW_SGS:description = “y-momentum flux vw (sgs)” ;
        CSP_VW_SGS:units = “m^2/s^2” ;
        CSP_VW_SGS:stagger = “” ;
    float CSP_WTH_SGS(Time, bottom_top) ;
        CSP_WTH_SGS:FieldType = 104 ;
        CSP_WTH_SGS:MemoryOrder = “Z  “ ;
        CSP_WTH_SGS:description = “Potential temperature flux (sgs)” ;
        CSP_WTH_SGS:units = “K m/s” ;
        CSP_WTH_SGS:stagger = “” ;
    float CSP_WTHV_SGS(Time, bottom_top) ;
        CSP_WTHV_SGS:FieldType = 104 ;
        CSP_WTHV_SGS:MemoryOrder = “Z  “ ;
        CSP_WTHV_SGS:description = “Virtual potential temperature flux (sgs)” ;
        CSP_WTHV_SGS:units = “K m/s” ;
        CSP_WTHV_SGS:stagger = “” ;
    float CSP_WTHL_SGS(Time, bottom_top) ;
        CSP_WTHL_SGS:FieldType = 104 ;
        CSP_WTHL_SGS:MemoryOrder = “Z  “ ;
```

```
        CSP_WTHL_SGS:description = “Liquid water potential temperature flux (sgs)” ;
        CSP_WTHL_SGS:units = “K m/s” ;
        CSP_WTHL_SGS:stagger = “” ;
float CSP_WQV_SGS(Time, bottom_top) ;
        CSP_WQV_SGS:FieldType = 104 ;
        CSP_WQV_SGS:MemoryOrder = “Z  “ ;
        CSP_WQV_SGS:description = “Water vapor flux (sgs)” ;
        CSP_WQV_SGS:units = “kg/kg m/s” ;
        CSP_WQV_SGS:stagger = “” ;
float CSP_WQC_SGS(Time, bottom_top) ;
        CSP_WQC_SGS:FieldType = 104 ;
        CSP_WQC_SGS:MemoryOrder = “Z  “ ;
        CSP_WQC_SGS:description = “Cloud water flux (sgs)” ;
        CSP_WQC_SGS:units = “kg/kg m/s” ;
        CSP_WQC_SGS:stagger = “” ;
float CSP_WQI_SGS(Time, bottom_top) ;
        CSP_WQI_SGS:FieldType = 104 ;
        CSP_WQI_SGS:MemoryOrder = “Z  “ ;
        CSP_WQI_SGS:description = “Ice crystal (cloud ice) flux (sgs)” ;
        CSP_WQI_SGS:units = “kg/kg m/s” ;
        CSP_WQI_SGS:stagger = “” ;
float CSP_WQL_SGS(Time, bottom_top) ;
        CSP_WQL_SGS:FieldType = 104 ;
        CSP_WQL_SGS:MemoryOrder = “Z  “ ;
        CSP_WQL_SGS:description = “Liquid water flux (sgs)” ;
        CSP_WQL_SGS:units = “kg/kg m/s” ;
        CSP_WQL_SGS:stagger = “” ;
float CSP_WQF_SGS(Time, bottom_top) ;
        CSP_WQF_SGS:FieldType = 104 ;
        CSP_WQF_SGS:MemoryOrder = “Z  “ ;
        CSP_WQF_SGS:description = “Frozen water flux (sgs)” ;
        CSP_WQF_SGS:units = “kg/kg m/s” ;
        CSP_WQF_SGS:stagger = “” ;
float CSP_WQT_SGS(Time, bottom_top) ;
        CSP_WQT_SGS:FieldType = 104 ;
        CSP_WQT_SGS:MemoryOrder = “Z  “ ;
        CSP_WQT_SGS:description = “Total water flux (sgs)” ;
        CSP_WQT_SGS:units = “kg/kg m/s” ;
        CSP_WQT_SGS:stagger = “” ;
float CSP_SEDFQC(Time, bottom_top) ;
        CSP_SEDFQC:FieldType = 104 ;
        CSP_SEDFQC:MemoryOrder = “Z  “ ;
        CSP_SEDFQC:description = “Sedimentation flux of qc” ;
        CSP_SEDFQC:units = “kg /m^2/s” ;
        CSP_SEDFQC:stagger = “” ;
float CSP_SEDFQR(Time, bottom_top) ;
        CSP_SEDFQR:FieldType = 104 ;
        CSP_SEDFQR:MemoryOrder = “Z  “ ;
        CSP_SEDFQR:description = “Sedimentation (Precipitation) flux of qr” ;
        CSP_SEDFQR:units = “kg /m^2/s” ;
        CSP_SEDFQR:stagger = “” ;
float CSP_THDT_COND(Time, bottom_top) ;
        CSP_THDT_COND:FieldType = 104 ;
        CSP_THDT_COND:MemoryOrder = “Z  “ ;
        CSP_THDT_COND:description = “dth/dt due to net condensation” ;
        CSP_THDT_COND:units = “K/s” ;
        CSP_THDT_COND:stagger = “” ;
float CSP_THDT_LW(Time, bottom_top) ;
        CSP_THDT_LW:FieldType = 104 ;
        CSP_THDT_LW:MemoryOrder = “Z  “ ;
```

```
        CSP_THDT_LW:description = “dth/dt due to LW radiation” ;
        CSP_THDT_LW:units = “K/s” ;
        CSP_THDT_LW:stagger = “” ;
float CSP_THDT_SW(Time, bottom_top) ;
        CSP_THDT_SW:FieldType = 104 ;
        CSP_THDT_SW:MemoryOrder = “Z  “ ;
        CSP_THDT_SW:description = “dth/dt due to SW radiation” ;
        CSP_THDT_SW:units = “K/s” ;
        CSP_THDT_SW:stagger = “” ;
float CSP_THDT_LS(Time, bottom_top) ;
        CSP_THDT_LS:FieldType = 104 ;
        CSP_THDT_LS:MemoryOrder = “Z  “ ;
        CSP_THDT_LS:description = “dth/dt due to large-scale forcing” ;
        CSP_THDT_LS:units = “K/s” ;
        CSP_THDT_LS:stagger = “” ;
float CSP_QVDT_PR(Time, bottom_top) ;
        CSP_QVDT_PR:FieldType = 104 ;
        CSP_QVDT_PR:MemoryOrder = “Z  “ ;
        CSP_QVDT_PR:description = “dqv/dt due to conversion to precipitation” ;
        CSP_QVDT_PR:units = “kg/kg/s” ;
        CSP_QVDT_PR:stagger = “” ;
float CSP_QVDT_COND(Time, bottom_top) ;
        CSP_QVDT_COND:FieldType = 104 ;
        CSP_QVDT_COND:MemoryOrder = “Z  “ ;
        CSP_QVDT_COND:description = “dqv/dt due to net condensation” ;
        CSP_QVDT_COND:units = “kg/kg/s” ;
        CSP_QVDT_COND:stagger = “” ;
float CSP_QVDT_LS(Time, bottom_top) ;
        CSP_QVDT_LS:FieldType = 104 ;
        CSP_QVDT_LS:MemoryOrder = “Z  “ ;
        CSP_QVDT_LS:description = “dqv/dt due to large-scale forcing” ;
        CSP_QVDT_LS:units = “kg/kg/s” ;
        CSP_QVDT_LS:stagger = “” ;
float CSP_QCDT_PR(Time, bottom_top) ;
        CSP_QCDT_PR:FieldType = 104 ;
        CSP_QCDT_PR:MemoryOrder = “Z  “ ;
        CSP_QCDT_PR:description = “dqc/dt due to conversion to precipitation” ;
        CSP_QCDT_PR:units = “kg/kg/s” ;
        CSP_QCDT_PR:stagger = “” ;
float CSP_QCDT_SED(Time, bottom_top) ;
        CSP_QCDT_SED:FieldType = 104 ;
        CSP_QCDT_SED:MemoryOrder = “Z  “ ;
        CSP_QCDT_SED:description = “dqc/dt due to sedimentation” ;
        CSP_QCDT_SED:units = “kg/kg/s” ;
        CSP_QCDT_SED:stagger = “” ;
float CSP_QRDT_SED(Time, bottom_top) ;
        CSP_QRDT_SED:FieldType = 104 ;
        CSP_QRDT_SED:MemoryOrder = “Z  “ ;
        CSP_QRDT_SED:description = “dqr/dt due to sedimentation” ;
        CSP_QRDT_SED:units = “kg/kg/s” ;
        CSP_QRDT_SED:stagger = “” ;
float CSP_THDT_LSHOR(Time, bottom_top) ;
        CSP_THDT_LSHOR:FieldType = 104 ;
        CSP_THDT_LSHOR:MemoryOrder = “Z  “ ;
        CSP_THDT_LSHOR:description = “th tendency due to LS horizontal adv” ;
        CSP_THDT_LSHOR:units = “K s-1” ;
        CSP_THDT_LSHOR:stagger = “” ;
float CSP_QVDT_LSHOR(Time, bottom_top) ;
        CSP_QVDT_LSHOR:FieldType = 104 ;
        CSP_QVDT_LSHOR:MemoryOrder = “Z  “ ;
```

```
        CSP_QVDT_LSHOR:description = “qv tendency due to LS horizontal adv” ;
        CSP_QVDT_LSHOR:units = “kg kg-1 s-1” ;
        CSP_QVDT_LSHOR:stagger = “” ;
float CSP_THDT_LSVER(Time, bottom_top) ;
        CSP_THDT_LSVER:FieldType = 104 ;
        CSP_THDT_LSVER:MemoryOrder = “Z  “ ;
        CSP_THDT_LSVER:description = “th tendency due to LS horizontal adv” ;
        CSP_THDT_LSVER:units = “K s-1” ;
        CSP_THDT_LSVER:stagger = “” ;
float CSP_QVDT_LSVER(Time, bottom_top) ;
        CSP_QVDT_LSVER:FieldType = 104 ;
        CSP_QVDT_LSVER:MemoryOrder = “Z  “ ;
        CSP_QVDT_LSVER:description = “qv tendency due to LS horizontal adv” ;
        CSP_QVDT_LSVER:units = “kg kg-1 s-1” ;
        CSP_QVDT_LSVER:stagger = “” ;
float CSP_THDT_LSRLX(Time, bottom_top) ;
        CSP_THDT_LSRLX:FieldType = 104 ;
        CSP_THDT_LSRLX:MemoryOrder = “Z  “ ;
        CSP_THDT_LSRLX:description = “th tendency due to relaxation to LS” ;
        CSP_THDT_LSRLX:units = “K s-1” ;
        CSP_THDT_LSRLX:stagger = “” ;
float CSP_QVDT_LSRLX(Time, bottom_top) ;
        CSP_QVDT_LSRLX:FieldType = 104 ;
        CSP_QVDT_LSRLX:MemoryOrder = “Z  “ ;
        CSP_QVDT_LSRLX:description = “qv tendency due to relaxation to LS” ;
        CSP_QVDT_LSRLX:units = “kg kg-1 s-1” ;
        CSP_QVDT_LSRLX:stagger = “” ;
float CSP_UDT_LS(Time, bottom_top) ;
        CSP_UDT_LS:FieldType = 104 ;
        CSP_UDT_LS:MemoryOrder = “Z  “ ;
        CSP_UDT_LS:description = “u tendency due to LS forcing” ;
        CSP_UDT_LS:units = “m s-2” ;
        CSP_UDT_LS:stagger = “” ;
float CSP_VDT_LS(Time, bottom_top) ;
        CSP_VDT_LS:FieldType = 104 ;
        CSP_VDT_LS:MemoryOrder = “Z  “ ;
        CSP_VDT_LS:description = “v tendency due to LS forcing” ;
        CSP_VDT_LS:units = “m s-2” ;
        CSP_VDT_LS:stagger = “” ;
float CSP_UDT_LSVER(Time, bottom_top) ;
        CSP_UDT_LSVER:FieldType = 104 ;
        CSP_UDT_LSVER:MemoryOrder = “Z  “ ;
        CSP_UDT_LSVER:description = “u tendency due to LS vertical adv” ;
        CSP_UDT_LSVER:units = “m s-2” ;
        CSP_UDT_LSVER:stagger = “” ;
float CSP_VDT_LSVER(Time, bottom_top) ;
        CSP_VDT_LSVER:FieldType = 104 ;
        CSP_VDT_LSVER:MemoryOrder = “Z  “ ;
        CSP_VDT_LSVER:description = “v tendency due to LS vertical adv” ;
        CSP_VDT_LSVER:units = “m s-2” ;
        CSP_VDT_LSVER:stagger = “” ;
float CSP_UDT_LSRLX(Time, bottom_top) ;
        CSP_UDT_LSRLX:FieldType = 104 ;
        CSP_UDT_LSRLX:MemoryOrder = “Z  “ ;
        CSP_UDT_LSRLX:description = “u tendency due to relaxation to LS” ;
        CSP_UDT_LSRLX:units = “m s-2” ;         CSP_UDT_LSRLX:stagger = “” ;
float CSP_VDT_LSRLX(Time, bottom_top) ;
        CSP_VDT_LSRLX:FieldType = 104 ;
        CSP_VDT_LSRLX:MemoryOrder = “Z  “ ;
        CSP_VDT_LSRLX:description = “v tendency due to relaxation to LS” ;
```

```
        CSP_VDT_LSRLX:units = “m s-2” ;
        CSP_VDT_LSRLX:stagger = “” ;
float CSP_SWUPF(Time, bottom_top) ;
        CSP_SWUPF:FieldType = 104 ;
        CSP_SWUPF:MemoryOrder = “Z  “ ;
        CSP_SWUPF:description = “SW flux upward” ;
        CSP_SWUPF:units = “W/m^2” ;
        CSP_SWUPF:stagger = “” ;
float CSP_SWDNF(Time, bottom_top) ;
        CSP_SWDNF:FieldType = 104 ;
        CSP_SWDNF:MemoryOrder = “Z  “ ;
        CSP_SWDNF:description = “SW flux downward” ;
        CSP_SWDNF:units = “W/m^2” ;
        CSP_SWDNF:stagger = “” ;
float CSP_LWUPF(Time, bottom_top) ;
        CSP_LWUPF:FieldType = 104 ;
        CSP_LWUPF:MemoryOrder = “Z  “ ;
        CSP_LWUPF:description = “LW flux upward” ;
        CSP_LWUPF:units = “W/m^2” ;
        CSP_LWUPF:stagger = “” ;
float CSP_LWDNF(Time, bottom_top) ;
        CSP_LWDNF:FieldType = 104 ;
        CSP_LWDNF:MemoryOrder = “Z  “ ;
        CSP_LWDNF:description = “LW flux downward” ;
        CSP_LWDNF:units = “W/m^2” ;
        CSP_LWDNF:stagger = “” ;
float CSP_TKE_RS(Time, bottom_top) ;
        CSP_TKE_RS:FieldType = 104 ;
        CSP_TKE_RS:MemoryOrder = “Z  “ ;
        CSP_TKE_RS:description = “RS TKE” ;
        CSP_TKE_RS:units = “m^2/s^2” ;
        CSP_TKE_RS:stagger = “” ;
float CSP_TKE_SH(Time, bottom_top) ;
        CSP_TKE_SH:FieldType = 104 ;
        CSP_TKE_SH:MemoryOrder = “Z  “ ;
        CSP_TKE_SH:description = “RS TKE shear production” ;
        CSP_TKE_SH:units = “m^2/s^3” ;
        CSP_TKE_SH:stagger = “” ;
float CSP_TKE_BU(Time, bottom_top) ;
        CSP_TKE_BU:FieldType = 104 ;
        CSP_TKE_BU:MemoryOrder = “Z  “ ;
        CSP_TKE_BU:description = “RS TKE buoyancy production” ;
        CSP_TKE_BU:units = “m^2/s^3” ;
        CSP_TKE_BU:stagger = “” ;
float CSP_TKE_TR(Time, bottom_top) ;
        CSP_TKE_TR:FieldType = 104 ;
        CSP_TKE_TR:MemoryOrder = “Z  “ ;
        CSP_TKE_TR:description = “RS TKE turbulent + pressure transport” ;
        CSP_TKE_TR:units = “m^2/s^3” ;
        CSP_TKE_TR:stagger = “” ;
float CSP_TKE_DI(Time, bottom_top) ;
        CSP_TKE_DI:FieldType = 104 ;
        CSP_TKE_DI:MemoryOrder = “Z  “ ;
        CSP_TKE_DI:description = “TKE dissipation” ;
        CSP_TKE_DI:units = “m^2/s^3” ;
        CSP_TKE_DI:stagger = “” ;
float CSP_TKE_SGS(Time, bottom_top) ;
        CSP_TKE_SGS:FieldType = 104 ;
        CSP_TKE_SGS:MemoryOrder = “Z  “ ;
```

```
        CSP_TKE_SGS:description = "SGS TKE" ;
        CSP_TKE_SGS:units = "m^2/s^2" ;
        CSP_TKE_SGS:stagger = "" ;
float CSP_W_C(Time, bottom_top) ;
        CSP_W_C:FieldType = 104 ;
        CSP_W_C:MemoryOrder = "Z  " ;
        CSP_W_C:description = "Average over all cloudy grid points of w" ;
        CSP_W_C:units = "m/s" ;
        CSP_W_C:stagger = "" ;
float CSP_THL_C(Time, bottom_top) ;
        CSP_THL_C:FieldType = 104 ;
        CSP_THL_C:MemoryOrder = "Z  " ;
        CSP_THL_C:description = "Average over all cloudy grid points of thl" ;
        CSP_THL_C:units = "K" ;
        CSP_THL_C:stagger = "" ;
float CSP_QT_C(Time, bottom_top) ;
        CSP_QT_C:FieldType = 104 ;
        CSP_QT_C:MemoryOrder = "Z  " ;
        CSP_QT_C:description = "Average over all cloudy grid points of qt" ;
        CSP_QT_C:units = "kg/kg" ;
        CSP_QT_C:stagger = "" ;
float CSP_QV_C(Time, bottom_top) ;
        CSP_QV_C:FieldType = 104 ;
        CSP_QV_C:MemoryOrder = "Z  " ;
        CSP_QV_C:description = "Average over all cloudy grid points of qv" ;
        CSP_QV_C:units = "kg/kg" ;
        CSP_QV_C:stagger = "" ;
float CSP_QL_C(Time, bottom_top) ;
        CSP_QL_C:FieldType = 104 ;
        CSP_QL_C:MemoryOrder = "Z  " ;
        CSP_QL_C:description = "Average over all cloudy grid points of ql" ;
        CSP_QL_C:units = "kg/kg" ;
        CSP_QL_C:stagger = "" ;
float CSP_QF_C(Time, bottom_top) ;
        CSP_QF_C:FieldType = 104 ;
        CSP_QF_C:MemoryOrder = "Z  " ;
        CSP_QF_C:description = "Average over all cloudy grid points of qf" ;
        CSP_QF_C:units = "kg/kg" ;
        CSP_QF_C:stagger = "" ;
float CSP_QC_C(Time, bottom_top) ;
        CSP_QC_C:FieldType = 104 ;
        CSP_QC_C:MemoryOrder = "Z  " ;
        CSP_QC_C:description = "Average over all cloudy grid points of qc" ;
        CSP_QC_C:units = "kg/kg" ;
        CSP_QC_C:stagger = "" ;
float CSP_QI_C(Time, bottom_top) ;
        CSP_QI_C:FieldType = 104 ;
        CSP_QI_C:MemoryOrder = "Z  " ;
        CSP_QI_C:description = "Average over all cloudy grid points of qi" ;
        CSP_QI_C:units = "kg/kg" ;
        CSP_QI_C:stagger = "" ;
float CSP_QNC_C(Time, bottom_top) ;
        CSP_QNC_C:FieldType = 104 ;
        CSP_QNC_C:MemoryOrder = "Z  " ;
        CSP_QNC_C:description = "Average over all cloudy grid points of qnc" ;
        CSP_QNC_C:units = "cm-3" ;
        CSP_QNC_C:stagger = "" ;
float CSP_THV_C(Time, bottom_top) ;
        CSP_THV_C:FieldType = 104 ;
        CSP_THV_C:MemoryOrder = "Z  " ;
```

```
            CSP_THV_C:description = "Average over all cloudy grid points of thv" ;
            CSP_THV_C:units = "K" ;
            CSP_THV_C:stagger = "" ;
      float CSP_W2_C(Time, bottom_top) ;
            CSP_W2_C:FieldType = 104 ;
            CSP_W2_C:MemoryOrder = "Z  " ;
            CSP_W2_C:description = "Average over all cloudy grid points of w vari-
ance" ;
            CSP_W2_C:units = "(m/s)^2" ;
            CSP_W2_C:stagger = "" ;
      float CSP_AW_C(Time, bottom_top) ;
            CSP_AW_C:FieldType = 104 ;
            CSP_AW_C:MemoryOrder = "Z  " ;
            CSP_AW_C:description = "Cloud fraction * average over all cloudy grid
points of w" ;
            CSP_AW_C:units = "m/s" ;
            CSP_AW_C:stagger = "" ;
      float CSP_AWTHL_C(Time, bottom_top) ;
            CSP_AWTHL_C:FieldType = 104 ;
            CSP_AWTHL_C:MemoryOrder = "Z  " ;
            CSP_AWTHL_C:description = "Cloud fraction * average over all cloudy grid
points of wthl" ;
            CSP_AWTHL_C:units = "K m/s" ;
            CSP_AWTHL_C:stagger = "" ;
      float CSP_AWQT_C(Time, bottom_top) ;
            CSP_AWQT_C:FieldType = 104 ;
            CSP_AWQT_C:MemoryOrder = "Z  " ;
            CSP_AWQT_C:description = "Cloud fraction * average over all cloudy grid
points of wqt" ;
            CSP_AWQT_C:units = "kg/kg m/s" ;
            CSP_AWQT_C:stagger = "" ;
      float CSP_AWQV_C(Time, bottom_top) ;
            CSP_AWQV_C:FieldType = 104 ;
            CSP_AWQV_C:MemoryOrder = "Z  " ;
            CSP_AWQV_C:description = "Cloud fraction * average over all cloudy grid
points of wqv" ;
            CSP_AWQV_C:units = "kg/kg m/s" ;
            CSP_AWQV_C:stagger = "" ;
      float CSP_AWQL_C(Time, bottom_top) ;
            CSP_AWQL_C:FieldType = 104 ;
            CSP_AWQL_C:MemoryOrder = "Z  " ;
            CSP_AWQL_C:description = "Cloud fraction * average over all cloudy grid
points of wql" ;
            CSP_AWQL_C:units = "kg/kg m/s" ;
            CSP_AWQL_C:stagger = "" ;
      float CSP_AWQF_C(Time, bottom_top) ;
            CSP_AWQF_C:FieldType = 104 ;
            CSP_AWQF_C:MemoryOrder = "Z  " ;
            CSP_AWQF_C:description = "Cloud fraction * average over all cloudy grid
points of wqf" ;
            CSP_AWQF_C:units = "kg/kg m/s" ;
            CSP_AWQF_C:stagger = "" ;
      float CSP_AWQC_C(Time, bottom_top) ;
            CSP_AWQC_C:FieldType = 104 ;
            CSP_AWQC_C:MemoryOrder = "Z  " ;
            CSP_AWQC_C:description = "Cloud fraction * average over all cloudy grid
points of wqc" ;
            CSP_AWQC_C:units = "kg/kg m/s" ;
            CSP_AWQC_C:stagger = "" ;
```

```
	float CSP_AWQI_C(Time, bottom_top) ;
		CSP_AWQI_C:FieldType = 104 ;
		CSP_AWQI_C:MemoryOrder = "Z  " ;
		CSP_AWQI_C:description = "Cloud fraction * average over all cloudy grid 
points of wqi" ;
		CSP_AWQI_C:units = "kg/kg m/s" ;
		CSP_AWQI_C:stagger = "" ;
	float CSP_AWTHV_C(Time, bottom_top) ;
		CSP_AWTHV_C:FieldType = 104 ;
		CSP_AWTHV_C:MemoryOrder = "Z  " ;
		CSP_AWTHV_C:description = "Cloud fraction * average over all cloudy grid 
points of wthv" ;
		CSP_AWTHV_C:units = "K m/s" ;
		CSP_AWTHV_C:stagger = "" ;
	float CSP_A_CC(Time, bottom_top) ;
		CSP_A_CC:FieldType = 104 ;
		CSP_A_CC:MemoryOrder = "Z  " ;
		CSP_A_CC:description = "Fraction of cloudcore grid points" ;
		CSP_A_CC:units = "(0-1)" ;
		CSP_A_CC:stagger = "" ;
	float CSP_W_CC(Time, bottom_top) ;
		CSP_W_CC:FieldType = 104 ;
		CSP_W_CC:MemoryOrder = "Z  " ;
		CSP_W_CC:description = "Average over all cloudcore grid points of w" ;
		CSP_W_CC:units = "m/s" ;
		CSP_W_CC:stagger = "" ;
	float CSP_THL_CC(Time, bottom_top) ;
		CSP_THL_CC:FieldType = 104 ;
		CSP_THL_CC:MemoryOrder = "Z  " ;
		CSP_THL_CC:description = "Average over all cloudcore grid points of thl" ;
		CSP_THL_CC:units = "K" ;
		CSP_THL_CC:stagger = "" ;
	float CSP_QT_CC(Time, bottom_top) ;
		CSP_QT_CC:FieldType = 104 ;
		CSP_QT_CC:MemoryOrder = "Z  " ;
		CSP_QT_CC:description = "Average over all cloudcore grid points of qt" ;
		CSP_QT_CC:units = "kg/kg" ;
		CSP_QT_CC:stagger = "" ;
	float CSP_QV_CC(Time, bottom_top) ;
		CSP_QV_CC:FieldType = 104 ;
		CSP_QV_CC:MemoryOrder = "Z  " ;
		CSP_QV_CC:description = "Average over all cloudcore grid points of qv" ;
		CSP_QV_CC:units = "kg/kg" ;
		CSP_QV_CC:stagger = "" ;
	float CSP_QL_CC(Time, bottom_top) ;
		CSP_QL_CC:FieldType = 104 ;
		CSP_QL_CC:MemoryOrder = "Z  " ;
		CSP_QL_CC:description = "Average over all cloudcore grid points of ql" ;
		CSP_QL_CC:units = "kg/kg" ;
		CSP_QL_CC:stagger = "" ;
	float CSP_QF_CC(Time, bottom_top) ;
		CSP_QF_CC:FieldType = 104 ;
		CSP_QF_CC:MemoryOrder = "Z  " ;
		CSP_QF_CC:description = "Average over all cloudcore grid points of qf" ;
		CSP_QF_CC:units = "kg/kg" ;
		CSP_QF_CC:stagger = "" ;
	float CSP_QC_CC(Time, bottom_top) ;
		CSP_QC_CC:FieldType = 104 ;
		CSP_QC_CC:MemoryOrder = "Z  " ;
```

```
            CSP_QC_CC:description = "Average over all cloudcore grid points of qc" ;
            CSP_QC_CC:units = "kg/kg" ;
            CSP_QC_CC:stagger = "" ;
      float CSP_QI_CC(Time, bottom_top) ;
            CSP_QI_CC:FieldType = 104 ;
            CSP_QI_CC:MemoryOrder = "Z  " ;
            CSP_QI_CC:description = "Average over all cloudcore grid points of qi" ;
            CSP_QI_CC:units = "kg/kg" ;
            CSP_QI_CC:stagger = "" ;
      float CSP_THV_CC(Time, bottom_top) ;
            CSP_THV_CC:FieldType = 104 ;
            CSP_THV_CC:MemoryOrder = "Z  " ;
            CSP_THV_CC:description = "Average over all cloudcore grid points of thv" ;
            CSP_THV_CC:units = "K" ;
            CSP_THV_CC:stagger = "" ;
      float CSP_W2_CC(Time, bottom_top) ;
            CSP_W2_CC:FieldType = 104 ;
            CSP_W2_CC:MemoryOrder = "Z  " ;
            CSP_W2_CC:description = "Average over all cloudcore grid points of w
variance" ;
            CSP_W2_CC:units = "(m/s)^2" ;
            CSP_W2_CC:stagger = "" ;
      float CSP_AW_CC(Time, bottom_top) ;
            CSP_AW_CC:FieldType = 104 ;
            CSP_AW_CC:MemoryOrder = "Z  " ;
            CSP_AW_CC:description = "Cloudcore fraction * average over all cloudcore
grid points of w" ;
            CSP_AW_CC:units = "m/s" ;
            CSP_AW_CC:stagger = "" ;
      float CSP_AWTHL_CC(Time, bottom_top) ;
            CSP_AWTHL_CC:FieldType = 104 ;
            CSP_AWTHL_CC:MemoryOrder = "Z  " ;
            CSP_AWTHL_CC:description = "Cloudcore fraction * average over all cloud-
core grid points of wthl" ;
            CSP_AWTHL_CC:units = "K m/s" ;
            CSP_AWTHL_CC:stagger = "" ;
      float CSP_AWQT_CC(Time, bottom_top) ;
            CSP_AWQT_CC:FieldType = 104 ;
            CSP_AWQT_CC:MemoryOrder = "Z  " ;
            CSP_AWQT_CC:description = "Cloudcore fraction * average over all cloud-
core grid points of wqt" ;
            CSP_AWQT_CC:units = "kg/kg m/s" ;
            CSP_AWQT_CC:stagger = "" ;
      float CSP_AWQV_CC(Time, bottom_top) ;
            CSP_AWQV_CC:FieldType = 104 ;
            CSP_AWQV_CC:MemoryOrder = "Z  " ;
            CSP_AWQV_CC:description = "Cloudcore fraction * average over all cloud-
core grid points of wqv" ;
            CSP_AWQV_CC:units = "kg/kg m/s" ;
            CSP_AWQV_CC:stagger = "" ;
      float CSP_AWQL_CC(Time, bottom_top) ;
            CSP_AWQL_CC:FieldType = 104 ;
            CSP_AWQL_CC:MemoryOrder = "Z  " ;
            CSP_AWQL_CC:description = "Cloudcore fraction * average over all cloud-
core grid points of wql" ;
            CSP_AWQL_CC:units = "kg/kg m/s" ;
            CSP_AWQL_CC:stagger = "" ;
      float CSP_AWQF_CC(Time, bottom_top) ;
            CSP_AWQF_CC:FieldType = 104 ;
```

```
             CSP_AWQF_CC:MemoryOrder = "Z  " ;
             CSP_AWQF_CC:description = "Cloudcore fraction * average over all cloud-
core grid points of wqf" ;
             CSP_AWQF_CC:units = "kg/kg m/s" ;
             CSP_AWQF_CC:stagger = "" ;
      float CSP_AWQC_CC(Time, bottom_top) ;
             CSP_AWQC_CC:FieldType = 104 ;
             CSP_AWQC_CC:MemoryOrder = "Z  " ;
             CSP_AWQC_CC:description = "Cloudcore fraction * average over all cloud-
core grid points of wqc" ;
             CSP_AWQC_CC:units = "kg/kg m/s" ;
             CSP_AWQC_CC:stagger = "" ;
      float CSP_AWQI_CC(Time, bottom_top) ;
             CSP_AWQI_CC:FieldType = 104 ;
             CSP_AWQI_CC:MemoryOrder = "Z  " ;
             CSP_AWQI_CC:description = "Cloudcore fraction * average over all cloud-
core grid points of wqi" ;
             CSP_AWQI_CC:units = "kg/kg m/s" ;
             CSP_AWQI_CC:stagger = "" ;
      float CSP_AWTHV_CC(Time, bottom_top) ;
             CSP_AWTHV_CC:FieldType = 104 ;
             CSP_AWTHV_CC:MemoryOrder = "Z  " ;
             CSP_AWTHV_CC:description = "Cloudcore fraction * average over all cloud-
core grid points of wthv" ;
             CSP_AWTHV_CC:units = "K m/s" ;
             CSP_AWTHV_CC:stagger = "" ;
      float CSP_SIGC_THL(Time, bottom_top) ;
             CSP_SIGC_THL:FieldType = 104 ;
             CSP_SIGC_THL:MemoryOrder = "Z  " ;
             CSP_SIGC_THL:description = "Incloud variance of thl" ;
             CSP_SIGC_THL:units = "K^2" ;
             CSP_SIGC_THL:stagger = "" ;
      float CSP_SIGC_QT(Time, bottom_top) ;
             CSP_SIGC_QT:FieldType = 104 ;
             CSP_SIGC_QT:MemoryOrder = "Z  " ;
             CSP_SIGC_QT:description = "Incloud variance of qt" ;
             CSP_SIGC_QT:units = "(kg/kg)^2" ;
             CSP_SIGC_QT:stagger = "" ;
      float CSP_SIGC_QL(Time, bottom_top) ;
             CSP_SIGC_QL:FieldType = 104 ;
             CSP_SIGC_QL:MemoryOrder = "Z  " ;
             CSP_SIGC_QL:description = "Incloud variance of ql" ;
             CSP_SIGC_QL:units = "(kg/kg)^2" ;
             CSP_SIGC_QL:stagger = "" ;
      float CSP_SIGC_QF(Time, bottom_top) ;
             CSP_SIGC_QF:FieldType = 104 ;
             CSP_SIGC_QF:MemoryOrder = "Z  " ;
             CSP_SIGC_QF:description = "Incloud variance of qf" ;
             CSP_SIGC_QF:units = "(kg/kg)^2" ;
             CSP_SIGC_QF:stagger = "" ;
      float CSP_SIGC_QC(Time, bottom_top) ;
             CSP_SIGC_QC:FieldType = 104 ;
             CSP_SIGC_QC:MemoryOrder = "Z  " ;
             CSP_SIGC_QC:description = "Incloud variance of qc" ;
             CSP_SIGC_QC:units = "(kg/kg)^2" ;
             CSP_SIGC_QC:stagger = "" ;
      float CSP_SIGC_QI(Time, bottom_top) ;
             CSP_SIGC_QI:FieldType = 104 ;
             CSP_SIGC_QI:MemoryOrder = "Z  " ;
```

```
            CSP_SIGC_QI:description = "Incloud variance of qi" ;
            CSP_SIGC_QI:units = "(kg/kg)^2" ;
            CSP_SIGC_QI:stagger = "" ;
      float CSP_SIGC_THV(Time, bottom_top) ;
            CSP_SIGC_THV:FieldType = 104 ;
            CSP_SIGC_THV:MemoryOrder = "Z  " ;
            CSP_SIGC_THV:description = "Incloud variance of thv" ;
            CSP_SIGC_THV:units = "K^2" ;
            CSP_SIGC_THV:stagger = "" ;
      float CSP_TH2(Time, bottom_top) ;
            CSP_TH2:FieldType = 104 ;
            CSP_TH2:MemoryOrder = "Z  " ;
            CSP_TH2:description = "Variance of th" ;
            CSP_TH2:units = "K^2" ;
            CSP_TH2:stagger = "" ;
      float CSP_THV2(Time, bottom_top) ;
            CSP_THV2:FieldType = 104 ;
            CSP_THV2:MemoryOrder = "Z  " ;
            CSP_THV2:description = "Variance of thv" ;
            CSP_THV2:units = "K^2" ;
            CSP_THV2:stagger = "" ;
      float CSP_THL2(Time, bottom_top) ;
            CSP_THL2:FieldType = 104 ;
            CSP_THL2:MemoryOrder = "Z  " ;
            CSP_THL2:description = "Variance of thl" ;
            CSP_THL2:units = "K^2" ;
            CSP_THL2:stagger = "" ;
      float CSP_QV2(Time, bottom_top) ;
            CSP_QV2:FieldType = 104 ;
            CSP_QV2:MemoryOrder = "Z  " ;
            CSP_QV2:description = "Variance of qv" ;
            CSP_QV2:units = "kg^2/kg^2" ;
            CSP_QV2:stagger = "" ;
      float CSP_SMAXACTMAX(Time, bottom_top) ;
            CSP_SMAXACTMAX:FieldType = 104 ;
            CSP_SMAXACTMAX:MemoryOrder = "Z  " ;
            CSP_SMAXACTMAX:description = "Max of max supersat in Morrison microphys-
ics" ;
            CSP_SMAXACTMAX:units = "" ;
            CSP_SMAXACTMAX:stagger = "" ;
      float CSP_RMINACTMIN(Time, bottom_top) ;
            CSP_RMINACTMIN:FieldType = 104 ;
            CSP_RMINACTMIN:MemoryOrder = "Z  " ;
            CSP_RMINACTMIN:description = "Min of min activated radius in Morrison mi-
crophysics" ;
            CSP_RMINACTMIN:units = "" ;
            CSP_RMINACTMIN:stagger = "" ;
      float CSP_WMAX(Time, bottom_top) ;
            CSP_WMAX:FieldType = 104 ;
            CSP_WMAX:MemoryOrder = "Z  " ;
            CSP_WMAX:description = "Max value of vertical motion" ;
            CSP_WMAX:units = "m s^-1" ;
            CSP_WMAX:stagger = "" ;
      float CSP_WMIN(Time, bottom_top) ;
            CSP_WMIN:FieldType = 104 ;
            CSP_WMIN:MemoryOrder = "Z  " ;
            CSP_WMIN:description = "Min value of vertical motion" ;
            CSP_WMIN:units = "m s^-1" ;
            CSP_WMIN:stagger = "" ;
```

```
float CSV_TH(Time, bottom_top, south_north, west_east) ;
       CSV_TH:FieldType = 104 ;
       CSV_TH:MemoryOrder = "XYZ" ;
       CSV_TH:description = "Time-averaged potential temperature" ;
       CSV_TH:units = "m s^-1" ;
       CSV_TH:stagger = "" ;
       CSV_TH:coordinates = "XLONG XLAT XTIME" ;
float CSV_U(Time, bottom_top, south_north, west_east_stag) ;
       CSV_U:FieldType = 104 ;
       CSV_U:MemoryOrder = "XYZ" ;
       CSV_U:description = "Time-averaged meridional wind speed" ;
       CSV_U:units = "m s^-1" ;
       CSV_U:stagger = "X" ;
       CSV_U:coordinates = "XLONG_U XLAT_U XTIME" ;
float CSV_V(Time, bottom_top, south_north_stag, west_east) ;
       CSV_V:FieldType = 104 ;
       CSV_V:MemoryOrder = "XYZ" ;
       CSV_V:description = "Time-averaged zonal wind speed" ;
       CSV_V:units = "m s^-1" ;
       CSV_V:stagger = "Y" ;
       CSV_V:coordinates = "XLONG_V XLAT_V XTIME" ;
float CSV_W(Time, bottom_top_stag, south_north, west_east) ;
       CSV_W:FieldType = 104 ;
       CSV_W:MemoryOrder = "XYZ" ;
       CSV_W:description = "Time-averaged vertical wind speed" ;
       CSV_W:units = "m s^-1" ;
       CSV_W:stagger = "Z" ;
       CSV_W:coordinates = "XLONG XLAT XTIME" ;
float CSV_W2(Time, bottom_top_stag, south_north, west_east) ;
       CSV_W2:FieldType = 104 ;
       CSV_W2:MemoryOrder = "XYZ" ;
       CSV_W2:description = "Time-averaged vertical wind speed variance" ;
       CSV_W2:units = "m^2 s^-2" ;
       CSV_W2:stagger = "Z" ;
       CSV_W2:coordinates = "XLONG XLAT XTIME" ;
float CSV_QV(Time, bottom_top, south_north, west_east) ;
       CSV_QV:FieldType = 104 ;
       CSV_QV:MemoryOrder = "XYZ" ;
       CSV_QV:description = "Time-averaged water vapor mixing ratio" ;
       CSV_QV:units = "kg/kg" ;
       CSV_QV:stagger = "" ;
       CSV_QV:coordinates = "XLONG XLAT XTIME" ;
float CSV_QC(Time, bottom_top, south_north, west_east) ;
       CSV_QC:FieldType = 104 ;
       CSV_QC:MemoryOrder = "XYZ" ;
       CSV_QC:description = "Time-averaged cloud droplet mixing ratio" ;
       CSV_QC:units = "kg/kg" ;
       CSV_QC:stagger = "" ;
       CSV_QC:coordinates = "XLONG XLAT XTIME" ;
float CSV_QR(Time, bottom_top, south_north, west_east) ;
       CSV_QR:FieldType = 104 ;
       CSV_QR:MemoryOrder = "XYZ" ;
       CSV_QR:description = "Time-averaged rain droplet mixing ratio" ;
       CSV_QR:units = "kg/kg" ;
       CSV_QR:stagger = "" ;
       CSV_QR:coordinates = "XLONG XLAT XTIME" ;
float CSV_QI(Time, bottom_top, south_north, west_east) ;
       CSV_QI:FieldType = 104 ;
       CSV_QI:MemoryOrder = "XYZ" ;
       CSV_QI:description = "Time-averaged cloud ice mixing ratio" ;
       CSV_QI:units = "kg/kg" ;
```

```
        CSV_QI:stagger = “” ;
        CSV_QI:coordinates = “XLONG XLAT XTIME” ;
float CSV_QS(Time, bottom_top, south_north, west_east) ;
        CSV_QS:FieldType = 104 ;
        CSV_QS:MemoryOrder = “XYZ” ;
        CSV_QS:description = “Time-averaged snow mixing ratio” ;
        CSV_QS:units = “kg/kg” ;
        CSV_QS:stagger = “” ;
        CSV_QS:coordinates = “XLONG XLAT XTIME” ;
float CSV_QG(Time, bottom_top, south_north, west_east) ;
        CSV_QG:FieldType = 104 ;
        CSV_QG:MemoryOrder = “XYZ” ;
        CSV_QG:description = “Time-averaged graupel mixing ratio” ;
        CSV_QG:units = “kg/kg” ;
        CSV_QG:stagger = “” ;
        CSV_QG:coordinates = “XLONG XLAT XTIME” ;
float CSV_LWC(Time, bottom_top, south_north, west_east) ;
        CSV_LWC:FieldType = 104 ;
        CSV_LWC:MemoryOrder = “XYZ” ;
        CSV_LWC:description = “Time-averaged liquid water content (based on ql)” ;
        CSV_LWC:units = “kg/m^3” ;
        CSV_LWC:stagger = “” ;
        CSV_LWC:coordinates = “XLONG XLAT XTIME” ;
float CSV_IWC(Time, bottom_top, south_north, west_east) ;
        CSV_IWC:FieldType = 104 ;
        CSV_IWC:MemoryOrder = “XYZ” ;
        CSV_IWC:description = “Time-averaged ice water content (based on qf)” ;
        CSV_IWC:units = “kg/m^3” ;
        CSV_IWC:stagger = “” ;
        CSV_IWC:coordinates = “XLONG XLAT XTIME” ;
float CSV_CLDFRAC(Time, bottom_top, south_north, west_east) ;
        CSV_CLDFRAC:FieldType = 104 ;
        CSV_CLDFRAC:MemoryOrder = “XYZ” ;
        CSV_CLDFRAC:description = “Time-averaged cloud fraction” ;
        CSV_CLDFRAC:units = “(0-1)” ;
        CSV_CLDFRAC:stagger = “” ;
        CSV_CLDFRAC:coordinates = “XLONG XLAT XTIME” ;
float CSS_LWP(Time, south_north, west_east) ;
        CSS_LWP:FieldType = 104 ;
        CSS_LWP:MemoryOrder = “XY “ ;
        CSS_LWP:description = “Time-averaged liquid water path (based on ql)” ;
        CSS_LWP:units = “kg/m^3” ;
        CSS_LWP:stagger = “” ;
        CSS_LWP:coordinates = “XLONG XLAT XTIME” ;
float CSS_IWP(Time, south_north, west_east) ;
        CSS_IWP:FieldType = 104 ;
        CSS_IWP:MemoryOrder = “XY “ ;
        CSS_IWP:description = “Time-averaged ice water path (based on qf)” ;
        CSS_IWP:units = “kg/m^3” ;
        CSS_IWP:stagger = “” ;
        CSS_IWP:coordinates = “XLONG XLAT XTIME” ;
float CSS_CLDTOT(Time, south_north, west_east) ;
        CSS_CLDTOT:FieldType = 104 ;
        CSS_CLDTOT:MemoryOrder = “XY “ ;
        CSS_CLDTOT:description = “Time-averaged fractional cloud cover” ;
        CSS_CLDTOT:units = “(0-1)” ;
        CSS_CLDTOT:stagger = “” ;
        CSS_CLDTOT:coordinates = “XLONG XLAT XTIME” ;
```

```
      float CSS_CLDLOW(Time, south_north, west_east) ;
             CSS_CLDLOW:FieldType = 104 ;
             CSS_CLDLOW:MemoryOrder = "XY " ;
             CSS_CLDLOW:description = "Time-averaged fractional low-cloud cover (<5
km)" ;
             CSS_CLDLOW:units = "(0-1)" ;
             CSS_CLDLOW:stagger = "" ;
             CSS_CLDLOW:coordinates = "XLONG XLAT XTIME" ;

// global attributes:
             :TITLE = " OUTPUT FROM WRF V3.8.1 MODEL" ;
             :START_DATE = "2016-06-10_12:00:00" ;
             :WEST-EAST_GRID_DIMENSION = 145 ;
             :SOUTH-NORTH_GRID_DIMENSION = 145 ;
             :BOTTOM-TOP_GRID_DIMENSION = 227 ;
             :DX = 100.f ;
             :DY = 100.f ;
             :GRIDTYPE = "C" ;
             :DIFF_OPT = 2 ;
             :KM_OPT = 2 ;
             :DAMP_OPT = 3 ;
             :DAMPCOEF = 0.2f ;
             :KHDIF = 1.f ;
             :KVDIF = 1.f ;
             :MP_PHYSICS = 50 ;
             :RA_LW_PHYSICS = 4 ;
             :RA_SW_PHYSICS = 4 ;
             :SF_SFCLAY_PHYSICS = 1 ;
             :SF_SURFACE_PHYSICS = 1 ;
             :BL_PBL_PHYSICS = 0 ;
             :CU_PHYSICS = 0 ;
             :SF_LAKE_PHYSICS = 0 ;
             :SURFACE_INPUT_SOURCE = 3 ;
             :SST_UPDATE = 0 ;
             :GRID_FDDA = 0 ;
             :GFDDA_INTERVAL_M = 0 ;
             :GFDDA_END_H = 0 ;
             :GRID_SFDDA = 0 ;
             :SGFDDA_INTERVAL_M = 0 ;
             :SGFDDA_END_H = 0 ;
             :HYPSOMETRIC_OPT = 1 ;
             :USE_THETA_M = 1 ;
             :WEST-EAST_PATCH_START_UNSTAG = 1 ;
             :WEST-EAST_PATCH_END_UNSTAG = 144 ;
             :WEST-EAST_PATCH_START_STAG = 1 ;
             :WEST-EAST_PATCH_END_STAG = 145 ;
             :SOUTH-NORTH_PATCH_START_UNSTAG = 1 ;
             :SOUTH-NORTH_PATCH_END_UNSTAG = 144 ;
             :SOUTH-NORTH_PATCH_START_STAG = 1 ;
             :SOUTH-NORTH_PATCH_END_STAG = 145 ;
             :BOTTOM-TOP_PATCH_START_UNSTAG = 1 ;
             :BOTTOM-TOP_PATCH_END_UNSTAG = 226 ;
             :BOTTOM-TOP_PATCH_START_STAG = 1 ;
             :BOTTOM-TOP_PATCH_END_STAG = 227 ;
             :GRID_ID = 1 ;
             :PARENT_ID = 0 ;
             :I_PARENT_START = 0 ;
             :J_PARENT_START = 0 ;
             :PARENT_GRID_RATIO = 1 ;
             :DT = 0.5f ;
```

```
            :CEN_LAT = 0.f ;
            :CEN_LON = 0.f ;
            :TRUELAT1 = 0.f ;
            :TRUELAT2 = 0.f ;
            :MOAD_CEN_LAT = 0.f ;
            :STAND_LON = 0.f ;
            :POLE_LAT = 0.f ;
            :POLE_LON = 0.f ;
            :GMT = 0.f ;
            :JULYR = 0 ;
            :JULDAY = 1 ;
            :MAP_PROJ = 0 ;
            :MAP_PROJ_CHAR = "Cartesian" ;
            :MMINLU = "" ;
            :NUM_LAND_CAT = 21 ;
            :ISWATER = 16 ;
            :ISLAKE = 0 ;
            :ISICE = 0 ;
            :ISURBAN = 0 ;
            :ISOILWATER = 0 ;
}
```

File: .../20160610/sim0100/raw_model/ sgp20160610_alpha2.varanal_300km_25mb_ls.varanal_300km_25mb_sf.sonde_init.144.nc

Description: 1D SAM output averaged between output times. The time label represents the middle of the averaging period.

```
netcdf sgp20160610_alpha2.varanal_300km_25mb_ls.varanal_300km_25mb_sf.sonde_init.144 {
dimensions:
      z = 226 ;
      time = 90 ;
variables:
      float z(z) ;
            z:units = "m" ;
            z:long_name = "height" ;
      float time(time) ;
            time:units = "day" ;
            time:long_name = "time" ;
      float p(z) ;
            p:units = "mb" ;
            p:long_name = "pressure" ;
      float SST(time) ;
            SST:long_name = "SST" ;
            SST:units = "K" ;
      float Ps(time) ;
            Ps:long_name = "Surface Pressure" ;
            Ps:units = "" ;
      float CLDSHD(time) ;
            CLDSHD:long_name = "Shaded Cloud Fraction" ;
            CLDSHD:units = "" ;
      float AREAPREC(time) ;
            AREAPREC:long_name = "Surface Precip. Fraction" ;
            AREAPREC:units = "" ;
      float CLD245(time) ;
            CLD245:long_name = "Cloud Fraction above 245K level" ;
            CLD245:units = "" ;
      float WMAX(time) ;
```

```
        WMAX:long_name = "Maximum Updraft Velocity" ;
        WMAX:units = "m/s" ;
float UMAX(time) ;
        UMAX:long_name = "Maximum Horizontal Wind" ;
        UMAX:units = "m/s" ;
float PREC(time) ;
        PREC:long_name = "Surface Precipitation" ;
        PREC:units = "mm/day" ;
float LHF(time) ;
        LHF:long_name = "Latent Heat Flux" ;
        LHF:units = "W/m2" ;
float SHF(time) ;
        SHF:long_name = "Sensible Heat Flux" ;
        SHF:units = "W/m2" ;
float PW(time) ;
        PW:long_name = "Precipitable Water" ;
        PW:units = "mm" ;
float PWOBS(time) ;
        PWOBS:long_name = "Observed Precipitable Water" ;
        PWOBS:units = "mm" ;
float CWP(time) ;
        CWP:long_name = "Cloud Water Path" ;
        CWP:units = "g/m2" ;
float IWP(time) ;
        IWP:long_name = "Ice Water Path" ;
        IWP:units = "g/m2" ;
float RWP(time) ;
        RWP:long_name = "Rain Water Path" ;
        RWP:units = "g/m2" ;
float SWP(time) ;
        SWP:long_name = "Snow Water Path" ;
        SWP:units = "g/m2" ;
float GWP(time) ;
        GWP:long_name = "Grauple Water Path" ;
        GWP:units = "g/m2" ;
float CAPE(time) ;
        CAPE:long_name = "CAPE" ;
        CAPE:units = "J/kg" ;
float CAPEOBS(time) ;
        CAPEOBS:long_name = "CAPEOBS" ;
        CAPEOBS:units = "J/kg" ;
float CIN(time) ;
        CIN:long_name = "CIN" ;
        CIN:units = "J/kg" ;
float CINOBS(time) ;
        CINOBS:long_name = "CINOBS" ;
        CINOBS:units = "J/kg" ;
float LWNS(time) ;
        LWNS:long_name = "Net LW flux at sfc" ;
        LWNS:units = "W/m2" ;
float LWNT(time) ;
        LWNT:long_name = "Net LW flux at Top-of-Model)" ;
        LWNT:units = "W/m2" ;
float LWNTOA(time) ;
        LWNTOA:long_name = "Net LW flux at TOA" ;
        LWNTOA:units = "W/m2" ;
float LWNSC(time) ;
        LWNSC:long_name = "Net LW flux at sfc (Clear Sky)" ;
        LWNSC:units = "W/m2" ;
```

```
float LWNTOAC(time) ;
       LWNTOAC:long_name = “Net LW flux at TOA (Clear Sky)” ;
       LWNTOAC:units = “W/m2” ;
float LWDS(time) ;
       LWDS:long_name = “Downward LW flux at sfc” ;
       LWDS:units = “W/m2” ;
float SWNS(time) ;
       SWNS:long_name = “Net SW flux at sfc” ;
       SWNS:units = “W/m2” ;
float SWNT(time) ;
       SWNT:long_name = “Net SW flux at Top-of-Model” ;
       SWNT:units = “W/m2” ;
float SWNTOA(time) ;
       SWNTOA:long_name = “Net SW flux at TOA” ;
       SWNTOA:units = “W/m2” ;
float SWNSC(time) ;
       SWNSC:long_name = “Net SW flux at sfc (Clear Sky)” ;
       SWNSC:units = “W/m2” ;
float SWNTOAC(time) ;
       SWNTOAC:long_name = “Net SW flux at TOA (Clear Sky)” ;
       SWNTOAC:units = “W/m2” ;
float SWDS(time) ;
       SWDS:long_name = “Downward SW flux at sfc” ;
       SWDS:units = “W/m2” ;
float SOLIN(time) ;
       SOLIN:long_name = “Incoming SW flux at TOA” ;
       SOLIN:units = “W/m2” ;
float SSTOBS(time) ;
       SSTOBS:long_name = “Observed SST” ;
       SSTOBS:units = “K” ;
float LHFOBS(time) ;
       LHFOBS:long_name = “Observed Latent Heat Flux” ;
       LHFOBS:units = “W/m2” ;
float SHFOBS(time) ;
       SHFOBS:long_name = “Observed Sensible Heat Flux” ;
       SHFOBS:units = “SHFOBS” ;
float CLDLOW(time) ;
       CLDLOW:long_name = “Low Cloud Fraction” ;
       CLDLOW:units = “” ;
float CLDMID(time) ;
       CLDMID:long_name = “Middle Cloud Fraction” ;
       CLDMID:units = “” ;
float CLDHI(time) ;
       CLDHI:long_name = “High Cloud Fraction” ;
       CLDHI:units = “” ;
float ISCCPTOT(time) ;
       ISCCPTOT:long_name = “ISCCP Total Cloud Fraction (tau > 0.3)” ;
       ISCCPTOT:units = “” ;
       ISCCPTOT:_FillValue = -1.f ;
float ISCCPLOW(time) ;
       ISCCPLOW:long_name = “ISCCP Low Cloud Fraction (tau > 0.3)” ;
       ISCCPLOW:units = “” ;
       ISCCPLOW:_FillValue = -1.f ;
float ISCCPMID(time) ;
       ISCCPMID:long_name = “ISCCP Middle Cloud Fraction (tau > 0.3)” ;
       ISCCPMID:units = “” ;
       ISCCPMID:_FillValue = -1.f ;
```

```
float ISCCPHGH(time) ;
       ISCCPHGH:long_name = "ISCCP High Cloud Fraction (tau > 0.3)" ;
       ISCCPHGH:units = "" ;
       ISCCPHGH:_FillValue = -1.f ;
float MODISTOT(time) ;
       MODISTOT:long_name = "MODIS Total Cloud Fraction" ;
       MODISTOT:units = "" ;
       MODISTOT:_FillValue = -1.f ;
float MODISLOW(time) ;
       MODISLOW:long_name = "MODIS Low Cloud Fraction" ;
       MODISLOW:units = "" ;
       MODISLOW:_FillValue = -1.f ;
float MODISMID(time) ;
       MODISMID:long_name = "MODIS Middle Cloud Fraction" ;
       MODISMID:units = "" ;
       MODISMID:_FillValue = -1.f ;
float MODISHGH(time) ;
       MODISHGH:long_name = "MODIS High Cloud Fraction (tau > 0.3)" ;
       MODISHGH:units = "" ;
       MODISHGH:_FillValue = -1.f ;
float MISRTOT(time) ;
       MISRTOT:long_name = "MISR Total Cloud Fraction" ;
       MISRTOT:units = "" ;
       MISRTOT:_FillValue = -1.f ;
float MODISREL(time) ;
       MODISREL:long_name = "MODIS Effective Radius (Liquid)" ;
       MODISREL:units = "mkm" ;
       MODISREL:_FillValue = -1.f ;
float MODISREI(time) ;
       MODISREI:long_name = "MODIS Effective Radius (Ice)" ;
       MODISREI:units = "mkm" ;
       MODISREI:_FillValue = -1.f ;
float MODISLWP(time) ;
       MODISLWP:long_name = "MODIS Liquid Water Path" ;
       MODISLWP:units = "g/m2" ;
       MODISLWP:_FillValue = -1.f ;
float MODISIWP(time) ;
       MODISIWP:long_name = "MODIS Ice Water Path" ;
       MODISIWP:units = "g/m2" ;
       MODISIWP:_FillValue = -1.f ;
float ISCCPTB(time) ;
       ISCCPTB:long_name = "ISCCP Brightness Temperature" ;
       ISCCPTB:units = "K" ;
       ISCCPTB:_FillValue = -1.f ;
float ISCCPTBCLR(time) ;
       ISCCPTBCLR:long_name = "ISCCP Brightness Temperature (Clear Sky)" ;
       ISCCPTBCLR:units = "K" ;
       ISCCPTBCLR:_FillValue = -1.f ;
float MODISTOTL(time) ;
       MODISTOTL:long_name = "MODIS Total Fraction (Liquid)" ;
       MODISTOTL:units = "" ;
       MODISTOTL:_FillValue = -1.f ;
float MODISTOTI(time) ;
       MODISTOTI:long_name = "MODIS Total Fraction (Ice)" ;
       MODISTOTI:units = "" ;
       MODISTOTI:_FillValue = -1.f ;
float ISCCPTAU(time) ;           ISCCPTAU:long_name = "ISCCP Optical Path" ;
       ISCCPTAU:units = "" ;
       ISCCPTAU:_FillValue = -1.f ;
```

```
float ISCCPALB(time) ;
       ISCCPALB:long_name = "ISCCP Cloud Albedo" ;
       ISCCPALB:units = "" ;
       ISCCPALB:_FillValue = -1.f ;
float ISCCPPTOP(time) ;
       ISCCPPTOP:long_name = "ISCCP Cloud-Top Pressure" ;
       ISCCPPTOP:units = "mb" ;
       ISCCPPTOP:_FillValue = -1.f ;
float MODISTAU(time) ;
       MODISTAU:long_name = "MODIS Cloud Optical Path" ;
       MODISTAU:units = "" ;
       MODISTAU:_FillValue = -1.f ;
float MODISTAUL(time) ;
       MODISTAUL:long_name = "MODIS Cloud Optical Path (Liquid)" ;
       MODISTAUL:units = "" ;
       MODISTAUL:_FillValue = -1.f ;
float MODISTAUI(time) ;
       MODISTAUI:long_name = "MODIS Cloud Optical Path (Ice)" ;
       MODISTAUI:units = "" ;
       MODISTAUI:_FillValue = -1.f ;
float MODISPTOP(time) ;
       MODISPTOP:long_name = "MODIS Cloud-Top Pressure" ;
       MODISPTOP:units = "mb" ;
       MODISPTOP:_FillValue = -1.f ;    float MISRZTOP(time) ;
       MISRZTOP:long_name = "MISR Cloud-Top Height" ;
       MISRZTOP:units = "km" ;
       MISRZTOP:_FillValue = -1.f ;
float ZINV(time) ;
       ZINV:long_name = "GCSS Inversion Height" ;
       ZINV:units = "km" ;
       ZINV:_FillValue = -1.f ;
float ZINV2(time) ;
       ZINV2:long_name = "GCSS Variance of the Inversion Height" ;
       ZINV2:units = "km2" ;
       ZINV2:_FillValue = -1.f ;
float ZCT(time) ;
       ZCT:long_name = "GCSS Mean Cloud-top Height" ;
       ZCT:units = "km" ;
       ZCT:_FillValue = -1.f ;
float ZCT2(time) ;
       ZCT2:long_name = "GCSS Variance of Cloud-top Height" ;
       ZCT2:units = "km2" ;
       ZCT2:_FillValue = -1.f ;
float ZCTMAX(time) ;
       ZCTMAX:long_name = "GCSS Maximum Cloud-top Height" ;
       ZCTMAX:units = "km" ;
       ZCTMAX:_FillValue = -1.f ;
float ZCB(time) ;
       ZCB:long_name = "GCSS Mean Cloud-base Height" ;
       ZCB:units = "km" ;
       ZCB:_FillValue = -1.f ;
float ZCB2(time) ;
       ZCB2:long_name = "GCSS Variance of Cloud-base Height" ;
       ZCB2:units = "km" ;
       ZCB2:_FillValue = -1.f ;
float ZCBMIN(time) ;
       ZCBMIN:long_name = "GCSS Minimum Cloud-base Height" ;
       ZCBMIN:units = "km" ;
       ZCBMIN:_FillValue = -1.f ;
```

```
float LWP(time) ;
       LWP:long_name = "GCSS Liquid Water Path" ;
       LWP:units = "g/m2" ;
       LWP:_FillValue = -1.f ;
float LWP2(time) ;
       LWP2:long_name = "GCSS Variance of Liquid Water Path" ;
       LWP2:units = "(g/m2)^2" ;
       LWP2:_FillValue = -1.f ;
float PRECMN(time) ;
       PRECMN:long_name = "GCSS Precipitation Rate" ;
       PRECMN:units = "mm/d" ;
       PRECMN:_FillValue = -1.f ;
float PREC2(time) ;
       PREC2:long_name = "GCSS Variance of Precipitation Rate" ;
       PREC2:units = "(mm/d)^2" ;
       PREC2:_FillValue = -1.f ;
float PRECMAX(time) ;
       PRECMAX:long_name = "GCSS Maximum Precipitation Rate" ;
       PRECMAX:units = "mm/d" ;
       PRECMAX:_FillValue = -1.f ;
float NCMN(time) ;
       NCMN:long_name = "GCSS Mean Drop Number Comcentration" ;
       NCMN:units = "#/cm3" ;
       NCMN:_FillValue = -1.f ;
float NRMN(time) ;
       NRMN:long_name = "GCSS Mean Rain Number Comcentration" ;
       NRMN:units = "#/cm3" ;
       NRMN:_FillValue = -1.f ;
float AREAPRTHR(time) ;
       AREAPRTHR:long_name = "GCSS Precip. over threshold Area Fraction" ;
       AREAPRTHR:units = "" ;
       AREAPRTHR:_FillValue = -1.f ;
float PRES(time, z) ;
       PRES:long_name = "Pressure" ;
       PRES:units = "mb" ;
       PRES:missing_value = -9999.f ;
float U(time, z) ;
       U:long_name = "x wind component" ;
       U:units = "m/s" ;
       U:missing_value = -9999.f ;
float V(time, z) ;
       V:long_name = "y wind component" ;
       V:units = "m/s" ;
       V:missing_value = -9999.f ;
float UOBS(time, z) ;
       UOBS:long_name = "Observed x wind component" ;
       UOBS:units = "m/s" ;
       UOBS:missing_value = -9999.f ;
float VOBS(time, z) ;
       VOBS:long_name = "Observed y wind component" ;
       VOBS:units = "m/s" ;
       VOBS:missing_value = -9999.f ;
float WOBS(time, z) ;
       WOBS:long_name = "Observed large-scale vert. velocity" ;
       WOBS:units = "m/s" ;
       WOBS:missing_value = -9999.f ;
float RHO(time, z) ;
       RHO:long_name = "Air density" ;
       RHO:units = "kg/m3" ;
       RHO:missing_value = -9999.f ;
```

```
float MSE(time, z) ;
       MSE:long_name = "Moist static energy" ;
       MSE:units = "K" ;
       MSE:missing_value = -9999.f ;
float DSE(time, z) ;
       DSE:long_name = "Dry static energy" ;
       DSE:units = "K" ;
       DSE:missing_value = -9999.f ;
float SSE(time, z) ;
       SSE:long_name = "Saturation static energy" ;
       SSE:units = "K" ;
       SSE:missing_value = -9999.f ;
float THETAE(time, z) ;
       THETAE:long_name = "Equivalent (generalized) potential temperature" ;
       THETAE:units = "K" ;
       THETAE:missing_value = -9999.f ;
float THETA(time, z) ;
       THETA:long_name = "Potential temperature" ;
       THETA:units = "K" ;
       THETA:missing_value = -9999.f ;
float THETAV(time, z) ;
       THETAV:long_name = "Virtual potential temperature" ;
       THETAV:units = "K" ;
       THETAV:missing_value = -9999.f ;
float THETAL(time, z) ;
       THETAL:long_name = "Liquid water potential temperature" ;
       THETAL:units = "K" ;
       THETAL:missing_value = -9999.f ;
float TABS(time, z) ;
       TABS:long_name = "Absolute temperature" ;
       TABS:units = "K" ;
       TABS:missing_value = -9999.f ;
float TABSOBS(time, z) ;
       TABSOBS:long_name = "Observed Absolute temperature" ;
       TABSOBS:units = "K" ;
       TABSOBS:missing_value = -9999.f ;
float TL(time, z) ;
       TL:long_name = "Liquid water static energy" ;
       TL:units = "K" ;
       TL:missing_value = -9999.f ;
float QT(time, z) ;
       QT:long_name = "Total water (no rain/snow included)" ;
       QT:units = "g/kg" ;
       QT:missing_value = -9999.f ;
float QV(time, z) ;
       QV:long_name = "Water vapor" ;
       QV:units = "g/kg" ;
       QV:missing_value = -9999.f ;
float QVOBS(time, z) ;
       QVOBS:long_name = "Observed Water vapor" ;
       QVOBS:units = "g/kg" ;
       QVOBS:missing_value = -9999.f ;
float TTEND(time, z) ;
       TTEND:long_name = "Observed Large-Scale Temperature Tendency" ;
       TTEND:units = "K/day" ;
       TTEND:missing_value = -9999.f ;
float QTEND(time, z) ;
       QTEND:long_name = "Observed Large-Scale Moisture Tendency" ;
       QTEND:units = "g/kg/day" ;
       QTEND:missing_value = -9999.f ;
```

```
float QCL(time, z) ;
       QCL:long_name = "Cloud water" ;
       QCL:units = "g/kg" ;
       QCL:missing_value = -9999.f ;
float QCI(time, z) ;
       QCI:long_name = "Cloud ice" ;
       QCI:units = "g/kg" ;
       QCI:missing_value = -9999.f ;
float QPL(time, z) ;
       QPL:long_name = "Rain content" ;
       QPL:units = "g/kg" ;
       QPL:missing_value = -9999.f ;
float QPI(time, z) ;
       QPI:long_name = "Snow content" ;
       QPI:units = "g/kg" ;
       QPI:missing_value = -9999.f ;
float QN(time, z) ;
       QN:long_name = "Cloud water and cloud ice" ;
       QN:units = "g/kg" ;
       QN:missing_value = -9999.f ;
float QP(time, z) ;
       QP:long_name = "Rain and Snow" ;
       QP:units = "g/kg" ;
       QP:missing_value = -9999.f ;
float QSAT(time, z) ;
       QSAT:long_name = "Saturation mixing ratio" ;
       QSAT:units = "g/kg" ;
       QSAT:missing_value = -9999.f ;
float QCOND(time, z) ;
       QCOND:long_name = "Total Condensate" ;
       QCOND:units = "g/kg" ;
       QCOND:missing_value = -9999.f ;
float PRECIP(time, z) ;
       PRECIP:long_name = "Precipitation flux" ;
       PRECIP:units = "mm/day" ;
       PRECIP:missing_value = -9999.f ;
float RELH(time, z) ;
       RELH:long_name = "Relative humidity" ;
       RELH:units = "per cent" ;
       RELH:missing_value = -9999.f ;
float TLFLUX(time, z) ;
       TLFLUX:long_name = "Liquid water static energy flux (Total)" ;
       TLFLUX:units = "W/m2" ;
       TLFLUX:missing_value = -9999.f ;
float TLFLUXS(time, z) ;
       TLFLUXS:long_name = "Liquid water static energy flux (SGS)" ;
       TLFLUXS:units = "W/m2" ;
       TLFLUXS:missing_value = -9999.f ;
float TVFLUX(time, z) ;
       TVFLUX:long_name = "Buoyancy flux (Resolved)" ;
       TVFLUX:units = "W/m2" ;
       TVFLUX:missing_value = -9999.f ;
float QCFLUX(time, z) ;
       QCFLUX:long_name = "Liquid water flux (Resolved)" ;
       QCFLUX:units = "W/m2" ;
       QCFLUX:missing_value = -9999.f ;
float QIFLUX(time, z) ;
       QIFLUX:long_name = "Ice flux (Resolved)" ;
       QIFLUX:units = "W/m2" ;
       QIFLUX:missing_value = -9999.f ;
```

```
float UW(time, z) ;
      UW:long_name = "x-momentum flux (Total)" ;
      UW:units = "m2/s2" ;
      UW:missing_value = -9999.f ;
float UWSB(time, z) ;
      UWSB:long_name = "x-momentum flux (SGS)" ;
      UWSB:units = "m2/s2" ;          UWSB:missing_value = -9999.f ;
float VW(time, z) ;
      VW:long_name = "y-momentum flux (Total)" ;
      VW:units = "m2/s2" ;
      VW:missing_value = -9999.f ;
float VWSB(time, z) ;
      VWSB:long_name = "y-momentum flux (SGS)" ;
      VWSB:units = "m2/s2" ;
      VWSB:missing_value = -9999.f ;
float RADLWUP(time, z) ;
      RADLWUP:long_name = "Upward longwave radiative flux" ;
      RADLWUP:units = "W/m2" ;
      RADLWUP:missing_value = -9999.f ;
float RADLWDN(time, z) ;
      RADLWDN:long_name = "Downward longwave radiative flux" ;
      RADLWDN:units = "W/m2" ;
      RADLWDN:missing_value = -9999.f ;
float RADSWUP(time, z) ;
      RADSWUP:long_name = "Upward shortwave radiative flux" ;
      RADSWUP:units = "W/m2" ;
      RADSWUP:missing_value = -9999.f ;
float RADSWDN(time, z) ;
      RADSWDN:long_name = "Downward shortwave radiative flux" ;
      RADSWDN:units = "W/m2" ;
      RADSWDN:missing_value = -9999.f ;
float RADQRLW(time, z) ;
      RADQRLW:long_name = "Longwave heating rate" ;
      RADQRLW:units = "K/day" ;
      RADQRLW:missing_value = -9999.f ;
float RADQRSW(time, z) ;
      RADQRSW:long_name = "Shortwave heating rate" ;
      RADQRSW:units = "K/day" ;
      RADQRSW:missing_value = -9999.f ;
float RADQR(time, z) ;
      RADQR:long_name = "Radiative heating rate" ;
      RADQR:units = "K/day" ;
      RADQR:missing_value = -9999.f ;
float RADQRS(time, z) ;
      RADQRS:long_name = "Clear-sky Radiative heating rate" ;
      RADQRS:units = "K/day" ;
      RADQRS:missing_value = -9999.f ;
float RADQRC(time, z) ;
      RADQRC:long_name = "Cloudy-sky Radiative heating rate" ;
      RADQRC:units = "K/day" ;
      RADQRC:missing_value = -9999.f ;
float Q1C(time, z) ;
      Q1C:long_name = "Apparent heat source: Q1 - QR" ;
      Q1C:units = "K/day" ;
      Q1C:missing_value = -9999.f ;
float Q2(time, z) ;
      Q2:long_name = "Apparent moisture sink: Q2" ;
      Q2:units = "K/day" ;
      Q2:missing_value = -9999.f ;
```

```
float U2(time, z) ;
       U2:long_name = "Variance of the x wind component" ;
       U2:units = "m2/s2" ;
       U2:missing_value = -9999.f ;
float V2(time, z) ;
       V2:long_name = "Variance of the y wind component" ;
       V2:units = "m2/s2" ;
       V2:missing_value = -9999.f ;
float W2(time, z) ;
       W2:long_name = "Variance of the z wind component" ;
       W2:units = "m2/s2" ;
       W2:missing_value = -9999.f ;
float TL2(time, z) ;
       TL2:long_name = "Variance of l.w.stat.energy" ;
       TL2:units = "K2" ;
       TL2:missing_value = -9999.f ;
float TQ(time, z) ;
       TQ:long_name = "CoVariance of HL and QT" ;
       TQ:units = "K2" ;
       TQ:missing_value = -9999.f ;
float QT2(time, z) ;
       QT2:long_name = "Variance of total water" ;
       QT2:units = "g2/kg2" ;
       QT2:missing_value = -9999.f ;
float QC2(time, z) ;
       QC2:long_name = "Variance of cloud water" ;
       QC2:units = "g2/kg2" ;
       QC2:missing_value = -9999.f ;
float QI2(time, z) ;
       QI2:long_name = "Variance of cloud ice" ;
       QI2:units = "g2/kg2" ;
       QI2:missing_value = -9999.f ;
float QS2(time, z) ;
       QS2:long_name = "Variance of saturation mixing ratio" ;
       QS2:units = "g2/kg2" ;
       QS2:missing_value = -9999.f ;
float W3(time, z) ;
       W3:long_name = "Third moment of the vertical velocity" ;
       W3:units = "m3/s3" ;
       W3:missing_value = -9999.f ;
float AUP(time, z) ;
       AUP:long_name = "domain fraction occupied by updrafts" ;
       AUP:units = "" ;
       AUP:missing_value = -9999.f ;
float WSKEW(time, z) ;
       WSKEW:long_name = "Vertical velocity skewness W3/(W2)^3/2" ;
       WSKEW:units = "" ;
       WSKEW:missing_value = -9999.f ;
float TKE(time, z) ;
       TKE:long_name = "Turbulent kinetic energy (Resolved)" ;
       TKE:units = "m2/s2" ;
       TKE:missing_value = -9999.f ;
float TKES(time, z) ;
       TKES:long_name = "Turbulent kinetic energy (SGS)" ;
       TKES:units = "m2/s2" ;
       TKES:missing_value = -9999.f ;
float TK(time, z) ;
       TK:long_name = "Eddy viscosity" ;
       TK:units = "m2/s" ;
       TK:missing_value = -9999.f ;
```

```
float TKH(time, z) ;
       TKH:long_name = "Eddy diffusivity" ;
       TKH:units = "m2/s" ;
       TKH:missing_value = -9999.f ;
float HLADV(time, z) ;
       HLADV:long_name = "Advective Transport of HL" ;
       HLADV:units = "K/day" ;
       HLADV:missing_value = -9999.f ;
float HLDIFF(time, z) ;
       HLDIFF:long_name = "Diffusive Transport of HL" ;
       HLDIFF:units = "K/day" ;
       HLDIFF:missing_value = -9999.f ;
float HLLAT(time, z) ;
       HLLAT:long_name = "Latent Heating of HL" ;
       HLLAT:units = "K/day" ;
       HLLAT:missing_value = -9999.f ;
float HLRAD(time, z) ;
       HLRAD:long_name = "Radiative Heating of HL" ;
       HLRAD:units = "K/day" ;
       HLRAD:missing_value = -9999.f ;
float SHEAR(time, z) ;
       SHEAR:long_name = "Shear production of TKE (resolved)" ;
       SHEAR:units = "m2/s3" ;
       SHEAR:missing_value = -9999.f ;
float SHEARS(time, z) ;
       SHEARS:long_name = "Shear production of TKE (SGS)" ;
       SHEARS:units = "m2/s3" ;
       SHEARS:missing_value = -9999.f ;
float BUOYA(time, z) ;
       BUOYA:long_name = "Buoyancy production of TKE (resolved)" ;
       BUOYA:units = "m2/s3" ;
       BUOYA:missing_value = -9999.f ;
float BUOYAS(time, z) ;
       BUOYAS:long_name = "Buoyancy production of TKE (SGS)" ;
       BUOYAS:units = "m2/s3" ;
       BUOYAS:missing_value = -9999.f ;
float ADVTR(time, z) ;
       ADVTR:long_name = "Turbulent advective transport of TKE (Resolved)" ;
       ADVTR:units = "m2/s3" ;
       ADVTR:missing_value = -9999.f ;
float PRESSTR(time, z) ;
       PRESSTR:long_name = "Pressure transport of TKE (Resolved)" ;
       PRESSTR:units = "m2/s3" ;
       PRESSTR:missing_value = -9999.f ;
float ADVTRS(time, z) ;
       ADVTRS:long_name = "Turbulent+pressure transport of TKE (SGS)" ;
       ADVTRS:units = "m2/s3" ;
       ADVTRS:missing_value = -9999.f ;
float DIFTR(time, z) ;
       DIFTR:long_name = "SGS transport of TKE (Resolved)" ;
       DIFTR:units = "m2/s3" ;
       DIFTR:missing_value = -9999.f ;
float DISSIP(time, z) ;
       DISSIP:long_name = "Dissipation (Resolved)" ;
       DISSIP:units = "m2/s3" ;
       DISSIP:missing_value = -9999.f ;
float DISSIPS(time, z) ;
       DISSIPS:long_name = "Dissipation (SGS)" ;
       DISSIPS:units = "m2/s3" ;
       DISSIPS:missing_value = -9999.f ;
```

```
float WUADV(time, z) ;
       WUADV:long_name = "WU advection (Resolved)" ;
       WUADV:units = "m2/s3" ;
       WUADV:missing_value = -9999.f ;
float WVADV(time, z) ;
       WVADV:long_name = "WV advection (Resolved)" ;
       WVADV:units = "m2/s3" ;
       WVADV:missing_value = -9999.f ;
float WUPRES(time, z) ;
       WUPRES:long_name = "WU pressure (Resolved)" ;
       WUPRES:units = "m2/s3" ;
       WUPRES:missing_value = -9999.f ;
float WVPRES(time, z) ;
       WVPRES:long_name = "WV pressure (Resolved)" ;
       WVPRES:units = "m2/s3" ;
       WVPRES:missing_value = -9999.f ;
float WUANIZ(time, z) ;
       WUANIZ:long_name = "WU return to anisotropy (Resolved)" ;
       WUANIZ:units = "m2/s3" ;
       WUANIZ:missing_value = -9999.f ;
float WVANIZ(time, z) ;
       WVANIZ:long_name = "WV return to anisotropy (Resolved)" ;
       WVANIZ:units = "m2/s3" ;
       WVANIZ:missing_value = -9999.f ;
float WUSHEAR(time, z) ;
       WUSHEAR:long_name = "WU shear(Resolved)" ;
       WUSHEAR:units = "m2/s3" ;
       WUSHEAR:missing_value = -9999.f ;
float WVSHEAR(time, z) ;
       WVSHEAR:long_name = "WV shear (Resolved)" ;
       WVSHEAR:units = "m2/s3" ;
       WVSHEAR:missing_value = -9999.f ;
float WUBUOY(time, z) ;
       WUBUOY:long_name = "WU buoyancy (Resolved)" ;
       WUBUOY:units = "m2/s3" ;
       WUBUOY:missing_value = -9999.f ;
float WVBUOY(time, z) ;
       WVBUOY:long_name = "WV buoyancy (Resolved)" ;
       WVBUOY:units = "m2/s3" ;
       WVBUOY:missing_value = -9999.f ;
float WUDIFF(time, z) ;
       WUDIFF:long_name = "WU diffusion (Resolved)" ;
       WUDIFF:units = "m2/s3" ;
       WUDIFF:missing_value = -9999.f ;
float WVDIFF(time, z) ;
       WVDIFF:long_name = "WV diffusion (Resolved)" ;
       WVDIFF:units = "m2/s3" ;
       WVDIFF:missing_value = -9999.f ;
float W2ADV(time, z) ;
       W2ADV:long_name = "WU advection (Resolved)" ;
       W2ADV:units = "m2/s3" ;
       W2ADV:missing_value = -9999.f ;
float W2PRES(time, z) ;
       W2PRES:long_name = "W2 pressure (Resolved)" ;
       W2PRES:units = "m2/s3" ;
       W2PRES:missing_value = -9999.f ;
float W2REDIS(time, z) ;
       W2REDIS:long_name = "W2 pressure redistribution(Resolved)" ;
       W2REDIS:units = "m2/s3" ;
       W2REDIS:missing_value = -9999.f ;
```

```
float W2BUOY(time, z) ;
       W2BUOY:long_name = "W2 buoyancy (Resolved)" ;
       W2BUOY:units = "m2/s3" ;
       W2BUOY:missing_value = -9999.f ;
float W2DIFF(time, z) ;
       W2DIFF:long_name = "W2 diffusion (Resolved)" ;
       W2DIFF:units = "m2/s3" ;
       W2DIFF:missing_value = -9999.f ;
float TWGRAD(time, z) ;
       TWGRAD:long_name = "Gradient production of l.w.s.e flux" ;
       TWGRAD:units = "m s-2 K" ;
       TWGRAD:missing_value = -9999.f ;
float TWADV(time, z) ;
       TWADV:long_name = "Transport of l.w.s.e flux" ;
       TWADV:units = "m s-2 K" ;
       TWADV:missing_value = -9999.f ;
float TWDIFF(time, z) ;
       TWDIFF:long_name = "Diffusion of l.w.s.e flux" ;
       TWDIFF:units = "m s-2 K" ;
       TWDIFF:missing_value = -9999.f ;
float TWBUOY(time, z) ;
       TWBUOY:long_name = "Buoyancy production of l.w.s.e flux" ;
       TWBUOY:units = "m s-2 K" ;
       TWBUOY:missing_value = -9999.f ;
float TWPRES(time, z) ;
       TWPRES:long_name = "Pressure production of l.w.s.e flux" ;
       TWPRES:units = "m s-2 K" ;
       TWPRES:missing_value = -9999.f ;
float TWPREC(time, z) ;
       TWPREC:long_name = "Precip. production of l.w.s.e flux" ;
       TWPREC:units = "m s-2 K" ;
       TWPREC:missing_value = -9999.f ;
float QWGRAD(time, z) ;
       QWGRAD:long_name = "Gradient production of total water flux" ;
       QWGRAD:units = "m s-2 K" ;
       QWGRAD:missing_value = -9999.f ;
float QWADV(time, z) ;
       QWADV:long_name = "Transport of total water flux" ;
       QWADV:units = "m s-2 K" ;
       QWADV:missing_value = -9999.f ;
float QWDIFF(time, z) ;
       QWDIFF:long_name = "Diffusion of total water flux" ;
       QWDIFF:units = "m s-2 K" ;
       QWDIFF:missing_value = -9999.f ;
float QWBUOY(time, z) ;
       QWBUOY:long_name = "Buoyancy production of total water flux" ;
       QWBUOY:units = "m s-2" ;
       QWBUOY:missing_value = -9999.f ;
float QWPRES(time, z) ;
       QWPRES:long_name = "Pressure production of total water flux" ;
       QWPRES:units = "m s-2" ;
       QWPRES:missing_value = -9999.f ;
float QWPREC(time, z) ;
       QWPREC:long_name = "Precip. production of total water flux" ;
       QWPREC:units = "m s-2" ;
       QWPREC:missing_value = -9999.f ;
float T2ADVTR(time, z) ;
       T2ADVTR:long_name = "Transport of l.w.s.e variance" ;
       T2ADVTR:units = "K2/s" ;
       T2ADVTR:missing_value = -9999.f ;
```

```
float T2GRAD(time, z) ;
       T2GRAD:long_name = “Gradient production of l.w.s.e variance” ;
       T2GRAD:units = “K2/s” ;
       T2GRAD:missing_value = -9999.f ;
float T2DISSIP(time, z) ;
       T2DISSIP:long_name = “Dissipation of l.w.s.e variance” ;
       T2DISSIP:units = “K2/s” ;
       T2DISSIP:missing_value = -9999.f ;
float T2DIFTR(time, z) ;
       T2DIFTR:long_name = “SGS transport of l.w.s.e variance” ;
       T2DIFTR:units = “K2/s” ;
       T2DIFTR:missing_value = -9999.f ;
float T2PREC(time, z) ;
       T2PREC:long_name = “Precipitation production of l.w.s.e variance” ;
       T2PREC:units = “K2/s” ;
       T2PREC:missing_value = -9999.f ;
float Q2ADVTR(time, z) ;
       Q2ADVTR:long_name = “Transport of total water variance” ;
       Q2ADVTR:units = “1/s” ;
       Q2ADVTR:missing_value = -9999.f ;
float Q2GRAD(time, z) ;
       Q2GRAD:long_name = “Gradient production of total water variance” ;
       Q2GRAD:units = “1/s” ;
       Q2GRAD:missing_value = -9999.f ;
float Q2DISSIP(time, z) ;
       Q2DISSIP:long_name = “Dissipation of total water variance” ;
       Q2DISSIP:units = “1/s” ;
       Q2DISSIP:missing_value = -9999.f ;
float Q2DIFTR(time, z) ;
       Q2DIFTR:long_name = “SGS transport of total water variance” ;
       Q2DIFTR:units = “1/s” ;
       Q2DIFTR:missing_value = -9999.f ;
float Q2PREC(time, z) ;
       Q2PREC:long_name = “Precipitation production of total water variance” ;
       Q2PREC:units = “1/s” ;
       Q2PREC:missing_value = -9999.f ;
float HYDRO(time, z) ;
       HYDRO:long_name = “Total fraction of hydrometeors” ;
       HYDRO:units = “” ;
       HYDRO:missing_value = -9999.f ;
float MCUP(time, z) ;
       MCUP:long_name = “Updraft cloud mass flux” ;
       MCUP:units = “kg/m2/s” ;
       MCUP:missing_value = -9999.f ;
float MCDNS(time, z) ;
       MCDNS:long_name = “Downdraft saturated cloud mass flux” ;
       MCDNS:units = “kg/m2/s” ;
       MCDNS:missing_value = -9999.f ;
float MCDNU(time, z) ;
       MCDNU:long_name = “Downdraft unsaturated mass flux” ;
       MCDNU:units = “kg/m2/s” ;
       MCDNU:missing_value = -9999.f ;
float MC(time, z) ;
       MC:long_name = “Cloud mass flux” ;
       MC:units = “kg/m2/s” ;
       MC:missing_value = -9999.f ;
float CORECL(time, z) ;
       CORECL:long_name = “Cloudy Updraft core fraction” ;
       CORECL:units = “” ;
       CORECL:missing_value = -9999.f ;
```

```
	float COREDNCL(time, z) ;
		COREDNCL:long_name = "Cloudy Downdraft core fraction" ;
		COREDNCL:units = "" ;
		COREDNCL:missing_value = -9999.f ;
	float MCRUP(time, z) ;
		MCRUP:long_name = "Updraft core mass flux" ;
		MCRUP:units = "kg/m2/s" ;
		MCRUP:missing_value = -9999.f ;
	float MCRDNS(time, z) ;
		MCRDNS:long_name = "Downdraft cloud core mass flux" ;
		MCRDNS:units = "kg/m2/s" ;
		MCRDNS:missing_value = -9999.f ;
	float MCRDNU(time, z) ;
		MCRDNU:long_name = "Downdraft unsaturated core mass flux" ;
		MCRDNU:units = "kg/m2/s" ;
		MCRDNU:missing_value = -9999.f ;
	float MCR(time, z) ;
		MCR:long_name = "Core mass flux" ;
		MCR:units = "kg/m2/s" ;
		MCR:missing_value = -9999.f ;
	float QVTEND(time, z) ;
		QVTEND:long_name = "Large-Scale Vertical Advection Moisture Tendency" ;
		QVTEND:units = "g/kg/day" ;
		QVTEND:missing_value = -9999.f ;
	float QHTEND(time, z) ;
		QHTEND:long_name = "Large-Scale Horizontal Advection Moisture Tendency" ;
		QHTEND:units = "g/kg/day" ;
		QHTEND:missing_value = -9999.f ;
	float QNUDGE(time, z) ;
		QNUDGE:long_name = "Large-Scale Moisture Nudging" ;
		QNUDGE:units = "g/kg/day" ;
		QNUDGE:missing_value = -9999.f ;
	float THTEND(time, z) ;
		THTEND:long_name = "Large-Scale Horizontal Advection Temperature Tenden-
cy" ;
		THTEND:units = "K/day" ;
		THTEND:missing_value = -9999.f ;
	float TVTEND(time, z) ;
		TVTEND:long_name = "Large-Scale Vertical Advection Temperature Tendency"
;
		TVTEND:units = "K/day" ;
		TVTEND:missing_value = -9999.f ;
	float TNUDGE(time, z) ;
		TNUDGE:long_name = "Large-Scale Temperature Nudging" ;
		TNUDGE:units = "K/day" ;
		TNUDGE:missing_value = -9999.f ;
	float WSTAR3(time, z) ;
		WSTAR3:long_name = "Value at inversion is w*^3 (conv. vel. scale)" ;
		WSTAR3:units = "m3/s3" ;
		WSTAR3:missing_value = -9999.f ;
	float UADV(time, z) ;
		UADV:long_name = "Resolved zonal momentum flux convergence" ;
		UADV:units = "m/s/day" ;
		UADV:missing_value = -9999.f ;
	float VADV(time, z) ;
		VADV:long_name = "Resolved meridioinal momentum flux convergence" ;
		VADV:units = "m/s/day" ;
		VADV:missing_value = -9999.f ;
```

```
        float UDIFF(time, z) ;
                UDIFF:long_name = "Subgrid zonal momentum flux convergence" ;
                UDIFF:units = "m/s/day" ;
                UDIFF:missing_value = -9999.f ;
        float VDIFF(time, z) ;
                VDIFF:long_name = "Subgrid meridioinal momentum flux convergence" ;
                VDIFF:units = "m/s/day" ;
                VDIFF:missing_value = -9999.f ;
        float UNUDGE(time, z) ;
                UNUDGE:long_name = "Zonal velocity nudging" ;
                UNUDGE:units = "m/s/day" ;
                UNUDGE:missing_value = -9999.f ;
        float VNUDGE(time, z) ;
                VNUDGE:long_name = "Meridional velocity nudging" ;
                VNUDGE:units = "m/s/day" ;
                VNUDGE:missing_value = -9999.f ;
        float ULSADVV(time, z) ;
                ULSADVV:long_name = "Large-scale vertical advection of Zonal velocity" ;
                ULSADVV:units = "m/s/day" ;
                ULSADVV:missing_value = -9999.f ;
        float VLSADVV(time, z) ;
                VLSADVV:long_name = "Large-scale vertical advection of meridional veloci-
ty" ;
                VLSADVV:units = "m/s/day" ;
                VLSADVV:missing_value = -9999.f ;
        float USTOR(time, z) ;
                USTOR:long_name = "Zonal velocity storage" ;
                USTOR:units = "m/s/day" ;
                USTOR:missing_value = -9999.f ;
        float VSTOR(time, z) ;
                VSTOR:long_name = "Meridional velocity storage" ;
                VSTOR:units = "m/s/day" ;
                VSTOR:missing_value = -9999.f ;
        float UTENDCOR(time, z) ;
                UTENDCOR:long_name = "Zonal velocity tendency due to Coriolis effect" ;
                UTENDCOR:units = "m/s/day" ;
                UTENDCOR:missing_value = -9999.f ;
        float URESID(time, z) ;
                URESID:long_name = "Zonal velocity budget residual" ;
                URESID:units = "m/s/day" ;
                URESID:missing_value = -9999.f ;
        float VTENDCOR(time, z) ;
                VTENDCOR:long_name = "Meridional velocity budget residual" ;
                VTENDCOR:units = "m/s/day" ;
                VTENDCOR:missing_value = -9999.f ;
        float VRESID(time, z) ;
                VRESID:long_name = "Meridional velocity budget residual" ;
                VRESID:units = "m/s/day" ;
                VRESID:missing_value = -9999.f ;
        float HLSTOR(time, z) ;
                HLSTOR:long_name = "Liquid-ice static energy storage" ;
                HLSTOR:units = "K/day" ;
                HLSTOR:missing_value = -9999.f ;
        float QTSTOR(time, z) ;
                QTSTOR:long_name = "Total water storage" ;
                QTSTOR:units = "K/day" ;
                QTSTOR:missing_value = -9999.f ;
        float CLD(time, z) ;
                CLD:long_name = "cloud Fraction" ;
                CLD:units = "" ;
```

```
        CLD:missing_value = -9999.f ;
float WCLD(time, z) ;
        WCLD:long_name = “Mean W in cloud” ;
        WCLD:units = “m/s” ;
        WCLD:missing_value = -9999.f ;
float UCLD(time, z) ;
        UCLD:long_name = “Mean U in cloud” ;
        UCLD:units = “m/s” ;
        UCLD:missing_value = -9999.f ;
float VCLD(time, z) ;
        VCLD:long_name = “Mean V in cloud” ;
        VCLD:units = “m/s” ;
        VCLD:missing_value = -9999.f ;
float MSECLD(time, z) ;
        MSECLD:long_name = “Mean moist static energy in cloud” ;
        MSECLD:units = “K” ;
        MSECLD:missing_value = -9999.f ;
float DSECLD(time, z) ;
        DSECLD:long_name = “Mean dry static energy in cloud” ;
        DSECLD:units = “K” ;
        DSECLD:missing_value = -9999.f ;
float TLCLD(time, z) ;
        TLCLD:long_name = “Mean liquid-ice static energy in cloud” ;
        TLCLD:units = “K” ;
        TLCLD:missing_value = -9999.f ;
float TACLD(time, z) ;
        TACLD:long_name = “Mean TABS in cloud” ;
        TACLD:units = “K” ;
        TACLD:missing_value = -9999.f ;
float TVCLD(time, z) ;
        TVCLD:long_name = “Mean THETAV in cloud” ;
        TVCLD:units = “K” ;
        TVCLD:missing_value = -9999.f ;
float TVCLDA(time, z) ;
        TVCLDA:long_name = “Mean THETAV anomaly in cloud” ;
        TVCLDA:units = “K” ;
        TVCLDA:missing_value = -9999.f ;
float QTCLD(time, z) ;
        QTCLD:long_name = “Mean QT in cloud” ;
        QTCLD:units = “g/kg” ;
        QTCLD:missing_value = -9999.f ;
float QNCLD(time, z) ;
        QNCLD:long_name = “Mean QN in cloud” ;
        QNCLD:units = “g/kg” ;
        QNCLD:missing_value = -9999.f ;
float QPCLD(time, z) ;
        QPCLD:long_name = “Mean QP in cloud” ;
        QPCLD:units = “g/kg” ;
        QPCLD:missing_value = -9999.f ;
float WCLDA(time, z) ;
        WCLDA:long_name = “W in cloud averaged over the whole domain” ;
        WCLDA:units = “m/s” ;
        WCLDA:missing_value = -9999.f ;
float TLWCLD(time, z) ;
        TLWCLD:long_name = “TLW in cloud averaged over the whole domain” ;
        TLWCLD:units = “Km/s” ;
        TLWCLD:missing_value = -9999.f ;
float TVWCLD(time, z) ;
        TVWCLD:long_name = “TVW in cloud averaged over the whole domain” ;
        TVWCLD:units = “Km/s” ;
```

```
        TVWCLD:missing_value = -9999.f ;
float QTWCLD(time, z) ;
        QTWCLD:long_name = “QTW in cloud averaged over the whole domain” ;
        QTWCLD:units = “g/kg m/s” ;
        QTWCLD:missing_value = -9999.f ;
float QCWCLD(time, z) ;
        QCWCLD:long_name = “QCW in cloud averaged over the whole domain” ;
        QCWCLD:units = “g/kg m/s” ;
        QCWCLD:missing_value = -9999.f ;
float QIWCLD(time, z) ;
        QIWCLD:long_name = “QIW in cloud averaged over the whole domain” ;
        QIWCLD:units = “g/kg m/s” ;
        QIWCLD:missing_value = -9999.f ;
float HFCLD(time, z) ;
        HFCLD:long_name = “Mean Frozen MSE in cloud” ;
        HFCLD:units = “K” ;
        HFCLD:missing_value = -9999.f ;
float HFCLDA(time, z) ;
        HFCLDA:long_name = “Mean Frozen MSE anomaly in cloud” ;
        HFCLDA:units = “K” ;
        HFCLDA:missing_value = -9999.f ;
float UCLDA(time, z) ;
        UCLDA:long_name = “Mean U anomaly in cloud” ;
        UCLDA:units = “m/s” ;
        UCLDA:missing_value = -9999.f ;
float VCLDA(time, z) ;
        VCLDA:long_name = “Mean V anomaly in cloud” ;
        VCLDA:units = “m/s” ;
        VCLDA:missing_value = -9999.f ;
float UPGFCLD(time, z) ;
        UPGFCLD:long_name = “Zonal pressure gradient in cloud” ;
        UPGFCLD:units = “m/s2” ;
        UPGFCLD:missing_value = -9999.f ;
float VPGFCLD(time, z) ;
        VPGFCLD:long_name = “Meridional pressure gradient in cloud” ;
        VPGFCLD:units = “m/s2” ;
        VPGFCLD:missing_value = -9999.f ;
float WPGFCLD(time, z) ;
        WPGFCLD:long_name = “Vertical pressure gradient in cloud” ;
        WPGFCLD:units = “m/s2” ;
        WPGFCLD:missing_value = -9999.f ;
float UWCLD(time, z) ;
        UWCLD:long_name = “UW in cloud” ;
        UWCLD:units = “m2/s2” ;
        UWCLD:missing_value = -9999.f ;
float VWCLD(time, z) ;
        VWCLD:long_name = “VW in cloud” ;
        VWCLD:units = “m2/s2” ;
        VWCLD:missing_value = -9999.f ;
float UWSBCLD(time, z) ;
        UWSBCLD:long_name = “Subgrid UW in cloud” ;
        UWSBCLD:units = “m2/s2” ;
        UWSBCLD:missing_value = -9999.f ;
float VWSBCLD(time, z) ;
        VWSBCLD:long_name = “Subgrid VW in cloud” ;
        VWSBCLD:units = “m2/s2” ;
        VWSBCLD:missing_value = -9999.f ;
```

```
      float MFCLD(time, z) ;
             MFCLD:long_name = "Mass flux in cloud averaged over the whole domain" ;
             MFCLD:units = "kg/m2/s" ;
             MFCLD:missing_value = -9999.f ;
      float MFHCLD(time, z) ;
             MFHCLD:long_name = "RHO*W*HF in cloud averaged over the whole domain" ;
             MFHCLD:units = "K kg/m2/s" ;
             MFHCLD:missing_value = -9999.f ;
      float MFHCLDA(time, z) ;
             MFHCLDA:long_name = "RHO*W*HF anomaly in cloud averaged over the whole
domain" ;
             MFHCLDA:units = "K kg/m2/s" ;
             MFHCLDA:missing_value = -9999.f ;
      float MFTLCLD(time, z) ;
             MFTLCLD:long_name = "RHO*W*TL in cloud averaged over the whole domain" ;
             MFTLCLD:units = "K kg/m2/s" ;
             MFTLCLD:missing_value = -9999.f ;
      float MFTLCLDA(time, z) ;
             MFTLCLDA:long_name = "RHO*W*TL anomaly in cloud averaged over the whole
domain" ;
             MFTLCLDA:units = "K kg/m2/s" ;
             MFTLCLDA:missing_value = -9999.f ;
      float MFTVCLD(time, z) ;
             MFTVCLD:long_name = "RHO*W*TV in cloud averaged over the whole domain" ;
             MFTVCLD:units = "K kg/m2/s" ;
             MFTVCLD:missing_value = -9999.f ;
      float MFTVCLDA(time, z) ;
             MFTVCLDA:long_name = "RHO*W*TV anomaly in cloud averaged over the whole
domain" ;
             MFTVCLDA:units = "K kg/m2/s" ;
             MFTVCLDA:missing_value = -9999.f ;
      float MFQTCLD(time, z) ;
             MFQTCLD:long_name = "RHO*W*QT in cloud averaged over the whole domain" ;
             MFQTCLD:units = "g/m2/s" ;
             MFQTCLD:missing_value = -9999.f ;
      float MFQTCLDA(time, z) ;
             MFQTCLDA:long_name = "RHO*W*QT anomaly in cloud averaged over the whole
domain" ;
             MFQTCLDA:units = "g/m2/s" ;
             MFQTCLDA:missing_value = -9999.f ;
      float RUWCLD(time, z) ;
             RUWCLD:long_name = "RHOUW in cloud averaged over the whole domain" ;
             RUWCLD:units = "kg/m/s2" ;
             RUWCLD:missing_value = -9999.f ;
      float RVWCLD(time, z) ;
             RVWCLD:long_name = "RHOVW in cloud averaged over the whole domain" ;
             RVWCLD:units = "kg/m/s2" ;
             RVWCLD:missing_value = -9999.f ;
      float RWWCLD(time, z) ;
             RWWCLD:long_name = "RHOWW in cloud averaged over the whole domain" ;
             RWWCLD:units = "kg/m/s2" ;
             RWWCLD:missing_value = -9999.f ;
      float COR(time, z) ;
             COR:long_name = "core Fraction" ;
             COR:units = "" ;
             COR:missing_value = -9999.f ;
      float WCOR(time, z) ;
             WCOR:long_name = "Mean W in core" ;
             WCOR:units = "m/s" ;
             WCOR:missing_value = -9999.f ;
```

```
float UCOR(time, z) ;
       UCOR:long_name = "Mean U in core" ;
       UCOR:units = "m/s" ;
       UCOR:missing_value = -9999.f ;
float VCOR(time, z) ;
       VCOR:long_name = "Mean V in core" ;
       VCOR:units = "m/s" ;
       VCOR:missing_value = -9999.f ;
float MSECOR(time, z) ;
       MSECOR:long_name = "Mean moist static energy in core" ;
       MSECOR:units = "K" ;
       MSECOR:missing_value = -9999.f ;
float DSECOR(time, z) ;
       DSECOR:long_name = "Mean dry static energy in core" ;
       DSECOR:units = "K" ;
       DSECOR:missing_value = -9999.f ;
float TLCOR(time, z) ;
       TLCOR:long_name = "Mean liquid-ice static energy in core" ;
       TLCOR:units = "K" ;
       TLCOR:missing_value = -9999.f ;
float TACOR(time, z) ;
       TACOR:long_name = "Mean TABS in core" ;
       TACOR:units = "K" ;
       TACOR:missing_value = -9999.f ;
float TVCOR(time, z) ;
       TVCOR:long_name = "Mean THETAV in core" ;
       TVCOR:units = "K" ;
       TVCOR:missing_value = -9999.f ;
float TVCORA(time, z) ;
       TVCORA:long_name = "Mean THETAV anomaly in core" ;
       TVCORA:units = "K" ;
       TVCORA:missing_value = -9999.f ;
float QTCOR(time, z) ;
       QTCOR:long_name = "Mean QT in core" ;
       QTCOR:units = "g/kg" ;
       QTCOR:missing_value = -9999.f ;
float QNCOR(time, z) ;
       QNCOR:long_name = "Mean QN in core" ;
       QNCOR:units = "g/kg" ;
       QNCOR:missing_value = -9999.f ;
float QPCOR(time, z) ;
       QPCOR:long_name = "Mean QP in core" ;
       QPCOR:units = "g/kg" ;
       QPCOR:missing_value = -9999.f ;
float WCORA(time, z) ;
       WCORA:long_name = "W in core averaged over the whole domain" ;
       WCORA:units = "m/s" ;
       WCORA:missing_value = -9999.f ;
float TLWCOR(time, z) ;
       TLWCOR:long_name = "TLW in core averaged over the whole domain" ;
       TLWCOR:units = "Km/s" ;
       TLWCOR:missing_value = -9999.f ;
float TVWCOR(time, z) ;
       TVWCOR:long_name = "TVW in core averaged over the whole domain" ;
       TVWCOR:units = "Km/s" ;
       TVWCOR:missing_value = -9999.f ;
float QTWCOR(time, z) ;
       QTWCOR:long_name = "QTW in core averaged over the whole domain" ;
       QTWCOR:units = "g/kg m/s" ;
       QTWCOR:missing_value = -9999.f ;
```

```
float QCWCOR(time, z) ;
       QCWCOR:long_name = "QCW in core averaged over the whole domain" ;
       QCWCOR:units = "g/kg m/s" ;
       QCWCOR:missing_value = -9999.f ;
float QIWCOR(time, z) ;
       QIWCOR:long_name = "QIW in core averaged over the whole domain" ;
       QIWCOR:units = "g/kg m/s" ;
       QIWCOR:missing_value = -9999.f ;
float HFCOR(time, z) ;
       HFCOR:long_name = "Mean Frozen MSE in core" ;
       HFCOR:units = "K" ;
       HFCOR:missing_value = -9999.f ;
float HFCORA(time, z) ;
       HFCORA:long_name = "Mean Frozen MSE anomaly in core" ;
       HFCORA:units = "K" ;
       HFCORA:missing_value = -9999.f ;
float UCORA(time, z) ;
       UCORA:long_name = "Mean U anomaly in core" ;
       UCORA:units = "m/s" ;
       UCORA:missing_value = -9999.f ;
float VCORA(time, z) ;
       VCORA:long_name = "Mean V anomaly in core" ;
       VCORA:units = "m/s" ;
       VCORA:missing_value = -9999.f ;
float UPGFCOR(time, z) ;
       UPGFCOR:long_name = "Zonal pressure gradient in core" ;
       UPGFCOR:units = "m/s2" ;
       UPGFCOR:missing_value = -9999.f ;
float VPGFCOR(time, z) ;
       VPGFCOR:long_name = "Meridional pressure gradient in core" ;
       VPGFCOR:units = "m/s2" ;
       VPGFCOR:missing_value = -9999.f ;
float WPGFCOR(time, z) ;
       WPGFCOR:long_name = "Vertical pressure gradient in core" ;
       WPGFCOR:units = "m/s2" ;
       WPGFCOR:missing_value = -9999.f ;
float UWCOR(time, z) ;
       UWCOR:long_name = "UW in core" ;
       UWCOR:units = "m2/s2" ;
       UWCOR:missing_value = -9999.f ;
float VWCOR(time, z) ;
       VWCOR:long_name = "VW in core" ;
       VWCOR:units = "m2/s2" ;
       VWCOR:missing_value = -9999.f ;
float UWSBCOR(time, z) ;
       UWSBCOR:long_name = "Subgrid UW in core" ;
       UWSBCOR:units = "m2/s2" ;
       UWSBCOR:missing_value = -9999.f ;
float VWSBCOR(time, z) ;
       VWSBCOR:long_name = "Subgrid VW in core" ;
       VWSBCOR:units = "m2/s2" ;
       VWSBCOR:missing_value = -9999.f ;
float MFCOR(time, z) ;
       MFCOR:long_name = "Mass flux in core averaged over the whole domain" ;
       MFCOR:units = "kg/m2/s" ;
       MFCOR:missing_value = -9999.f ;
float MFHCOR(time, z) ;
       MFHCOR:long_name = "RHO*W*HF in core averaged over the whole domain" ;
       MFHCOR:units = "K kg/m2/s" ;
       MFHCOR:missing_value = -9999.f ;
```

```
	float MFHCORA(time, z) ;
		MFHCORA:long_name = "RHO*W*HF anomaly in core averaged over the whole do-
main" ;
		MFHCORA:units = "K kg/m2/s" ;
		MFHCORA:missing_value = -9999.f ;
	float MFTLCOR(time, z) ;
		MFTLCOR:long_name = "RHO*W*TL in core averaged over the whole domain" ;
		MFTLCOR:units = "K kg/m2/s" ;
		MFTLCOR:missing_value = -9999.f ;
	float MFTLCORA(time, z) ;
		MFTLCORA:long_name = "RHO*W*TL anomaly in core averaged over the whole
domain" ;
		MFTLCORA:units = "K kg/m2/s" ;
		MFTLCORA:missing_value = -9999.f ;
	float MFTVCOR(time, z) ;
		MFTVCOR:long_name = "RHO*W*TV in core averaged over the whole domain" ;
		MFTVCOR:units = "K kg/m2/s" ;
		MFTVCOR:missing_value = -9999.f ;
	float MFTVCORA(time, z) ;
		MFTVCORA:long_name = "RHO*W*TV anomaly in core averaged over the whole
domain" ;
		MFTVCORA:units = "K kg/m2/s" ;
		MFTVCORA:missing_value = -9999.f ;
	float MFQTCOR(time, z) ;
		MFQTCOR:long_name = "RHO*W*QT in core averaged over the whole domain" ;
		MFQTCOR:units = "g/m2/s" ;
		MFQTCOR:missing_value = -9999.f ;
	float MFQTCORA(time, z) ;
		MFQTCORA:long_name = "RHO*W*QT anomaly in core averaged over the whole
domain" ;
		MFQTCORA:units = "g/m2/s" ;
		MFQTCORA:missing_value = -9999.f ;
	float RUWCOR(time, z) ;
		RUWCOR:long_name = "RHOUW in core averaged over the whole domain" ;
		RUWCOR:units = "kg/m/s2" ;
		RUWCOR:missing_value = -9999.f ;
	float RVWCOR(time, z) ;
		RVWCOR:long_name = "RHOVW in core averaged over the whole domain" ;
		RVWCOR:units = "kg/m/s2" ;
		RVWCOR:missing_value = -9999.f ;
	float RWWCOR(time, z) ;
		RWWCOR:long_name = "RHOWW in core averaged over the whole domain" ;
		RWWCOR:units = "kg/m/s2" ;
		RWWCOR:missing_value = -9999.f ;
	float CDN(time, z) ;
		CDN:long_name = "downdraft core Fraction" ;
		CDN:units = "" ;
		CDN:missing_value = -9999.f ;
	float WCDN(time, z) ;
		WCDN:long_name = "Mean W in downdraft core" ;
		WCDN:units = "m/s" ;
		WCDN:missing_value = -9999.f ;
	float UCDN(time, z) ;
		UCDN:long_name = "Mean U in downdraft core" ;
		UCDN:units = "m/s" ;
		UCDN:missing_value = -9999.f ;
	float VCDN(time, z) ;
		VCDN:long_name = "Mean V in downdraft core" ;
		VCDN:units = "m/s" ;
		VCDN:missing_value = -9999.f ;
```

```
float MSECDN(time, z) ;
       MSECDN:long_name = "Mean moist static energy in downdraft core" ;
       MSECDN:units = "K" ;
       MSECDN:missing_value = -9999.f ;
float DSECDN(time, z) ;
       DSECDN:long_name = "Mean dry static energy in downdraft core" ;
       DSECDN:units = "K" ;
       DSECDN:missing_value = -9999.f ;
float TLCDN(time, z) ;
       TLCDN:long_name = "Mean liquid-ice static energy in downdraft core" ;
       TLCDN:units = "K" ;
       TLCDN:missing_value = -9999.f ;
float TACDN(time, z) ;
       TACDN:long_name = "Mean TABS in downdraft core" ;
       TACDN:units = "K" ;
       TACDN:missing_value = -9999.f ;
float TVCDN(time, z) ;
       TVCDN:long_name = "Mean THETAV in downdraft core" ;
       TVCDN:units = "K" ;
       TVCDN:missing_value = -9999.f ;
float TVCDNA(time, z) ;
       TVCDNA:long_name = "Mean THETAV anomaly in downdraft core" ;
       TVCDNA:units = "K" ;
       TVCDNA:missing_value = -9999.f ;
float QTCDN(time, z) ;
       QTCDN:long_name = "Mean QT in downdraft core" ;
       QTCDN:units = "g/kg" ;
       QTCDN:missing_value = -9999.f ;
float QNCDN(time, z) ;
       QNCDN:long_name = "Mean QN in downdraft core" ;
       QNCDN:units = "g/kg" ;
       QNCDN:missing_value = -9999.f ;
float QPCDN(time, z) ;
       QPCDN:long_name = "Mean QP in downdraft core" ;
       QPCDN:units = "g/kg" ;
       QPCDN:missing_value = -9999.f ;
float WCDNA(time, z) ;
       WCDNA:long_name = "W in downdraft core averaged over the whole domain" ;
       WCDNA:units = "m/s" ;
       WCDNA:missing_value = -9999.f ;
float TLWCDN(time, z) ;
       TLWCDN:long_name = "TLW in downdraft core averaged over the whole domain" ;
       TLWCDN:units = "Km/s" ;
       TLWCDN:missing_value = -9999.f ;
float TVWCDN(time, z) ;
       TVWCDN:long_name = "TVW in downdraft core averaged over the whole domain" ;
       TVWCDN:units = "Km/s" ;
       TVWCDN:missing_value = -9999.f ;
float QTWCDN(time, z) ;
       QTWCDN:long_name = "QTW in downdraft core averaged over the whole domain" ;
       QTWCDN:units = "g/kg m/s" ;
       QTWCDN:missing_value = -9999.f ;
float QCWCDN(time, z) ;
       QCWCDN:long_name = "QCW in downdraft core averaged over the whole domain" ;
       QCWCDN:units = "g/kg m/s" ;
       QCWCDN:missing_value = -9999.f ;
float QIWCDN(time, z) ;
       QIWCDN:long_name = "QIW in downdraft core averaged over the whole domain" ;
       QIWCDN:units = "g/kg m/s" ;
       QIWCDN:missing_value = -9999.f ;
```

```
        float HFCDN(time, z) ;
                HFCDN:long_name = "Mean Frozen MSE in downdraft core" ;
                HFCDN:units = "K" ;
                HFCDN:missing_value = -9999.f ;
        float HFCDNA(time, z) ;
                HFCDNA:long_name = "Mean Frozen MSE anomaly in downdraft core" ;
                HFCDNA:units = "K" ;
                HFCDNA:missing_value = -9999.f ;
        float UCDNA(time, z) ;
                UCDNA:long_name = "Mean U anomaly in downdraft core" ;
                UCDNA:units = "m/s" ;
                UCDNA:missing_value = -9999.f ;
        float VCDNA(time, z) ;
                VCDNA:long_name = "Mean V anomaly in downdraft core" ;
                VCDNA:units = "m/s" ;
                VCDNA:missing_value = -9999.f ;
        float UPGFCDN(time, z) ;
                UPGFCDN:long_name = "Zonal pressure gradient in downdraft core" ;
                UPGFCDN:units = "m/s2" ;
                UPGFCDN:missing_value = -9999.f ;
        float VPGFCDN(time, z) ;
                VPGFCDN:long_name = "Meridional pressure gradient in downdraft core" ;
                VPGFCDN:units = "m/s2" ;
                VPGFCDN:missing_value = -9999.f ;
        float WPGFCDN(time, z) ;
                WPGFCDN:long_name = "Vertical pressure gradient in downdraft core" ;
                WPGFCDN:units = "m/s2" ;
                WPGFCDN:missing_value = -9999.f ;
        float UWCDN(time, z) ;
                UWCDN:long_name = "UW in downdraft core" ;
                UWCDN:units = "m2/s2" ;
                UWCDN:missing_value = -9999.f ;
        float VWCDN(time, z) ;
                VWCDN:long_name = "VW in downdraft core" ;
                VWCDN:units = "m2/s2" ;
                VWCDN:missing_value = -9999.f ;
        float UWSBCDN(time, z) ;
                UWSBCDN:long_name = "Subgrid UW in downdraft core" ;
                UWSBCDN:units = "m2/s2" ;
                UWSBCDN:missing_value = -9999.f ;
        float VWSBCDN(time, z) ;
                VWSBCDN:long_name = "Subgrid VW in downdraft core" ;
                VWSBCDN:units = "m2/s2" ;
                VWSBCDN:missing_value = -9999.f ;
        float MFCDN(time, z) ;
                MFCDN:long_name = "Mass flux in downdraft core averaged over the whole do-
main" ;
                MFCDN:units = "kg/m2/s" ;
                MFCDN:missing_value = -9999.f ;
        float MFHCDN(time, z) ;
                MFHCDN:long_name = "RHO*W*HF in downdraft core averaged over the whole
domain" ;
                MFHCDN:units = "K kg/m2/s" ;
                MFHCDN:missing_value = -9999.f ;
        float MFHCDNA(time, z) ;
                MFHCDNA:long_name = "RHO*W*HF anomaly in downdraft core averaged over the
whole domain" ;
                MFHCDNA:units = "K kg/m2/s" ;
                MFHCDNA:missing_value = -9999.f ;
```

```
        float MFTLCDN(time, z) ;
                MFTLCDN:long_name = "RHO*W*TL in downdraft core averaged over the whole
domain" ;
                MFTLCDN:units = "K kg/m2/s" ;
                MFTLCDN:missing_value = -9999.f ;
        float MFTLCDNA(time, z) ;
                MFTLCDNA:long_name = "RHO*W*TL anomaly in downdraft core averaged over
the whole domain" ;
                MFTLCDNA:units = "K kg/m2/s" ;
                MFTLCDNA:missing_value = -9999.f ;
        float MFTVCDN(time, z) ;
                MFTVCDN:long_name = "RHO*W*TV in downdraft core averaged over the whole
domain" ;
                MFTVCDN:units = "K kg/m2/s" ;
                MFTVCDN:missing_value = -9999.f ;
        float MFTVCDNA(time, z) ;
                MFTVCDNA:long_name = "RHO*W*TV anomaly in downdraft core averaged over
the whole domain" ;
                MFTVCDNA:units = "K kg/m2/s" ;
                MFTVCDNA:missing_value = -9999.f ;
        float MFQTCDN(time, z) ;
                MFQTCDN:long_name = "RHO*W*QT in downdraft core averaged over the whole
domain" ;
                MFQTCDN:units = "g/m2/s" ;
                MFQTCDN:missing_value = -9999.f ;
        float MFQTCDNA(time, z) ;
                MFQTCDNA:long_name = "RHO*W*QT anomaly in downdraft core averaged over
the whole domain" ;
                MFQTCDNA:units = "g/m2/s" ;
                MFQTCDNA:missing_value = -9999.f ;
        float RUWCDN(time, z) ;
                RUWCDN:long_name = "RHOUW in downdraft core averaged over the whole do-
main" ;
                RUWCDN:units = "kg/m/s2" ;
                RUWCDN:missing_value = -9999.f ;
        float RVWCDN(time, z) ;
                RVWCDN:long_name = "RHOVW in downdraft core averaged over the whole do-
main" ;
                RVWCDN:units = "kg/m/s2" ;
                RVWCDN:missing_value = -9999.f ;
        float RWWCDN(time, z) ;
                RWWCDN:long_name = "RHOWW in downdraft core averaged over the whole do-
main" ;
                RWWCDN:units = "kg/m/s2" ;
                RWWCDN:missing_value = -9999.f ;
        float SUP(time, z) ;
                SUP:long_name = "saturated updrafts Fraction" ;
                SUP:units = "" ;
                SUP:missing_value = -9999.f ;
        float WSUP(time, z) ;
                WSUP:long_name = "Mean W in saturated updrafts" ;
                WSUP:units = "m/s" ;
                WSUP:missing_value = -9999.f ;
        float USUP(time, z) ;
                USUP:long_name = "Mean U in saturated updrafts" ;
                USUP:units = "m/s" ;
                USUP:missing_value = -9999.f ;
```

```
	float VSUP(time, z) ;
		VSUP:long_name = "Mean V in saturated updrafts" ;
		VSUP:units = "m/s" ;
		VSUP:missing_value = -9999.f ;
	float MSESUP(time, z) ;
		MSESUP:long_name = "Mean moist static energy in saturated updrafts" ;
		MSESUP:units = "K" ;
		MSESUP:missing_value = -9999.f ;
	float DSESUP(time, z) ;
		DSESUP:long_name = "Mean dry static energy in saturated updrafts" ;
		DSESUP:units = "K" ;
		DSESUP:missing_value = -9999.f ;
	float TLSUP(time, z) ;
		TLSUP:long_name = "Mean liquid-ice static energy in saturated updrafts" ;
		TLSUP:units = "K" ;
		TLSUP:missing_value = -9999.f ;
	float TASUP(time, z) ;
		TASUP:long_name = "Mean TABS in saturated updrafts" ;
		TASUP:units = "K" ;
		TASUP:missing_value = -9999.f ;
	float TVSUP(time, z) ;
		TVSUP:long_name = "Mean THETAV in saturated updrafts" ;
		TVSUP:units = "K" ;
		TVSUP:missing_value = -9999.f ;
	float TVSUPA(time, z) ;
		TVSUPA:long_name = "Mean THETAV anomaly in saturated updrafts" ;
		TVSUPA:units = "K" ;
		TVSUPA:missing_value = -9999.f ;
	float QTSUP(time, z) ;
		QTSUP:long_name = "Mean QT in saturated updrafts" ;
		QTSUP:units = "g/kg" ;
		QTSUP:missing_value = -9999.f ;
	float QNSUP(time, z) ;
		QNSUP:long_name = "Mean QN in saturated updrafts" ;
		QNSUP:units = "g/kg" ;
		QNSUP:missing_value = -9999.f ;
	float QPSUP(time, z) ;
		QPSUP:long_name = "Mean QP in saturated updrafts" ;
		QPSUP:units = "g/kg" ;
		QPSUP:missing_value = -9999.f ;
	float WSUPA(time, z) ;
		WSUPA:long_name = "W in saturated updrafts averaged over the whole do-
main" ;
		WSUPA:units = "m/s" ;
		WSUPA:missing_value = -9999.f ;
	float TLWSUP(time, z) ;
		TLWSUP:long_name = "TLW in saturated updrafts averaged over the whole do-
main" ;
		TLWSUP:units = "Km/s" ;
		TLWSUP:missing_value = -9999.f ;
	float TVWSUP(time, z) ;
		TVWSUP:long_name = "TVW in saturated updrafts averaged over the whole do-
main" ;
		TVWSUP:units = "Km/s" ;
		TVWSUP:missing_value = -9999.f ;
	float QTWSUP(time, z) ;
		QTWSUP:long_name = "QTW in saturated updrafts averaged over the whole do-
main" ;
		QTWSUP:units = "g/kg m/s" ;
		QTWSUP:missing_value = -9999.f ;
```

```
	float QCWSUP(time, z) ;
		QCWSUP:long_name = "QCW in saturated updrafts averaged over the whole do-
main" ;
		QCWSUP:units = "g/kg m/s" ;
		QCWSUP:missing_value = -9999.f ;
	float QIWSUP(time, z) ;
		QIWSUP:long_name = "QIW in saturated updrafts averaged over the whole do-
main" ;
		QIWSUP:units = "g/kg m/s" ;
		QIWSUP:missing_value = -9999.f ;
	float HFSUP(time, z) ;
		HFSUP:long_name = "Mean Frozen MSE in saturated updrafts" ;
		HFSUP:units = "K" ;
		HFSUP:missing_value = -9999.f ;
	float HFSUPA(time, z) ;
		HFSUPA:long_name = "Mean Frozen MSE anomaly in saturated updrafts" ;
		HFSUPA:units = "K" ;
		HFSUPA:missing_value = -9999.f ;
	float USUPA(time, z) ;
		USUPA:long_name = "Mean U anomaly in saturated updrafts" ;
		USUPA:units = "m/s" ;
		USUPA:missing_value = -9999.f ;
	float VSUPA(time, z) ;
		VSUPA:long_name = "Mean V anomaly in saturated updrafts" ;
		VSUPA:units = "m/s" ;
		VSUPA:missing_value = -9999.f ;
	float UPGFSUP(time, z) ;
		UPGFSUP:long_name = "Zonal pressure gradient in saturated updrafts" ;
		UPGFSUP:units = "m/s2" ;
		UPGFSUP:missing_value = -9999.f ;
	float VPGFSUP(time, z) ;
		VPGFSUP:long_name = "Meridional pressure gradient in saturated updrafts" ;
		VPGFSUP:units = "m/s2" ;
		VPGFSUP:missing_value = -9999.f ;
	float WPGFSUP(time, z) ;
		WPGFSUP:long_name = "Vertical pressure gradient in saturated updrafts" ;
		WPGFSUP:units = "m/s2" ;
		WPGFSUP:missing_value = -9999.f ;
	float UWSUP(time, z) ;
		UWSUP:long_name = "UW in saturated updrafts" ;
		UWSUP:units = "m2/s2" ;
		UWSUP:missing_value = -9999.f ;
	float VWSUP(time, z) ;
		VWSUP:long_name = "VW in saturated updrafts" ;
		VWSUP:units = "m2/s2" ;
		VWSUP:missing_value = -9999.f ;
	float UWSBSUP(time, z) ;
		UWSBSUP:long_name = "Subgrid UW in saturated updrafts" ;
		UWSBSUP:units = "m2/s2" ;
		UWSBSUP:missing_value = -9999.f ;
	float VWSBSUP(time, z) ;
		VWSBSUP:long_name = "Subgrid VW in saturated updrafts" ;
		VWSBSUP:units = "m2/s2" ;
		VWSBSUP:missing_value = -9999.f ;
	float MFSUP(time, z) ;
		MFSUP:long_name = "Mass flux in saturated updrafts averaged over the whole
domain" ;
		MFSUP:units = "kg/m2/s" ;
		MFSUP:missing_value = -9999.f ;
```

```
        float MFHSUP(time, z) ;
              MFHSUP:long_name = "RHO*W*HF in saturated updrafts averaged over the
whole domain" ;
              MFHSUP:units = "K kg/m2/s" ;
              MFHSUP:missing_value = -9999.f ;
        float MFHSUPA(time, z) ;
              MFHSUPA:long_name = "RHO*W*HF anomaly in saturated updrafts averaged over
the whole domain" ;
              MFHSUPA:units = "K kg/m2/s" ;
              MFHSUPA:missing_value = -9999.f ;
        float MFTLSUP(time, z) ;
              MFTLSUP:long_name = "RHO*W*TL in saturated updrafts averaged over the
whole domain" ;
              MFTLSUP:units = "K kg/m2/s" ;
              MFTLSUP:missing_value = -9999.f ;
        float MFTLSUPA(time, z) ;
              MFTLSUPA:long_name = "RHO*W*TL anomaly in saturated updrafts averaged
over the whole domain" ;
              MFTLSUPA:units = "K kg/m2/s" ;
              MFTLSUPA:missing_value = -9999.f ;
        float MFTVSUP(time, z) ;
              MFTVSUP:long_name = "RHO*W*TV in saturated updrafts averaged over the
whole domain" ;
              MFTVSUP:units = "K kg/m2/s" ;
              MFTVSUP:missing_value = -9999.f ;
        float MFTVSUPA(time, z) ;
              MFTVSUPA:long_name = "RHO*W*TV anomaly in saturated updrafts averaged
over the whole domain" ;
              MFTVSUPA:units = "K kg/m2/s" ;
              MFTVSUPA:missing_value = -9999.f ;
        float MFQTSUP(time, z) ;
              MFQTSUP:long_name = "RHO*W*QT in saturated updrafts averaged over the
whole domain" ;
              MFQTSUP:units = "g/m2/s" ;
              MFQTSUP:missing_value = -9999.f ;
        float MFQTSUPA(time, z) ;
              MFQTSUPA:long_name = "RHO*W*QT anomaly in saturated updrafts averaged
over the whole domain" ;
              MFQTSUPA:units = "g/m2/s" ;
              MFQTSUPA:missing_value = -9999.f ;
        float RUWSUP(time, z) ;
              RUWSUP:long_name = "RHOUW in saturated updrafts averaged over the whole
domain" ;
              RUWSUP:units = "kg/m/s2" ;
              RUWSUP:missing_value = -9999.f ;
        float RVWSUP(time, z) ;
              RVWSUP:long_name = "RHOVW in saturated updrafts averaged over the whole
domain" ;
              RVWSUP:units = "kg/m/s2" ;
              RVWSUP:missing_value = -9999.f ;
        float RWWSUP(time, z) ;
              RWWSUP:long_name = "RHOWW in saturated updrafts averaged over the whole
domain" ;
              RWWSUP:units = "kg/m/s2" ;
              RWWSUP:missing_value = -9999.f ;
        float SDN(time, z) ;
              SDN:long_name = "saturated downdrafts Fraction" ;
              SDN:units = "" ;
              SDN:missing_value = -9999.f ;
```

```
        float WSDN(time, z) ;
                WSDN:long_name = "Mean W in saturated downdrafts" ;
                WSDN:units = "m/s" ;
                WSDN:missing_value = -9999.f ;
        float USDN(time, z) ;
                USDN:long_name = "Mean U in saturated downdrafts" ;
                USDN:units = "m/s" ;
                USDN:missing_value = -9999.f ;
        float VSDN(time, z) ;
                VSDN:long_name = "Mean V in saturated downdrafts" ;
                VSDN:units = "m/s" ;
                VSDN:missing_value = -9999.f ;
        float MSESDN(time, z) ;
                MSESDN:long_name = "Mean moist static energy in saturated downdrafts" ;
                MSESDN:units = "K" ;
                MSESDN:missing_value = -9999.f ;
        float DSESDN(time, z) ;
                DSESDN:long_name = "Mean dry static energy in saturated downdrafts" ;
                DSESDN:units = "K" ;
                DSESDN:missing_value = -9999.f ;
        float TLSDN(time, z) ;
                TLSDN:long_name = "Mean liquid-ice static energy in saturated downdrafts" ;
                TLSDN:units = "K" ;
                TLSDN:missing_value = -9999.f ;
        float TASDN(time, z) ;
                TASDN:long_name = "Mean TABS in saturated downdrafts" ;
                TASDN:units = "K" ;
                TASDN:missing_value = -9999.f ;
        float TVSDN(time, z) ;
                TVSDN:long_name = "Mean THETAV in saturated downdrafts" ;
                TVSDN:units = "K" ;
                TVSDN:missing_value = -9999.f ;
        float TVSDNA(time, z) ;
                TVSDNA:long_name = "Mean THETAV anomaly in saturated downdrafts" ;
                TVSDNA:units = "K" ;
                TVSDNA:missing_value = -9999.f ;
        float QTSDN(time, z) ;
                QTSDN:long_name = "Mean QT in saturated downdrafts" ;
                QTSDN:units = "g/kg" ;
                QTSDN:missing_value = -9999.f ;
        float QNSDN(time, z) ;
                QNSDN:long_name = "Mean QN in saturated downdrafts" ;
                QNSDN:units = "g/kg" ;
                QNSDN:missing_value = -9999.f ;
        float QPSDN(time, z) ;
                QPSDN:long_name = "Mean QP in saturated downdrafts" ;
                QPSDN:units = "g/kg" ;
                QPSDN:missing_value = -9999.f ;
        float WSDNA(time, z) ;
                WSDNA:long_name = "W in saturated downdrafts averaged over the whole do-
main" ;
                WSDNA:units = "m/s" ;
                WSDNA:missing_value = -9999.f ;
        float TLWSDN(time, z) ;
                TLWSDN:long_name = "TLW in saturated downdrafts averaged over the whole
domain" ;
                TLWSDN:units = "Km/s" ;
                TLWSDN:missing_value = -9999.f ;
```

```
      float TVWSDN(time, z) ;
             TVWSDN:long_name = "TVW in saturated downdrafts averaged over the whole
domain" ;
             TVWSDN:units = "Km/s" ;
             TVWSDN:missing_value = -9999.f ;
      float QTWSDN(time, z) ;
             QTWSDN:long_name = "QTW in saturated downdrafts averaged over the whole
domain" ;
             QTWSDN:units = "g/kg m/s" ;
             QTWSDN:missing_value = -9999.f ;
      float QCWSDN(time, z) ;
             QCWSDN:long_name = "QCW in saturated downdrafts averaged over the whole
domain" ;
             QCWSDN:units = "g/kg m/s" ;
             QCWSDN:missing_value = -9999.f ;
      float QIWSDN(time, z) ;
             QIWSDN:long_name = "QIW in saturated downdrafts averaged over the whole
domain" ;
             QIWSDN:units = "g/kg m/s" ;
             QIWSDN:missing_value = -9999.f ;
      float HFSDN(time, z) ;
             HFSDN:long_name = "Mean Frozen MSE in saturated downdrafts" ;
             HFSDN:units = "K" ;
             HFSDN:missing_value = -9999.f ;
      float HFSDNA(time, z) ;
             HFSDNA:long_name = "Mean Frozen MSE anomaly in saturated downdrafts" ;
             HFSDNA:units = "K" ;
             HFSDNA:missing_value = -9999.f ;
      float USDNA(time, z) ;
             USDNA:long_name = "Mean U anomaly in saturated downdrafts" ;
             USDNA:units = "m/s" ;
             USDNA:missing_value = -9999.f ;
      float VSDNA(time, z) ;
             VSDNA:long_name = "Mean V anomaly in saturated downdrafts" ;
             VSDNA:units = "m/s" ;
             VSDNA:missing_value = -9999.f ;
      float UPGFSDN(time, z) ;
             UPGFSDN:long_name = "Zonal pressure gradient in saturated downdrafts" ;
             UPGFSDN:units = "m/s2" ;
             UPGFSDN:missing_value = -9999.f ;
      float VPGFSDN(time, z) ;
             VPGFSDN:long_name = "Meridional pressure gradient in saturated down-
drafts" ;
             VPGFSDN:units = "m/s2" ;
             VPGFSDN:missing_value = -9999.f ;
      float WPGFSDN(time, z) ;
             WPGFSDN:long_name = "Vertical pressure gradient in saturated downdrafts" ;
             WPGFSDN:units = "m/s2" ;
             WPGFSDN:missing_value = -9999.f ;
      float UWSDN(time, z) ;
             UWSDN:long_name = "UW in saturated downdrafts" ;
             UWSDN:units = "m2/s2" ;
             UWSDN:missing_value = -9999.f ;
      float VWSDN(time, z) ;
             VWSDN:long_name = "VW in saturated downdrafts" ;
             VWSDN:units = "m2/s2" ;
             VWSDN:missing_value = -9999.f ;
```

```
	float UWSBSDN(time, z) ;
		UWSBSDN:long_name = "Subgrid UW in saturated downdrafts" ;
		UWSBSDN:units = "m2/s2" ;
		UWSBSDN:missing_value = -9999.f ;
	float VWSBSDN(time, z) ;
		VWSBSDN:long_name = "Subgrid VW in saturated downdrafts" ;
		VWSBSDN:units = "m2/s2" ;
		VWSBSDN:missing_value = -9999.f ;
	float MFSDN(time, z) ;
		MFSDN:long_name = "Mass flux in saturated downdrafts averaged over the whole domain" ;
		MFSDN:units = "kg/m2/s" ;
		MFSDN:missing_value = -9999.f ;
	float MFHSDN(time, z) ;
		MFHSDN:long_name = "RHO*W*HF in saturated downdrafts averaged over the whole domain" ;
		MFHSDN:units = "K kg/m2/s" ;
		MFHSDN:missing_value = -9999.f ;
	float MFHSDNA(time, z) ;
		MFHSDNA:long_name = "RHO*W*HF anomaly in saturated downdrafts averaged over the whole domain" ;
		MFHSDNA:units = "K kg/m2/s" ;
		MFHSDNA:missing_value = -9999.f ;
	float MFTLSDN(time, z) ;
		MFTLSDN:long_name = "RHO*W*TL in saturated downdrafts averaged over the whole domain" ;
		MFTLSDN:units = "K kg/m2/s" ;
		MFTLSDN:missing_value = -9999.f ;
	float MFTLSDNA(time, z) ;
		MFTLSDNA:long_name = "RHO*W*TL anomaly in saturated downdrafts averaged over the whole domain" ;
		MFTLSDNA:units = "K kg/m2/s" ;
		MFTLSDNA:missing_value = -9999.f ;
	float MFTVSDN(time, z) ;
		MFTVSDN:long_name = "RHO*W*TV in saturated downdrafts averaged over the whole domain" ;
		MFTVSDN:units = "K kg/m2/s" ;
		MFTVSDN:missing_value = -9999.f ;
	float MFTVSDNA(time, z) ;
		MFTVSDNA:long_name = "RHO*W*TV anomaly in saturated downdrafts averaged over the whole domain" ;
		MFTVSDNA:units = "K kg/m2/s" ;
		MFTVSDNA:missing_value = -9999.f ;
	float MFQTSDN(time, z) ;
		MFQTSDN:long_name = "RHO*W*QT in saturated downdrafts averaged over the whole domain" ;
		MFQTSDN:units = "g/m2/s" ;
		MFQTSDN:missing_value = -9999.f ;
	float MFQTSDNA(time, z) ;
		MFQTSDNA:long_name = "RHO*W*QT anomaly in saturated downdrafts averaged over the whole domain" ;
		MFQTSDNA:units = "g/m2/s" ;
		MFQTSDNA:missing_value = -9999.f ;
	float RUWSDN(time, z) ;
		RUWSDN:long_name = "RHOUW in saturated downdrafts averaged over the whole domain" ;
		RUWSDN:units = "kg/m/s2" ;
		RUWSDN:missing_value = -9999.f ;
```

```
	float RVWSDN(time, z) ;
		RVWSDN:long_name = “RHOVW in saturated downdrafts averaged over the whole 
domain” ;
		RVWSDN:units = “kg/m/s2” ;
		RVWSDN:missing_value = -9999.f ;
	float RWWSDN(time, z) ;
		RWWSDN:long_name = “RHOWW in saturated downdrafts averaged over the whole 
domain” ;
		RWWSDN:units = “kg/m/s2” ;
		RWWSDN:missing_value = -9999.f ;
	float ENV(time, z) ;
		ENV:long_name = “unsaturated environment Fraction” ;
		ENV:units = “” ;
		ENV:missing_value = -9999.f ;
	float WENV(time, z) ;
		WENV:long_name = “Mean W in unsaturated environment” ;
		WENV:units = “m/s” ;
		WENV:missing_value = -9999.f ;
	float UENV(time, z) ;
		UENV:long_name = “Mean U in unsaturated environment” ;
		UENV:units = “m/s” ;
		UENV:missing_value = -9999.f ;
	float VENV(time, z) ;
		VENV:long_name = “Mean V in unsaturated environment” ;
		VENV:units = “m/s” ;
		VENV:missing_value = -9999.f ;
	float MSEENV(time, z) ;
		MSEENV:long_name = “Mean moist static energy in unsaturated environment” ;
		MSEENV:units = “K” ;
		MSEENV:missing_value = -9999.f ;
	float DSEENV(time, z) ;
		DSEENV:long_name = “Mean dry static energy in unsaturated environment” ;
		DSEENV:units = “K” ;
		DSEENV:missing_value = -9999.f ;
	float TLENV(time, z) ;
		TLENV:long_name = “Mean liquid-ice static energy in unsaturated environ-
ment” ;
		TLENV:units = “K” ;
		TLENV:missing_value = -9999.f ;
	float TAENV(time, z) ;
		TAENV:long_name = “Mean TABS in unsaturated environment” ;
		TAENV:units = “K” ;
		TAENV:missing_value = -9999.f ;
	float TVENV(time, z) ;
		TVENV:long_name = “Mean THETAV in unsaturated environment” ;
		TVENV:units = “K” ;
		TVENV:missing_value = -9999.f ;
	float TVENVA(time, z) ;
		TVENVA:long_name = “Mean THETAV anomaly in unsaturated environment” ;
		TVENVA:units = “K” ;
		TVENVA:missing_value = -9999.f ;
	float QTENV(time, z) ;
		QTENV:long_name = “Mean QT in unsaturated environment” ;
		QTENV:units = “g/kg” ;
		QTENV:missing_value = -9999.f ;
	float QNENV(time, z) ;
		QNENV:long_name = “Mean QN in unsaturated environment” ;
		QNENV:units = “g/kg” ;
		QNENV:missing_value = -9999.f ;
```

```
       float QPENV(time, z) ;
              QPENV:long_name = "Mean QP in unsaturated environment" ;
              QPENV:units = "g/kg" ;
              QPENV:missing_value = -9999.f ;
       float WENVA(time, z) ;
              WENVA:long_name = "W in unsaturated environment averaged over the whole
domain" ;
              WENVA:units = "m/s" ;
              WENVA:missing_value = -9999.f ;
       float TLWENV(time, z) ;
              TLWENV:long_name = "TLW in unsaturated environment averaged over the
whole domain" ;
              TLWENV:units = "Km/s" ;
              TLWENV:missing_value = -9999.f ;
       float TVWENV(time, z) ;
              TVWENV:long_name = "TVW in unsaturated environment averaged over the
whole domain" ;
              TVWENV:units = "Km/s" ;
              TVWENV:missing_value = -9999.f ;
       float QTWENV(time, z) ;
              QTWENV:long_name = "QTW in unsaturated environment averaged over the
whole domain" ;
              QTWENV:units = "g/kg m/s" ;
              QTWENV:missing_value = -9999.f ;
       float QCWENV(time, z) ;
              QCWENV:long_name = "QCW in unsaturated environment averaged over the
whole domain" ;
              QCWENV:units = "g/kg m/s" ;
              QCWENV:missing_value = -9999.f ;
       float QIWENV(time, z) ;
              QIWENV:long_name = "QIW in unsaturated environment averaged over the
whole domain" ;
              QIWENV:units = "g/kg m/s" ;
              QIWENV:missing_value = -9999.f ;
       float HFENV(time, z) ;
              HFENV:long_name = "Mean Frozen MSE in unsaturated environment" ;
              HFENV:units = "K" ;
              HFENV:missing_value = -9999.f ;
       float HFENVA(time, z) ;
              HFENVA:long_name = "Mean Frozen MSE anomaly in unsaturated environment" ;
              HFENVA:units = "K" ;
              HFENVA:missing_value = -9999.f ;
       float UENVA(time, z) ;
              UENVA:long_name = "Mean U anomaly in unsaturated environment" ;
              UENVA:units = "m/s" ;
              UENVA:missing_value = -9999.f ;
       float VENVA(time, z) ;
              VENVA:long_name = "Mean V anomaly in unsaturated environment" ;
              VENVA:units = "m/s" ;
              VENVA:missing_value = -9999.f ;
       float UPGFENV(time, z) ;
              UPGFENV:long_name = "Zonal pressure gradient in unsaturated environment" ;
              UPGFENV:units = "m/s2" ;
              UPGFENV:missing_value = -9999.f ;
       float VPGFENV(time, z) ;
              VPGFENV:long_name = "Meridional pressure gradient in unsaturated environ-
ment" ;
              VPGFENV:units = "m/s2" ;
              VPGFENV:missing_value = -9999.f ;
```

```
        float WPGFENV(time, z) ;
                WPGFENV:long_name = "Vertical pressure gradient in unsaturated environ-
ment" ;
                WPGFENV:units = "m/s2" ;
                WPGFENV:missing_value = -9999.f ;
        float UWENV(time, z) ;
                UWENV:long_name = "UW in unsaturated environment" ;
                UWENV:units = "m2/s2" ;
                UWENV:missing_value = -9999.f ;
        float VWENV(time, z) ;
                VWENV:long_name = "VW in unsaturated environment" ;
                VWENV:units = "m2/s2" ;
                VWENV:missing_value = -9999.f ;
        float UWSBENV(time, z) ;
                UWSBENV:long_name = "Subgrid UW in unsaturated environment" ;
                UWSBENV:units = "m2/s2" ;
                UWSBENV:missing_value = -9999.f ;
        float VWSBENV(time, z) ;
                VWSBENV:long_name = "Subgrid VW in unsaturated environment" ;
                VWSBENV:units = "m2/s2" ;
                VWSBENV:missing_value = -9999.f ;
        float MFENV(time, z) ;
                MFENV:long_name = "Mass flux in unsaturated environment averaged over the
whole domain" ;
                MFENV:units = "kg/m2/s" ;
                MFENV:missing_value = -9999.f ;
        float MFHENV(time, z) ;
                MFHENV:long_name = "RHO*W*HF in unsaturated environment averaged over the
whole domain" ;
                MFHENV:units = "K kg/m2/s" ;
                MFHENV:missing_value = -9999.f ;
        float MFHENVA(time, z) ;
                MFHENVA:long_name = "RHO*W*HF anomaly in unsaturated environment averaged
over the whole domain" ;
                MFHENVA:units = "K kg/m2/s" ;
                MFHENVA:missing_value = -9999.f ;
        float MFTLENV(time, z) ;
                MFTLENV:long_name = "RHO*W*TL in unsaturated environment averaged over
the whole domain" ;
                MFTLENV:units = "K kg/m2/s" ;
                MFTLENV:missing_value = -9999.f ;
        float MFTLENVA(time, z) ;
                MFTLENVA:long_name = "RHO*W*TL anomaly in unsaturated environment aver-
aged over the whole domain" ;
                MFTLENVA:units = "K kg/m2/s" ;
                MFTLENVA:missing_value = -9999.f ;
        float MFTVENV(time, z) ;
                MFTVENV:long_name = "RHO*W*TV in unsaturated environment averaged over
the whole domain" ;
                MFTVENV:units = "K kg/m2/s" ;
                MFTVENV:missing_value = -9999.f ;
        float MFTVENVA(time, z) ;
                MFTVENVA:long_name = "RHO*W*TV anomaly in unsaturated environment aver-
aged over the whole domain" ;
                MFTVENVA:units = "K kg/m2/s" ;
                MFTVENVA:missing_value = -9999.f ;
        float MFQTENV(time, z) ;
```

```
             MFQTENV:long_name = “RHO*W*QT in unsaturated environment averaged over
the whole domain” ;
             MFQTENV:units = “g/m2/s” ;
             MFQTENV:missing_value = -9999.f ;
      float MFQTENVA(time, z) ;
             MFQTENVA:long_name = “RHO*W*QT anomaly in unsaturated environment aver-
aged over the whole domain” ;
             MFQTENVA:units = “g/m2/s” ;
             MFQTENVA:missing_value = -9999.f ;
      float RUWENV(time, z) ;
             RUWENV:long_name = “RHOUW in unsaturated environment averaged over the
whole domain” ;
             RUWENV:units = “kg/m/s2” ;
             RUWENV:missing_value = -9999.f ;
      float RVWENV(time, z) ;
             RVWENV:long_name = “RHOVW in unsaturated environment averaged over the
whole domain” ;
             RVWENV:units = “kg/m/s2” ;
             RVWENV:missing_value = -9999.f ;
      float RWWENV(time, z) ;
             RWWENV:long_name = “RHOWW in unsaturated environment averaged over the
whole domain” ;
             RWWENV:units = “kg/m/s2” ;
             RWWENV:missing_value = -9999.f ;
      float QTFLUX(time, z) ;
             QTFLUX:long_name = “Total (resolved + subgrid) total water (vapor+cloud)
flux” ;
             QTFLUX:units = “W/m2” ;
             QTFLUX:missing_value = -9999.f ;
      float QTO(time, z) ;
             QTO:long_name = “TOTAL WATER (VAPOR + CLOUD LIQUID)” ;
             QTO:units = “g/kg” ;
             QTO:missing_value = -9999.f ;
      float QTO2(time, z) ;
             QTO2:long_name = “Variance of TOTAL WATER (VAPOR + CLOUD LIQUID)” ;
             QTO2:units = “(g/kg)^2” ;
             QTO2:missing_value = -9999.f ;
      float QTOADV(time, z) ;
             QTOADV:long_name = “Tendency of TOTAL WATER (VAPOR + CLOUD LIQUID) due to
resolved vertical advectio” ;
             QTOADV:units = “g/kg/day” ;
             QTOADV:missing_value = -9999.f ;
      float QTODIFF(time, z) ;
             QTODIFF:long_name = “Tendency of TOTAL WATER (VAPOR + CLOUD LIQUID) due
to vertical SGS transport” ;
             QTODIFF:units = “g/kg/day” ;
             QTODIFF:missing_value = -9999.f ;
      float QTOLSADV(time, z) ;
             QTOLSADV:long_name = “Tendency of TOTAL WATER (VAPOR + CLOUD LIQUID) due
to large-scale vertical advec” ;
             QTOLSADV:units = “g/kg/day” ;
             QTOLSADV:missing_value = -9999.f ;
      float QTOMPHY(time, z) ;
             QTOMPHY:long_name = “Tendency of TOTAL WATER (VAPOR + CLOUD LIQUID) due
to microphysical processes” ;
             QTOMPHY:units = “g/kg/day” ;
             QTOMPHY:missing_value = -9999.f ;
      float QTOSED(time, z) ;
             QTOSED:long_name = “Tendency of TOTAL WATER (VAPOR + CLOUD LIQUID) due to
sedimentation” ;
```

```
            QTOSED:units = “g/kg/day” ;
            QTOSED:missing_value = -9999.f ;
      float QTOFLXR(time, z) ;
            QTOFLXR:long_name = “Resolved flux of TOTAL WATER (VAPOR + CLOUD LIQUID)”
;
            QTOFLXR:units = “W/m2” ;
            QTOFLXR:missing_value = -9999.f ;
      float QTOFLXS(time, z) ;
            QTOFLXS:long_name = “Subgrid flux of TOTAL WATER (VAPOR + CLOUD LIQUID)” ;
            QTOFLXS:units = “W/m2” ;
            QTOFLXS:missing_value = -9999.f ;
      float QTOSDFLX(time, z) ;
            QTOSDFLX:long_name = “Sedimentation flux of TOTAL WATER (VAPOR + CLOUD
LIQUID)” ;
            QTOSDFLX:units = “W/m2” ;
            QTOSDFLX:missing_value = -9999.f ;
      float NC(time, z) ;
            NC:long_name = “CLOUD WATER NUMBER CONCENTRATION” ;
            NC:units = “#/cm3” ;
            NC:missing_value = -9999.f ;
      float NCADV(time, z) ;
            NCADV:long_name = “Tendency of CLOUD WATER NUMBER CONCENTRATION due to
resolved vertical advection” ;
            NCADV:units = “#/cm3/day” ;
            NCADV:missing_value = -9999.f ;
      float NCDIFF(time, z) ;
            NCDIFF:long_name = “Tendency of CLOUD WATER NUMBER CONCENTRATION due to
vertical SGS transport” ;
            NCDIFF:units = “#/cm3/day” ;
            NCDIFF:missing_value = -9999.f ;
      float NCLSADV(time, z) ;
            NCLSADV:long_name = “Tendency of CLOUD WATER NUMBER CONCENTRATION due to
large-scale vertical advecti” ;
            NCLSADV:units = “#/cm3/day” ;
            NCLSADV:missing_value = -9999.f ;
      float NCMPHY(time, z) ;
            NCMPHY:long_name = “Tendency of CLOUD WATER NUMBER CONCENTRATION due to
microphysical processes” ;
            NCMPHY:units = “#/cm3/day” ;
            NCMPHY:missing_value = -9999.f ;
      float NCSED(time, z) ;
            NCSED:long_name = “Tendency of CLOUD WATER NUMBER CONCENTRATION due to
sedimentation” ;
            NCSED:units = “#/cm3/day” ;
            NCSED:missing_value = -9999.f ;
      float NCFLXR(time, z) ;
            NCFLXR:long_name = “Resolved flux of CLOUD WATER NUMBER CONCENTRATION” ;
            NCFLXR:units = “#/m2/s” ;
            NCFLXR:missing_value = -9999.f ;
      float NCFLXS(time, z) ;
            NCFLXS:long_name = “Subgrid flux of CLOUD WATER NUMBER CONCENTRATION” ;
            NCFLXS:units = “#/m2/s” ;
            NCFLXS:missing_value = -9999.f ;
      float NCSDFLX(time, z) ;
            NCSDFLX:long_name = “Sedimentation flux of CLOUD WATER NUMBER CONCENTRA-
TION” ;
            NCSDFLX:units = “#/m2/s” ;
            NCSDFLX:missing_value = -9999.f ;
```

```
        float QR(time, z) ;
                QR:long_name = "RAIN" ;
                QR:units = "g/kg" ;
                QR:missing_value = -9999.f ;
        float QRADV(time, z) ;
                QRADV:long_name = "Tendency of RAIN due to resolved vertical advection" ;
                QRADV:units = "g/kg/day" ;
                QRADV:missing_value = -9999.f ;
        float QRDIFF(time, z) ;
                QRDIFF:long_name = "Tendency of RAIN due to vertical SGS transport" ;
                QRDIFF:units = "g/kg/day" ;
                QRDIFF:missing_value = -9999.f ;
        float QRLSADV(time, z) ;
                QRLSADV:long_name = "Tendency of RAIN due to large-scale vertical advec-
tion" ;
                QRLSADV:units = "g/kg/day" ;
                QRLSADV:missing_value = -9999.f ;
        float QRMPHY(time, z) ;
                QRMPHY:long_name = "Tendency of RAIN due to microphysical processes" ;
                QRMPHY:units = "g/kg/day" ;
                QRMPHY:missing_value = -9999.f ;
        float QRSED(time, z) ;
                QRSED:long_name = "Tendency of RAIN due to sedimentation" ;
                QRSED:units = "g/kg/day" ;
                QRSED:missing_value = -9999.f ;
        float QRFLXR(time, z) ;
                QRFLXR:long_name = "Resolved flux of RAIN" ;
                QRFLXR:units = "W/m2" ;
                QRFLXR:missing_value = -9999.f ;
        float QRFLXS(time, z) ;
                QRFLXS:long_name = "Subgrid flux of RAIN" ;
                QRFLXS:units = "W/m2" ;
                QRFLXS:missing_value = -9999.f ;
        float QRSDFLX(time, z) ;
                QRSDFLX:long_name = "Sedimentation flux of RAIN" ;
                QRSDFLX:units = "W/m2" ;
                QRSDFLX:missing_value = -9999.f ;
        float QRFRAC(time, z) ;
                QRFRAC:long_name = "RAIN FRACTION" ;
                QRFRAC:units = "1" ;
                QRFRAC:missing_value = -9999.f ;
        float TAUQR(time, z) ;
                TAUQR:long_name = "Approx optical depth of RAIN" ;
                TAUQR:units = "1" ;
                TAUQR:missing_value = -9999.f ;
        float QROEFFR(time, z) ;
                QROEFFR:long_name = "Mixing ratio of RAIN over effective radius, EFFR =
QR/QROEFFR" ;
                QROEFFR:units = "g/kg/micro" ;
                QROEFFR:missing_value = -9999.f ;
        float NR(time, z) ;
                NR:long_name = "RAIN NUMBER CONCENTRATION" ;
                NR:units = "#/cm3" ;
                NR:missing_value = -9999.f ;
        float NRADV(time, z) ;
                NRADV:long_name = "Tendency of RAIN NUMBER CONCENTRATION due to resolved
vertical advection" ;
                NRADV:units = "#/cm3/day" ;
                NRADV:missing_value = -9999.f ;
```

```
      float NRDIFF(time, z) ;
             NRDIFF:long_name = "Tendency of RAIN NUMBER CONCENTRATION due to vertical
SGS transport" ;
             NRDIFF:units = "#/cm3/day" ;
             NRDIFF:missing_value = -9999.f ;
      float NRLSADV(time, z) ;
             NRLSADV:long_name = "Tendency of RAIN NUMBER CONCENTRATION due to large-
scale vertical advection" ;
             NRLSADV:units = "#/cm3/day" ;
             NRLSADV:missing_value = -9999.f ;
      float NRMPHY(time, z) ;
             NRMPHY:long_name = "Tendency of RAIN NUMBER CONCENTRATION due to micro-
physical processes" ;
             NRMPHY:units = "#/cm3/day" ;
             NRMPHY:missing_value = -9999.f ;
      float NRSED(time, z) ;
             NRSED:long_name = "Tendency of RAIN NUMBER CONCENTRATION due to sedimen-
tation" ;
             NRSED:units = "#/cm3/day" ;
             NRSED:missing_value = -9999.f ;
      float NRFLXR(time, z) ;
             NRFLXR:long_name = "Resolved flux of RAIN NUMBER CONCENTRATION" ;
             NRFLXR:units = "#/m2/s" ;
             NRFLXR:missing_value = -9999.f ;
      float NRFLXS(time, z) ;
             NRFLXS:long_name = "Subgrid flux of RAIN NUMBER CONCENTRATION" ;
             NRFLXS:units = "#/m2/s" ;
             NRFLXS:missing_value = -9999.f ;
      float NRSDFLX(time, z) ;
             NRSDFLX:long_name = "Sedimentation flux of RAIN NUMBER CONCENTRATION" ;
             NRSDFLX:units = "#/m2/s" ;
             NRSDFLX:missing_value = -9999.f ;
      float QI(time, z) ;
             QI:long_name = "CLOUD ICE" ;
             QI:units = "g/kg" ;
             QI:missing_value = -9999.f ;
      float QIADV(time, z) ;
             QIADV:long_name = "Tendency of CLOUD ICE due to resolved vertical advec-
tion" ;
             QIADV:units = "g/kg/day" ;
             QIADV:missing_value = -9999.f ;
      float QIDIFF(time, z) ;
             QIDIFF:long_name = "Tendency of CLOUD ICE due to vertical SGS transport" ;
             QIDIFF:units = "g/kg/day" ;
             QIDIFF:missing_value = -9999.f ;
      float QILSADV(time, z) ;
             QILSADV:long_name = "Tendency of CLOUD ICE due to large-scale vertical
advection" ;
             QILSADV:units = "g/kg/day" ;
             QILSADV:missing_value = -9999.f ;
      float QIMPHY(time, z) ;
             QIMPHY:long_name = "Tendency of CLOUD ICE due to microphysical processes" ;
             QIMPHY:units = "g/kg/day" ;
             QIMPHY:missing_value = -9999.f ;
      float QISED(time, z) ;
             QISED:long_name = "Tendency of CLOUD ICE due to sedimentation" ;
             QISED:units = "g/kg/day" ;
             QISED:missing_value = -9999.f ;
```

```
        float QIFLXR(time, z) ;
                QIFLXR:long_name = "Resolved flux of CLOUD ICE" ;
                QIFLXR:units = "W/m2" ;
                QIFLXR:missing_value = -9999.f ;
        float QIFLXS(time, z) ;
                QIFLXS:long_name = "Subgrid flux of CLOUD ICE" ;
                QIFLXS:units = "W/m2" ;
                QIFLXS:missing_value = -9999.f ;
        float QISDFLX(time, z) ;
                QISDFLX:long_name = "Sedimentation flux of CLOUD ICE" ;
                QISDFLX:units = "W/m2" ;
                QISDFLX:missing_value = -9999.f ;
        float QIFRAC(time, z) ;
                QIFRAC:long_name = "CLOUD ICE FRACTION" ;
                QIFRAC:units = "1" ;
                QIFRAC:missing_value = -9999.f ;
        float TAUQI(time, z) ;
                TAUQI:long_name = "Approx optical depth of CLOUD ICE" ;
                TAUQI:units = "1" ;
                TAUQI:missing_value = -9999.f ;
        float QIOEFFR(time, z) ;
                QIOEFFR:long_name = "Mixing ratio of CLOUD ICE over effective radius,
EFFR = QI/QIOEFFR" ;
                QIOEFFR:units = "g/kg/micro" ;
                QIOEFFR:missing_value = -9999.f ;
        float NI(time, z) ;
                NI:long_name = "CLOUD ICE NUMBER CONCENTRATION" ;
                NI:units = "#/cm3" ;
                NI:missing_value = -9999.f ;
        float NIADV(time, z) ;
                NIADV:long_name = "Tendency of CLOUD ICE NUMBER CONCENTRATION due to re-
solved vertical advection" ;
                NIADV:units = "#/cm3/day" ;
                NIADV:missing_value = -9999.f ;
        float NIDIFF(time, z) ;
                NIDIFF:long_name = "Tendency of CLOUD ICE NUMBER CONCENTRATION due to
vertical SGS transport" ;
                NIDIFF:units = "#/cm3/day" ;
                NIDIFF:missing_value = -9999.f ;
        float NILSADV(time, z) ;
                NILSADV:long_name = "Tendency of CLOUD ICE NUMBER CONCENTRATION due to
large-scale vertical advection" ;
                NILSADV:units = "#/cm3/day" ;
                NILSADV:missing_value = -9999.f ;
        float NIMPHY(time, z) ;
                NIMPHY:long_name = "Tendency of CLOUD ICE NUMBER CONCENTRATION due to mi-
crophysical processes" ;
                NIMPHY:units = "#/cm3/day" ;
                NIMPHY:missing_value = -9999.f ;
        float NISED(time, z) ;
                NISED:long_name = "Tendency of CLOUD ICE NUMBER CONCENTRATION due to sed-
imentation" ;
                NISED:units = "#/cm3/day" ;
                NISED:missing_value = -9999.f ;
        float NIFLXR(time, z) ;
                NIFLXR:long_name = "Resolved flux of CLOUD ICE NUMBER CONCENTRATION" ;
                NIFLXR:units = "#/m2/s" ;
                NIFLXR:missing_value = -9999.f ;
```

```
        float NIFLXS(time, z) ;
                NIFLXS:long_name = "Subgrid flux of CLOUD ICE NUMBER CONCENTRATION" ;
                NIFLXS:units = "#/m2/s" ;
                NIFLXS:missing_value = -9999.f ;
        float NISDFLX(time, z) ;
                NISDFLX:long_name = "Sedimentation flux of CLOUD ICE NUMBER CONCENTRATION" ;
                NISDFLX:units = "#/m2/s" ;
                NISDFLX:missing_value = -9999.f ;
        float QS(time, z) ;
                QS:long_name = "SNOW" ;
                QS:units = "g/kg" ;
                QS:missing_value = -9999.f ;
        float QSADV(time, z) ;
                QSADV:long_name = "Tendency of SNOW due to resolved vertical advection" ;
                QSADV:units = "g/kg/day" ;
                QSADV:missing_value = -9999.f ;
        float QSDIFF(time, z) ;
                QSDIFF:long_name = "Tendency of SNOW due to vertical SGS transport" ;
                QSDIFF:units = "g/kg/day" ;
                QSDIFF:missing_value = -9999.f ;
        float QSLSADV(time, z) ;
                QSLSADV:long_name = "Tendency of SNOW due to large-scale vertical advec-
tion" ;
                QSLSADV:units = "g/kg/day" ;
                QSLSADV:missing_value = -9999.f ;
        float QSMPHY(time, z) ;
                QSMPHY:long_name = "Tendency of SNOW due to microphysical processes" ;
                QSMPHY:units = "g/kg/day" ;
                QSMPHY:missing_value = -9999.f ;
        float QSSED(time, z) ;
                QSSED:long_name = "Tendency of SNOW due to sedimentation" ;
                QSSED:units = "g/kg/day" ;
                QSSED:missing_value = -9999.f ;
        float QSFLXR(time, z) ;
                QSFLXR:long_name = "Resolved flux of SNOW" ;
                QSFLXR:units = "W/m2" ;
                QSFLXR:missing_value = -9999.f ;
        float QSFLXS(time, z) ;
                QSFLXS:long_name = "Subgrid flux of SNOW" ;
                QSFLXS:units = "W/m2" ;
                QSFLXS:missing_value = -9999.f ;
        float QSSDFLX(time, z) ;
                QSSDFLX:long_name = "Sedimentation flux of SNOW" ;
                QSSDFLX:units = "W/m2" ;
                QSSDFLX:missing_value = -9999.f ;
        float QSFRAC(time, z) ;
                QSFRAC:long_name = "SNOW FRACTION" ;
                QSFRAC:units = "1" ;
                QSFRAC:missing_value = -9999.f ;
        float TAUQS(time, z) ;
                TAUQS:long_name = "Approx optical depth of SNOW" ;
                TAUQS:units = "1" ;
                TAUQS:missing_value = -9999.f ;
        float QSOEFFR(time, z) ;
                QSOEFFR:long_name = "Mixing ratio of SNOW over effective radius, EFFR =
QS/QSOEFFR" ;
                QSOEFFR:units = "g/kg/micro" ;
                QSOEFFR:missing_value = -9999.f ;
```

```
       float NS(time, z) ;
              NS:long_name = "SNOW NUMBER CONCENTRATION" ;
              NS:units = "#/cm3" ;
              NS:missing_value = -9999.f ;
       float NSADV(time, z) ;
              NSADV:long_name = "Tendency of SNOW NUMBER CONCENTRATION due to resolved
vertical advection" ;
              NSADV:units = "#/cm3/day" ;
              NSADV:missing_value = -9999.f ;
       float NSDIFF(time, z) ;
              NSDIFF:long_name = "Tendency of SNOW NUMBER CONCENTRATION due to vertical
SGS transport" ;
              NSDIFF:units = "#/cm3/day" ;
              NSDIFF:missing_value = -9999.f ;
       float NSLSADV(time, z) ;
              NSLSADV:long_name = "Tendency of SNOW NUMBER CONCENTRATION due to large-
scale vertical advection" ;
              NSLSADV:units = "#/cm3/day" ;
              NSLSADV:missing_value = -9999.f ;
       float NSMPHY(time, z) ;
              NSMPHY:long_name = "Tendency of SNOW NUMBER CONCENTRATION due to micro-
physical processes" ;
              NSMPHY:units = "#/cm3/day" ;
              NSMPHY:missing_value = -9999.f ;
       float NSSED(time, z) ;
              NSSED:long_name = "Tendency of SNOW NUMBER CONCENTRATION due to sedimen-
tation" ;
              NSSED:units = "#/cm3/day" ;
              NSSED:missing_value = -9999.f ;
       float NSFLXR(time, z) ;
              NSFLXR:long_name = "Resolved flux of SNOW NUMBER CONCENTRATION" ;
              NSFLXR:units = "#/m2/s" ;
              NSFLXR:missing_value = -9999.f ;
       float NSFLXS(time, z) ;
              NSFLXS:long_name = "Subgrid flux of SNOW NUMBER CONCENTRATION" ;
              NSFLXS:units = "#/m2/s" ;
              NSFLXS:missing_value = -9999.f ;
       float NSSDFLX(time, z) ;
              NSSDFLX:long_name = "Sedimentation flux of SNOW NUMBER CONCENTRATION" ;
              NSSDFLX:units = "#/m2/s" ;
              NSSDFLX:missing_value = -9999.f ;
       float QG(time, z) ;
              QG:long_name = "GRAUPEL" ;
              QG:units = "g/kg" ;
              QG:missing_value = -9999.f ;
       float QGADV(time, z) ;
              QGADV:long_name = "Tendency of GRAUPEL due to resolved vertical advec-
tion" ;
              QGADV:units = "g/kg/day" ;
              QGADV:missing_value = -9999.f ;
       float QGDIFF(time, z) ;
              QGDIFF:long_name = "Tendency of GRAUPEL due to vertical SGS transport" ;
              QGDIFF:units = "g/kg/day" ;
              QGDIFF:missing_value = -9999.f ;
       float QGLSADV(time, z) ;
              QGLSADV:long_name = "Tendency of GRAUPEL due to large-scale vertical ad-
vection" ;
              QGLSADV:units = "g/kg/day" ;
              QGLSADV:missing_value = -9999.f ;
```

```
        float QGMPHY(time, z) ;
                QGMPHY:long_name = "Tendency of GRAUPEL due to microphysical processes" ;
                QGMPHY:units = "g/kg/day" ;
                QGMPHY:missing_value = -9999.f ;
        float QGSED(time, z) ;
                QGSED:long_name = "Tendency of GRAUPEL due to sedimentation" ;
                QGSED:units = "g/kg/day" ;
                QGSED:missing_value = -9999.f ;
        float QGFLXR(time, z) ;
                QGFLXR:long_name = "Resolved flux of GRAUPEL" ;
                QGFLXR:units = "W/m2" ;
                QGFLXR:missing_value = -9999.f ;
        float QGFLXS(time, z) ;
                QGFLXS:long_name = "Subgrid flux of GRAUPEL" ;
                QGFLXS:units = "W/m2" ;
                QGFLXS:missing_value = -9999.f ;
        float QGSDFLX(time, z) ;
                QGSDFLX:long_name = "Sedimentation flux of GRAUPEL" ;
                QGSDFLX:units = "W/m2" ;
                QGSDFLX:missing_value = -9999.f ;
        float QGFRAC(time, z) ;
                QGFRAC:long_name = "GRAUPEL FRACTION" ;
                QGFRAC:units = "1" ;
                QGFRAC:missing_value = -9999.f ;
        float TAUQG(time, z) ;
                TAUQG:long_name = "Approx optical depth of GRAUPEL" ;
                TAUQG:units = "1" ;
                TAUQG:missing_value = -9999.f ;
        float QGOEFFR(time, z) ;
                QGOEFFR:long_name = "Mixing ratio of GRAUPEL over effective radius, EFFR
= QG/QGOEFFR" ;
                QGOEFFR:units = "g/kg/micro" ;
                QGOEFFR:missing_value = -9999.f ;
        float NG(time, z) ;
                NG:long_name = "GRAUPEL NUMBER CONCENTRATION" ;
                NG:units = "#/cm3" ;
                NG:missing_value = -9999.f ;
        float NGADV(time, z) ;
                NGADV:long_name = "Tendency of GRAUPEL NUMBER CONCENTRATION due to re-
solved vertical advection" ;
                NGADV:units = "#/cm3/day" ;
                NGADV:missing_value = -9999.f ;
        float NGDIFF(time, z) ;
                NGDIFF:long_name = "Tendency of GRAUPEL NUMBER CONCENTRATION due to ver-
tical SGS transport" ;
                NGDIFF:units = "#/cm3/day" ;
                NGDIFF:missing_value = -9999.f ;
        float NGLSADV(time, z) ;
                NGLSADV:long_name = "Tendency of GRAUPEL NUMBER CONCENTRATION due to
large-scale vertical advection" ;
                NGLSADV:units = "#/cm3/day" ;
                NGLSADV:missing_value = -9999.f ;
        float NGMPHY(time, z) ;
                NGMPHY:long_name = "Tendency of GRAUPEL NUMBER CONCENTRATION due to mi-
crophysical processes" ;
                NGMPHY:units = "#/cm3/day" ;
                NGMPHY:missing_value = -9999.f ;
```

```
	float NGSED(time, z) ;
		NGSED:long_name = "Tendency of GRAUPEL NUMBER CONCENTRATION due to sedi-
mentation" ;
		NGSED:units = "#/cm3/day" ;
		NGSED:missing_value = -9999.f ;
	float NGFLXR(time, z) ;
		NGFLXR:long_name = "Resolved flux of GRAUPEL NUMBER CONCENTRATION" ;
		NGFLXR:units = "#/m2/s" ;
		NGFLXR:missing_value = -9999.f ;
	float NGFLXS(time, z) ;
		NGFLXS:long_name = "Subgrid flux of GRAUPEL NUMBER CONCENTRATION" ;
		NGFLXS:units = "#/m2/s" ;
		NGFLXS:missing_value = -9999.f ;
	float NGSDFLX(time, z) ;
		NGSDFLX:long_name = "Sedimentation flux of GRAUPEL NUMBER CONCENTRATION" ;
		NGSDFLX:units = "#/m2/s" ;
		NGSDFLX:missing_value = -9999.f ;
	float QC(time, z) ;
		QC:long_name = "Cloud liquid water mass mixing ratio" ;
		QC:units = "g/kg" ;
		QC:missing_value = -9999.f ;
	float TAUQC(time, z) ;
		TAUQC:long_name = "Approx optical depth of cloud liquid water" ;
		TAUQC:units = "1" ;
		TAUQC:missing_value = -9999.f ;
	float QCOEFFR(time, z) ;
		QCOEFFR:long_name = "Mixing ratio of QC over effective radius, EFFR = QC/
QCOEFFR" ;
		QCOEFFR:units = "g/kg/micro" ;
		QCOEFFR:missing_value = -9999.f ;
	float QVCLD(time, z) ;
		QVCLD:long_name = "Water vapor mixing ratio in cloud" ;
		QVCLD:units = "kg/kg" ;
		QVCLD:missing_value = -9999.f ;
	float QCCLD(time, z) ;
		QCCLD:long_name = "Cloud liquid water mixing ratio in cloud" ;
		QCCLD:units = "kg/kg" ;
		QCCLD:missing_value = -9999.f ;
	float QTOCLD(time, z) ;
		QTOCLD:long_name = "TOTAL WATER (VAPOR + CLOUD LIQUID) in cloud" ;
		QTOCLD:units = "g/kg" ;
		QTOCLD:missing_value = -9999.f ;
	float NCCLD(time, z) ;
		NCCLD:long_name = "CLOUD WATER NUMBER CONCENTRATION in cloud" ;
		NCCLD:units = "#/cm3" ;
		NCCLD:missing_value = -9999.f ;
	float QRCLD(time, z) ;
		QRCLD:long_name = "RAIN in cloud" ;
		QRCLD:units = "g/kg" ;
		QRCLD:missing_value = -9999.f ;
	float NRCLD(time, z) ;
		NRCLD:long_name = "RAIN NUMBER CONCENTRATION in cloud" ;
		NRCLD:units = "#/cm3" ;
		NRCLD:missing_value = -9999.f ;
	float QICLD(time, z) ;
		QICLD:long_name = "CLOUD ICE in cloud" ;
		QICLD:units = "g/kg" ;
		QICLD:missing_value = -9999.f ;
```

```
float NICLD(time, z) ;
       NICLD:long_name = "CLOUD ICE NUMBER CONCENTRATION in cloud" ;
       NICLD:units = "#/cm3" ;
       NICLD:missing_value = -9999.f ;
float QSCLD(time, z) ;
       QSCLD:long_name = "SNOW in cloud" ;
       QSCLD:units = "g/kg" ;
       QSCLD:missing_value = -9999.f ;
float NSCLD(time, z) ;
       NSCLD:long_name = "SNOW NUMBER CONCENTRATION in cloud" ;
       NSCLD:units = "#/cm3" ;
       NSCLD:missing_value = -9999.f ;
float QGCLD(time, z) ;
       QGCLD:long_name = "GRAUPEL in cloud" ;
       QGCLD:units = "g/kg" ;
       QGCLD:missing_value = -9999.f ;
float NGCLD(time, z) ;
       NGCLD:long_name = "GRAUPEL NUMBER CONCENTRATION in cloud" ;
       NGCLD:units = "#/cm3" ;
       NGCLD:missing_value = -9999.f ;
float QVCOR(time, z) ;
       QVCOR:long_name = "Water vapor mixing ratio in core" ;
       QVCOR:units = "kg/kg" ;
       QVCOR:missing_value = -9999.f ;
float QCCOR(time, z) ;
       QCCOR:long_name = "Cloud liquid water mixing ratio in core" ;
       QCCOR:units = "kg/kg" ;
       QCCOR:missing_value = -9999.f ;
float QTOCOR(time, z) ;
       QTOCOR:long_name = "TOTAL WATER (VAPOR + CLOUD LIQUID) in core" ;
       QTOCOR:units = "g/kg" ;
       QTOCOR:missing_value = -9999.f ;
float NCCOR(time, z) ;
       NCCOR:long_name = "CLOUD WATER NUMBER CONCENTRATION in core" ;
       NCCOR:units = "#/cm3" ;
       NCCOR:missing_value = -9999.f ;
float QRCOR(time, z) ;
       QRCOR:long_name = "RAIN in core" ;
       QRCOR:units = "g/kg" ;
       QRCOR:missing_value = -9999.f ;
float NRCOR(time, z) ;
       NRCOR:long_name = "RAIN NUMBER CONCENTRATION in core" ;
       NRCOR:units = "#/cm3" ;
       NRCOR:missing_value = -9999.f ;
float QICOR(time, z) ;
       QICOR:long_name = "CLOUD ICE in core" ;
       QICOR:units = "g/kg" ;
       QICOR:missing_value = -9999.f ;
float NICOR(time, z) ;
       NICOR:long_name = "CLOUD ICE NUMBER CONCENTRATION in core" ;
       NICOR:units = "#/cm3" ;
       NICOR:missing_value = -9999.f ;
float QSCOR(time, z) ;
       QSCOR:long_name = "SNOW in core" ;
       QSCOR:units = "g/kg" ;
       QSCOR:missing_value = -9999.f ;
float NSCOR(time, z) ;
       NSCOR:long_name = "SNOW NUMBER CONCENTRATION in core" ;
       NSCOR:units = "#/cm3" ;
       NSCOR:missing_value = -9999.f ;
```

```
float QGCOR(time, z) ;
       QGCOR:long_name = "GRAUPEL in core" ;
       QGCOR:units = "g/kg" ;
       QGCOR:missing_value = -9999.f ;
float NGCOR(time, z) ;
       NGCOR:long_name = "GRAUPEL NUMBER CONCENTRATION in core" ;
       NGCOR:units = "#/cm3" ;
       NGCOR:missing_value = -9999.f ;
float QVCDN(time, z) ;
       QVCDN:long_name = "Water vapor mixing ratio in downdraft core" ;
       QVCDN:units = "kg/kg" ;
       QVCDN:missing_value = -9999.f ;
float QCCDN(time, z) ;
       QCCDN:long_name = "Cloud liquid water mixing ratio in downdraft core" ;
       QCCDN:units = "kg/kg" ;
       QCCDN:missing_value = -9999.f ;
float QTOCDN(time, z) ;
       QTOCDN:long_name = "TOTAL WATER (VAPOR + CLOUD LIQUID) in downdraft core" ;
       QTOCDN:units = "g/kg" ;
       QTOCDN:missing_value = -9999.f ;
float NCCDN(time, z) ;
       NCCDN:long_name = "CLOUD WATER NUMBER CONCENTRATION in downdraft core" ;
       NCCDN:units = "#/cm3" ;
       NCCDN:missing_value = -9999.f ;
float QRCDN(time, z) ;
       QRCDN:long_name = "RAIN in downdraft core" ;
       QRCDN:units = "g/kg" ;
       QRCDN:missing_value = -9999.f ;
float NRCDN(time, z) ;
       NRCDN:long_name = "RAIN NUMBER CONCENTRATION in downdraft core" ;
       NRCDN:units = "#/cm3" ;
       NRCDN:missing_value = -9999.f ;
float QICDN(time, z) ;
       QICDN:long_name = "CLOUD ICE in downdraft core" ;
       QICDN:units = "g/kg" ;
       QICDN:missing_value = -9999.f ;
float NICDN(time, z) ;
       NICDN:long_name = "CLOUD ICE NUMBER CONCENTRATION in downdraft core" ;
       NICDN:units = "#/cm3" ;
       NICDN:missing_value = -9999.f ;
float QSCDN(time, z) ;
       QSCDN:long_name = "SNOW in downdraft core" ;
       QSCDN:units = "g/kg" ;
       QSCDN:missing_value = -9999.f ;
float NSCDN(time, z) ;
       NSCDN:long_name = "SNOW NUMBER CONCENTRATION in downdraft core" ;
       NSCDN:units = "#/cm3" ;
       NSCDN:missing_value = -9999.f ;
float QGCDN(time, z) ;
       QGCDN:long_name = "GRAUPEL in downdraft core" ;
       QGCDN:units = "g/kg" ;
       QGCDN:missing_value = -9999.f ;
float NGCDN(time, z) ;
       NGCDN:long_name = "GRAUPEL NUMBER CONCENTRATION in downdraft core" ;
       NGCDN:units = "#/cm3" ;
       NGCDN:missing_value = -9999.f ;
float QVSUP(time, z) ;
       QVSUP:long_name = "Water vapor mixing ratio in saturated updrafts" ;
       QVSUP:units = "kg/kg" ;
       QVSUP:missing_value = -9999.f ;
```

```
        float QCSUP(time, z) ;
                QCSUP:long_name = "Cloud liquid water mixing ratio in saturated updrafts" ;
                QCSUP:units = "kg/kg" ;
                QCSUP:missing_value = -9999.f ;
        float QTOSUP(time, z) ;
                QTOSUP:long_name = "TOTAL WATER (VAPOR + CLOUD LIQUID) in saturated up-
drafts" ;
                QTOSUP:units = "g/kg" ;
                QTOSUP:missing_value = -9999.f ;
        float NCSUP(time, z) ;
                NCSUP:long_name = "CLOUD WATER NUMBER CONCENTRATION in saturated up-
drafts" ;
                NCSUP:units = "#/cm3" ;
                NCSUP:missing_value = -9999.f ;
        float QRSUP(time, z) ;
                QRSUP:long_name = "RAIN in saturated updrafts" ;
                QRSUP:units = "g/kg" ;
                QRSUP:missing_value = -9999.f ;
        float NRSUP(time, z) ;
                NRSUP:long_name = "RAIN NUMBER CONCENTRATION in saturated updrafts" ;
                NRSUP:units = "#/cm3" ;
                NRSUP:missing_value = -9999.f ;
        float QISUP(time, z) ;
                QISUP:long_name = "CLOUD ICE in saturated updrafts" ;
                QISUP:units = "g/kg" ;
                QISUP:missing_value = -9999.f ;
        float NISUP(time, z) ;
                NISUP:long_name = "CLOUD ICE NUMBER CONCENTRATION in saturated updrafts" ;
                NISUP:units = "#/cm3" ;
                NISUP:missing_value = -9999.f ;
        float QSSUP(time, z) ;
                QSSUP:long_name = "SNOW in saturated updrafts" ;
                QSSUP:units = "g/kg" ;
                QSSUP:missing_value = -9999.f ;
        float NSSUP(time, z) ;
                NSSUP:long_name = "SNOW NUMBER CONCENTRATION in saturated updrafts" ;
                NSSUP:units = "#/cm3" ;
                NSSUP:missing_value = -9999.f ;
        float QGSUP(time, z) ;
                QGSUP:long_name = "GRAUPEL in saturated updrafts" ;
                QGSUP:units = "g/kg" ;
                QGSUP:missing_value = -9999.f ;
        float NGSUP(time, z) ;
                NGSUP:long_name = "GRAUPEL NUMBER CONCENTRATION in saturated updrafts" ;
                NGSUP:units = "#/cm3" ;
                NGSUP:missing_value = -9999.f ;
        float QVSDN(time, z) ;
                QVSDN:long_name = "Water vapor mixing ratio in saturated downdrafts" ;
                QVSDN:units = "kg/kg" ;
                QVSDN:missing_value = -9999.f ;
        float QCSDN(time, z) ;
                QCSDN:long_name = "Cloud liquid water mixing ratio in saturated down-
drafts" ;
                QCSDN:units = "kg/kg" ;
                QCSDN:missing_value = -9999.f ;
        float QTOSDN(time, z) ;
                QTOSDN:long_name = "TOTAL WATER (VAPOR + CLOUD LIQUID) in saturated down-
drafts" ;
                QTOSDN:units = "g/kg" ;
                QTOSDN:missing_value = -9999.f ;
```

```
	float NCSDN(time, z) ;
		NCSDN:long_name = "CLOUD WATER NUMBER CONCENTRATION in saturated down-
drafts" ;
		NCSDN:units = "#/cm3" ;
		NCSDN:missing_value = -9999.f ;
	float QRSDN(time, z) ;
		QRSDN:long_name = "RAIN in saturated downdrafts" ;
		QRSDN:units = "g/kg" ;
		QRSDN:missing_value = -9999.f ;
	float NRSDN(time, z) ;
		NRSDN:long_name = "RAIN NUMBER CONCENTRATION in saturated downdrafts" ;
		NRSDN:units = "#/cm3" ;
		NRSDN:missing_value = -9999.f ;
	float QISDN(time, z) ;
		QISDN:long_name = "CLOUD ICE in saturated downdrafts" ;
		QISDN:units = "g/kg" ;
		QISDN:missing_value = -9999.f ;
	float NISDN(time, z) ;
		NISDN:long_name = "CLOUD ICE NUMBER CONCENTRATION in saturated down-
drafts" ;
		NISDN:units = "#/cm3" ;
		NISDN:missing_value = -9999.f ;
	float QSSDN(time, z) ;
		QSSDN:long_name = "SNOW in saturated downdrafts" ;
		QSSDN:units = "g/kg" ;
		QSSDN:missing_value = -9999.f ;
	float NSSDN(time, z) ;
		NSSDN:long_name = "SNOW NUMBER CONCENTRATION in saturated downdrafts" ;
		NSSDN:units = "#/cm3" ;
		NSSDN:missing_value = -9999.f ;
	float QGSDN(time, z) ;
		QGSDN:long_name = "GRAUPEL in saturated downdrafts" ;
		QGSDN:units = "g/kg" ;
		QGSDN:missing_value = -9999.f ;
	float NGSDN(time, z) ;
		NGSDN:long_name = "GRAUPEL NUMBER CONCENTRATION in saturated downdrafts" ;
		NGSDN:units = "#/cm3" ;
		NGSDN:missing_value = -9999.f ;
	float QVENV(time, z) ;
		QVENV:long_name = "Water vapor mixing ratio in unsaturated environment" ;
		QVENV:units = "kg/kg" ;
		QVENV:missing_value = -9999.f ;
	float QCENV(time, z) ;
		QCENV:long_name = "Cloud liquid water mixing ratio in unsaturated envi-
ronment" ;
		QCENV:units = "kg/kg" ;
		QCENV:missing_value = -9999.f ;
	float QTOENV(time, z) ;
		QTOENV:long_name = "TOTAL WATER (VAPOR + CLOUD LIQUID) in unsaturated en-
vironment" ;
		QTOENV:units = "g/kg" ;
		QTOENV:missing_value = -9999.f ;
	float NCENV(time, z) ;
		NCENV:long_name = "CLOUD WATER NUMBER CONCENTRATION in unsaturated envi-
ronment" ;
		NCENV:units = "#/cm3" ;
		NCENV:missing_value = -9999.f ;
```

```
       float QRENV(time, z) ;
              QRENV:long_name = "RAIN in unsaturated environment" ;
              QRENV:units = "g/kg" ;
              QRENV:missing_value = -9999.f ;
       float NRENV(time, z) ;
              NRENV:long_name = "RAIN NUMBER CONCENTRATION in unsaturated environment" ;
              NRENV:units = "#/cm3" ;
              NRENV:missing_value = -9999.f ;
       float QIENV(time, z) ;
              QIENV:long_name = "CLOUD ICE in unsaturated environment" ;
              QIENV:units = "g/kg" ;
              QIENV:missing_value = -9999.f ;
       float NIENV(time, z) ;
              NIENV:long_name = "CLOUD ICE NUMBER CONCENTRATION in unsaturated environ-
ment" ;
              NIENV:units = "#/cm3" ;
              NIENV:missing_value = -9999.f ;
       float QSENV(time, z) ;
              QSENV:long_name = "SNOW in unsaturated environment" ;
              QSENV:units = "g/kg" ;
              QSENV:missing_value = -9999.f ;
       float NSENV(time, z) ;
              NSENV:long_name = "SNOW NUMBER CONCENTRATION in unsaturated environment" ;
              NSENV:units = "#/cm3" ;
              NSENV:missing_value = -9999.f ;
       float QGENV(time, z) ;
              QGENV:long_name = "GRAUPEL in unsaturated environment" ;
              QGENV:units = "g/kg" ;
              QGENV:missing_value = -9999.f ;
       float NGENV(time, z) ;
              NGENV:long_name = "GRAUPEL NUMBER CONCENTRATION in unsaturated environ-
ment" ;
              NGENV:units = "#/cm3" ;
              NGENV:missing_value = -9999.f ;

// global attributes:
              :SAM\ version = "" ;
              :caseid = "" ;
}
```

File: .../20160610/sim0100/raw_model//sgp20160610_alpha2.varanal_300km_25mb_ls.varanal_300km_25mb_sf.sonde_init.144_144.2Dcom_1.nc

Description: 2D (X-Y) SAM output averaged between output times. The time label represents the middle of the averaging period.

```
netcdf sgp20160610_alpha2.varanal_300km_25mb_ls.varanal_300km_25mb_sf.sonde_
init.144_144.2Dcom_1 {
dimensions:
       x = 144 ;
       y = 144 ;
       time = UNLIMITED ; // (90 currently)
variables:
       float x(x) ;
              x:units = "m" ;
       float y(y) ;
              y:units = "m" ;
       float time(time) ;
              time:units = "day" ;
              time:long_name = "time" ;
```

```
float Prec(time, y, x) ;
       Prec:long_name = "Surface Precip. Rate" ;
       Prec:units = "mm/day" ;
float LWNS(time, y, x) ;
       LWNS:long_name = "Net LW at the surface" ;
       LWNS:units = "W/m2" ;
float LWNSC(time, y, x) ;
       LWNSC:long_name = "Net clear-sky LW at the surface" ;
       LWNSC:units = "W/m2" ;
float LWNT(time, y, x) ;
       LWNT:long_name = "Net LW at TOA" ;
       LWNT:units = "W/m2" ;
float LWNTC(time, y, x) ;
       LWNTC:long_name = "Clear-Sky Net LW at TOA" ;
       LWNTC:units = "W/m2" ;
float SOLIN(time, y, x) ;
       SOLIN:long_name = "Solar TOA insolation" ;
       SOLIN:units = "W/m2" ;
float SWNS(time, y, x) ;
       SWNS:long_name = "Net SW at the surface" ;
       SWNS:units = "W/m2" ;
float SWNSC(time, y, x) ;
       SWNSC:long_name = "Net Clear-sky SW at the surface" ;
       SWNSC:units = "W/m2" ;
float SWNT(time, y, x) ;
       SWNT:long_name = "Net SW at TOA" ;
       SWNT:units = "W/m2" ;
float SWNTC(time, y, x) ;
       SWNTC:long_name = "Net Clear-Sky SW at TOA" ;
       SWNTC:units = "W/m2" ;
float CWP(time, y, x) ;
       CWP:long_name = "Cloud Water Path" ;
       CWP:units = "mm" ;
float IWP(time, y, x) ;
       IWP:long_name = "Ice Path" ;
       IWP:units = "mm" ;
float CLD(time, y, x) ;
       CLD:long_name = "Cloud Frequency" ;
       CLD:units = "%" ;
float PW(time, y, x) ;
       PW:long_name = "Precipitable Water" ;
       PW:units = "mm" ;
float USFC(time, y, x) ;
       USFC:long_name = "U at the surface" ;
       USFC:units = "m/s" ;
float U200(time, y, x) ;
       U200:long_name = "U at 200 mb" ;
       U200:units = "m/s" ;
float VSFC(time, y, x) ;
       VSFC:long_name = "V at the surface" ;
       VSFC:units = "m/s" ;
float V200(time, y, x) ;
       V200:long_name = "V at 200 mb" ;
       V200:units = "m/s" ;
float W500(time, y, x) ;
       W500:long_name = "W at 500 mb" ;
       W500:units = "m/s" ;
float PSFC(time, y, x) ;
       PSFC:long_name = "P at the surface" ;
       PSFC:units = "mbar" ;
```

```
        float SWVP(time, y, x) ;
                SWVP:long_name = "Saturated Water Vapor Path" ;
                SWVP:units = "mm" ;
        float U850(time, y, x) ;
                U850:long_name = "850 mbar zonal velocity" ;
                U850:units = "m/s" ;
        float V850(time, y, x) ;
                V850:long_name = "850 mbar meridional velocity" ;
                V850:units = "m/s" ;
        float ZC(time, y, x) ;
                ZC:long_name = "Cloud top height (Instantaneous)" ;
                ZC:units = "km" ;
        float TB(time, y, x) ;
                TB:long_name = "Cloud top temperature (Instantaneous)" ;
                TB:units = "K" ;
        float ZE(time, y, x) ;
                ZE:long_name = "Echo top height (Instantaneous)" ;
                ZE:units = "km" ;
}
```

File: .../20160610/sim0100/raw_model/sgp20160610_alpha2.varanal_300km_25mb_ls.varanal_300km_25mb_sf.sonde_init.144_144_0000001200.nc

Description: Instantaneous 3D (X-Y-Z) SAM output. Each file contains one time.

```
netcdf sgp20160610_alpha2.varanal_300km_25mb_ls.varanal_300km_25mb_sf.sonde_
init.144_144_0000001200 {
dimensions:
        x = 144 ;
        y = 144 ;
        z = 226 ;
        time = UNLIMITED ; // (1 currently)
variables:
        float x(x) ;
                x:units = "m" ;
        float y(y) ;
                y:units = "m" ;
        float z(z) ;
                z:units = "m" ;
                z:long_name = "height" ;
        float time(time) ;
                time:units = "d" ;
                time:long_name = "time" ;
        float p(z) ;
                p:units = "mb" ;
                p:long_name = "pressure" ;
        float U(time, z, y, x) ;
                U:long_name = "X Wind Component" ;
                U:units = "m/s       " ;
        float V(time, z, y, x) ;
                V:long_name = "Y Wind Component" ;
                V:units = "m/s       " ;
        float W(time, z, y, x) ;
                W:long_name = "Z Wind Component" ;
                W:units = "m/s       " ;
        float PP(time, z, y, x) ;
                PP:long_name = "Pressure Perturbation" ;
                PP:units = "Pa        " ;
```

```
        float QRAD(time, z, y, x) ;
               QRAD:long_name = "Radiative heating rate" ;
               QRAD:units = "K/day      " ;
        float TABS(time, z, y, x) ;
               TABS:long_name = "Absolute Temperature" ;
               TABS:units = "K          " ;
        float QV(time, z, y, x) ;
               QV:long_name = "Water Vapor" ;
               QV:units = "g/kg       " ;
        float QN(time, z, y, x) ;
               QN:long_name = "Non-precipitating Condensate (Water+Ice) " ;
               QN:units = "g/kg       " ;
        float QP(time, z, y, x) ;
               QP:long_name = "Precipitating Water (Rain+Snow)" ;
               QP:units = "g/kg       " ;
        float NC(time, z, y, x) ;
               NC:long_name = "CLOUD WATER NUMBER CONCENTRATION" ;
               NC:units = "#/cm3      " ;
        float QR(time, z, y, x) ;
               QR:long_name = "RAIN" ;
               QR:units = "g/kg       " ;
        float NR(time, z, y, x) ;
               NR:long_name = "RAIN NUMBER CONCENTRATION" ;
               NR:units = "#/cm3      " ;
        float QI(time, z, y, x) ;
               QI:long_name = "CLOUD ICE" ;
               QI:units = "g/kg       " ;
        float NI(time, z, y, x) ;
               NI:long_name = "CLOUD ICE NUMBER CONCENTRATION" ;
               NI:units = "#/cm3      " ;
        float QS(time, z, y, x) ;
               QS:long_name = "SNOW" ;
               QS:units = "g/kg       " ;
        float NS(time, z, y, x) ;
               NS:long_name = "SNOW NUMBER CONCENTRATION" ;
               NS:units = "#/cm3      " ;
        float QG(time, z, y, x) ;
               QG:long_name = "GRAUPEL" ;
               QG:units = "g/kg       " ;
        float NG(time, z, y, x) ;
               NG:long_name = "GRAUPEL NUMBER CONCENTRATION" ;
               NG:units = "#/cm3      " ;
}
```

Appendix D: Supplemental Cases for 2015

Description of Alpha 2 Supplemental Cases

The LASSO project has released two data sets during the LASSO Pilot Phase. Simulations for 2015 cases have been run as part of the Alpha 1 release and 2016 cases as part of the Alpha 2 release. Evolution of the LASSO product between releases has resulted in the inability to directly host the Alpha 1 data bundles side by side with the Alpha 2 data bundles within the LASSO Bundle Browser (https://adc.arm.gov/lassobrowser) used for data discovery purposes. In addition, several significant bugs have been corrected in the code used to generate the results for Alpha 2. Thus, the five case dates from Alpha 1 have been re-run and released as a supplement to Alpha 2. This makes the 2015 data bundles again accessible within the Bundle Browser and removes the requirement for users to avoid certain variables due to known issues in the Alpha 1 results. The Alpha 1 results have now been largely superseded.

Not all Alpha 1 simulations have been reprocessed for the Alpha 2 supplement. Instead, an ensemble of simulations for each date have been run that match the target ensemble members defined by the range of forcings chosen for LASSO. Plus, we include a comparison of Morrison versus Thompson microphysics with the Weather Research and Forecasting (WRF) model and an initial condition variation alternative with the System for Atmospheric Modeling (SAM). The initial condition variation consists of runs that use the same initial condition profile supplied from the large-scale forcing data, which results in a “self consistent” set of initial conditions and large-scale forcing data. For example, the initial sounding would come from the ECMWF data when using large-scale forcing from ECMWF. This is in contrast to the way most of the LASSO simulations are run with the initial sounding profile coming from the morning radiosonde sounding.

To avoid possible issues with incorrect statistical output calculations with the WRF simulations and an error with an uninitialized variable in the SAM simulations from Alpha 1, it is recommended that users use the newer 2015 runs. Users should contact the LASSO team if there is a need to use the original Alpha 1 simulations so that the specific data used can be verified for its integrity.

Simulation Details

The re-run simulations are for the following days:

6-Jun-2015	27-Jun-2015	29-Aug-2015
9-Jun-2015	1-Aug-2015	

The synoptic conditions associated with each day can be found in Section 6 of the *Description of the LASSO Alpha 1 Release* (Gustafson et al. 2016). Details regarding code changes versus Alpha 1 can be found in Appendix A of the present report with the exception that the radar wind profiler (RWP) data have not been included in the 2015 Multiscale Data Assimilation (MSDA) forcing data.

Each date has the following sets of simulations:

WRF: 7 large-scale forcings * 2 microphysics options + 1 no-large-scale forcing = 15 runs

SAM: 7 large-scale forcings + 6 self-consistent initial conditions = 13 runs

So, the total number of new simulations is 140. The metadata associated with each simulation has been included in an updated spreadsheet listing the available simulations, which can be found at https://www.arm.gov/capabilities/modeling/lasso/SimulationListAlpha2_release2.xlsx. The data bundles associated with the simulations can be downloaded from the Bundle Browser.

Users should be aware that the new 2015 simulations were run on a different computer than the original simulations from Alpha 1. Specifically, the original simulations were run on Eos at the Argonne Leadership Computing Facility, while the new runs were run on ARM's Cumulus cluster. This has resulted in some differences in the simulated cloud fields when doing side-by-side comparisons. Sensitivity tests using single precision (as in the original run) alongside double and single precision but with "precise" floating point settings for the Intel compiler show that the default WRF compiler options for Cray computers with the Intel compilers are not mathematically reliable. However, by turning on the precise flag (specifically, "-fp-model precise"), the single-precision results are very close to those using double precision. Thus, the published simulations have been run with single precision and precise floating point math.

ARM

U.S. DEPARTMENT OF
ENERGY
Office of Science