Office of Satellite and Product Operations
Environmental Satellite Processing Center

JRR Land Surface Temperature System Maintenance Manual

Version 2.2

February 25, 2025

U.S. Department of Commerce
National Oceanic and Atmospheric Administration
National Environmental Satellite, Data, and Information Service
Office of Satellite and Product Operations

Authors

Mark Hardy (OCS)

Updated by Jeffrey Augenbaum (OMS, ERT)

Approval Page

Environmental Satellite Processing Center

JRR Land Surface Temperature System Maintenance Manual

GROUP: OMS Date: 02/26/2025 Jeffrey Augenbaum, Lead Programmer	GROUP: OSPO Date: 03/21/2025 Hanjun Ding, Product Area Lead
GROUP: OMS Date: 03/19/2025 Clay Davenport, Products Manager	

Changes/Revisions Record

This system maintenance manual is changed as required to reflect system, operational, or organizational changes. Modifications made to this document are recorded in the Changes/Revisions Record below. This record will be maintained throughout the life of the document.

Version Number	Date	Description of Change/Revision	Section/Pages Affected	Changes Made by Name/Title/ Organization
1.0	02/2023	Original version	All	Lindeman
2.0	09/2024	Updated version	All	Hardy
2.1	02/2025	Updated for NCCF; Added links to NCCF SMM; Added links to Product monitoring and visualization; Added data flow diagrams	All; 1.3, 2.1, 2.2, 4.2.2, 4.2.3, 4.4.1; 5.2-5.3; 6.1-6.3; 5.1.1; 7.1	Augenbaum
2.2	02/25/2025	Transfer contents to latest template, perform minor copy editing and formatting	All	Hannah Bowie, Technical Writer, ERT
2.2	02/25/2025	Quality Assurance	All	Clint Sherwood, Quality Assurance Manager, ERT

Preface

This document comprises the National Oceanic and Atmospheric Administration (NOAA) National Environmental Satellite, Data, and Information Service (NESDIS), Office of Satellite and Product Operations (OSPO), publication of this JPSS Risk Reduction (JRR) Land Surface Temperature (LST) System Maintenance Manual (SMM). This document reflects current operations for the DOC/NOAA/NESDIS Environmental Satellite Processing Center (ESPC) (NOAA5045) information technology systems. This document describes the established ESPC procedures for JRR LST system maintenance in accordance with Federal, DOC, NOAA, NESDIS and OSPO requirements.

Future updates and revisions to this document will be produced and controlled by DOC/NOAA/NESDIS for ESPC information technology systems.

The published version of this document can be found at the OSPO SharePoint Products Library.

Table of Contents

List of Tables

List of Figures

Executive Summary

This is a System Maintenance Manual (SMM) document describing the JPSS Risk Reduction (JRR) Land Surface Temperature (LST) processing system. These products are included in a single NetCDF output file for the LST algorithm. The algorithms are part of the Joint Polar Satellite System (JPSS) Risk Reduction (RR) system. In this document, 'JPSS RR' is often referred to as 'JRR'.

The Land Surface Temperature system was designed to run within the NESDIS Cloud Common Framework (NCCF) production environment. All of the output product files will be archived at National Centers for Environmental Information (NCEI).

The LST Development product team consists of members from Office of Common Services (OCS). The roles and contact information for the different product team members are identified in Table 0-1.

Table 0-1 - Product Team Members

Team Member	Organization	Role	Contact Information
Walter Wolf	OCS	OCS Product Management Division Chief	walter.wolf@noaa.gov
Yunyue Yu	OCS	Science Team Lead	yunyue.yu@noaa.gov
Yuling Liu	OCS	Science Team	yuling.lie@noaa.gov
Hanjun Ding	OSPO	OSPO PAL	hanjun.ding@noaa.gov
Shuang Qiu	OSPO	OSPO PAL	shuang.qiu@noaa.gov
Changyi Tan	OSPO	OSPO PAL	changyi.tan@noaa.gov
Isabel Zaragosa	OCS	PPM	isabel.zaragosa@noaa.gov
Jianbin Yang	OCS	PIB	jianbing.yang@noaa.gov
Priyanka Roy	OCS	ASSISTT Management Team	priyanka.roy@noaa.gov
Mike Butler	OCS	ASSISTT Framework Integration Team Lead(s)	mike.butler@noaa.gov
Yuxiang He	OCS	ASSISTT Framework Integration Team	yuxiang.he@noaa.gov
Yunhui Zhao	OCS	ASSISTT Configuration Management Team Lead	yunhui.zhao@noaa.gov
Ramaswamy Tiruchirapalli	OCS	ASSISTT R2O Team Lead	ramaswamy.tiruchirapalli@noaa.gov
Eric Buzan	OCS	ASSISTT R2O Team	eric.buzan@noaa.gov
Brandon Laufer	OCS	ASSISTT R2O Team	brandon.laufer@noaa.gov
Mark Hardy	OCS	ASSISTT Technical Writer	mark.hardy@noaa.gov
Tracey Dorian	OCS	ASSISTT Lifecycle QA Team Lead	tracey.dorian@noaa.gov
Mingming Yao	OCS	ASSISTT Lifecycle QA Team	mingming.yao@noaa.gov
Hua Xie	OCS	ASSISTT Science QA Team Lead	hua.xie@noaa.gov

The low-level code within the Land Surface Temperature processing system is written in Fortran 90 and C++. This low-level code performs all data processing, scientific computation, reading/writing, reformatting, and opening/closing of files. All high-level code within the LST processing system is written in Python. The high-level code performs tasks such as file management, system management, making system calls, and error trapping from the lower-level processing. The driver script will manage the software and call any necessary unit scripts. The system is comprised of one unit that handles all of

the LST processing, and another unit which creates offline Land Surface Emissivity (LSE) product which is used in LST processing. The LST unit will, therefore, produce the expected output product file.

Details about the Land Surface Temperature algorithm can be found in its LST Algorithm Theoretical Basis Document (ATBD).

The NESDIS' Policy on Access and Distribution of Environmental Data and Products is provided at: http://www.ospo.noaa.gov/Organization/About/access.html.

The JRR Land Surface Temperature products will be used as a risk reduction assessment for a cost-effective implementation of common NESDIS algorithms for the JRR system. The OCS Algorithm Scientific Software Integration and System Transition Team (ASSISTT) integration and python development teams have prepared for its implementation into the NESDIS Cloud Common Framework (NCCF). LST is run operationally by the Office of Satellite and Product Operations (OSPO) on NCCF for JPSS satellites. The output products are intended for operational and scientific users. Table 0-2 provides information about the algorithms and products.

Table 0-2 - JRR LST Algorithms and Products

Product Category	Algorithm	Product
JRR Land Surface Products	Land Surface Temperature	Land Surface Temperature

1. Introduction

1.1. Product Overview

Land surface temperature (LST) as an important proxy of surface energy is required in a wide variety of applications of hydrology, meteorology and climatology. Remote sensing is a unique approach which allows global and regional coverage for LST. Many algorithms have been proposed to retrieve LST from thermal infrared (TIR) observations, in which temperature coupled with LSE and atmospheric downward radiance. From the perspective of emissivity, these methods could be roughly classified into two types: LST retrieval with known LSE, such as split window (SW) algorithm and single-channel algorithm, and simultaneous temperature and emissivity separation. The latter one requires more temporal or spectral information.

For sensors with only one or two TIR channels, emissivity is an important input and its variation is still the biggest impediment in satellite LST retrieval. Besides, as an intrinsic property of the surface, broad band emissivity is important for the precise determination of longwave radiative energy, and this is particularly necessary for arid/semi-arid regions, where LSE deviate considerably from the behavior of black body. Satellite LST&E have been assimilated into climate, atmospheric and land surface models and high-quality LSE has demonstrated its significant contribution.

The Joint Polar Satellite System (JPSS) and the Geostationary Operational Environmental Satellites R Series (GOES-R) are the nation's next generation of the polar-orbiting environmental satellites and geostationary weather satellites, respectively. The Visible Infrared Radiometer Suite (VIIRS) onboard JPSS series and the Advanced Baseline Imager (ABI) onboard GOES-R will play an important role in developing LST records. The new proposed LSE product will enhance LST production of JPSS and GOES-R missions as well as to support forecasting models. In the United States of America, demands of satellite LSE data are from a variety of government agencies as well as from universities and research institutes.

Accuracy of the satellite LST measurement is limited by the atmospheric correction, the complexity of surface emission characteristics, and sensor performance. Therefore, the performance of LST algorithms depends on the retrieval conditions. It is worth mentioning that LST performance varies significantly over region, season, day/night, dry/moist etc. conditions according to published worldwide LST validation results.

By now satellite LSTs have been routinely produced for over forty years from a variety of polar-orbiting and geostationary satellites. For producing an LST climate data record from those programs, consistency among LST products from different satellite missions are considered for better cross-satellite evaluation and better geographic global validation. A primary objective of the enterprise LST development is to provide a state-of-the-art LST algorithm that is applicable to multiple sensors and has good quality. The enterprise LST algorithm should be comprehensive, robust, and applicable to both geostationary orbit (GEO) and Low Earth orbit (LEO) satellite missions.

Note that the enterprise LST products are available only for cloud clear, probably clear and probably cloudy pixels; and the LUT is stratified for the following conditions: daytime and nighttime, levels of

satellite viewing zenith angles and levels of dry and moist atmospheres. Specifications of the LST product such as resolution, accuracy and refresh rate will be described in the next section.

Currently, surface emissivity variation is still the biggest impediment in satellite LST retrieval. The remote sensing community has been working for years to obtain high quality and time series of global land surface emissivity (LSE). The enterprise LST algorithm should potentially benefit from such technical improvement of emissivity estimation. Concurrent with the enterprise LST development, the global gridded emissivity product is also developed to support the emissivity explicit LST algorithms. For details about the emissivity algorithm, please refer to the emissivity ATBD.

1.2. Algorithm Overview

The processing outline of the LST starts from the initialization module in which all input and output data are initialized. The pixel level processing checks the non-retrieval conditions as a priority: sea water, bad SDR data, cloudy condition, missing input, BT out of range and emissivity out of range. Any pixel with missing input that includes spectral emissivity, total precipitable water, sensor zenith angle, solar zenith angle, cloud mask and land/sea mask will be skipped for LST retrieval. The valid BT range is set as [190,343] and [190,340] for VIIRS band 15 and band 16, respectively based on the SDR ATBD. The emissivity valid range is set to be [0.8, 1.0] consistent to the settings in the emissivity product. Cloudy condition means that the cloud mask indicates the pixel as confidently cloudy. If any one of the non-retrieval conditions is met, the LST will be set as a fill value and the corresponding quality flag will be set accordingly. Note that the gridded emissivity product has been developed for this purpose. The nearest neighbor method is used for mapping the gridded emissivity data into granule pixels.

Following that a valid retrieval will be performed. Before calculating LST for each non-confident-cloudy land pixel, day/night time flag is determined from the solar zenith angle ($\leq$ or $>$ 85º) of the sensor geometric data; and dry/moist atmospheric condition flag is determined using the NCEP water vapor information. LST of the pixel is calculated accordingly with the corresponding algorithm coefficients set. Note that LST will be calculated for in-land snow/ice pixels as well and this is indicated in the quality control flags. Large view angle will be flagged also. Finally, the calculated LST values and their associated quality control flags, which were generated in each of the above steps, are combined with the LST product package and are written to files for user access.

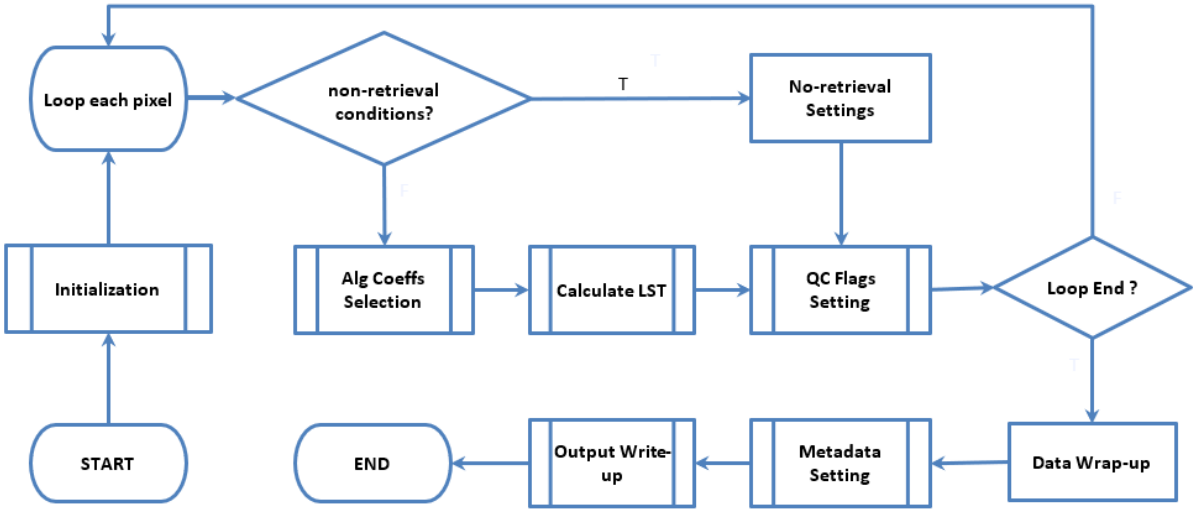

Figure 1-1 - LST Main Processing Steps

Likewise, the LSE product is generated based on the vegetation cover method (VCM), which combines two constant emissivity values of the bare ground and full vegetation situations of each pixel. The real time emissivity is adjusted according to the green vegetation fraction (GVF) and the snow fraction. The processing includes two parts as figure 1-2 shows: the static emissivity module aims to generate background component. Emissivity climatology served as the bare component in the algorithm, and the dynamic emissivity module accounts for the emissivity variation due to GVF and snow fraction.

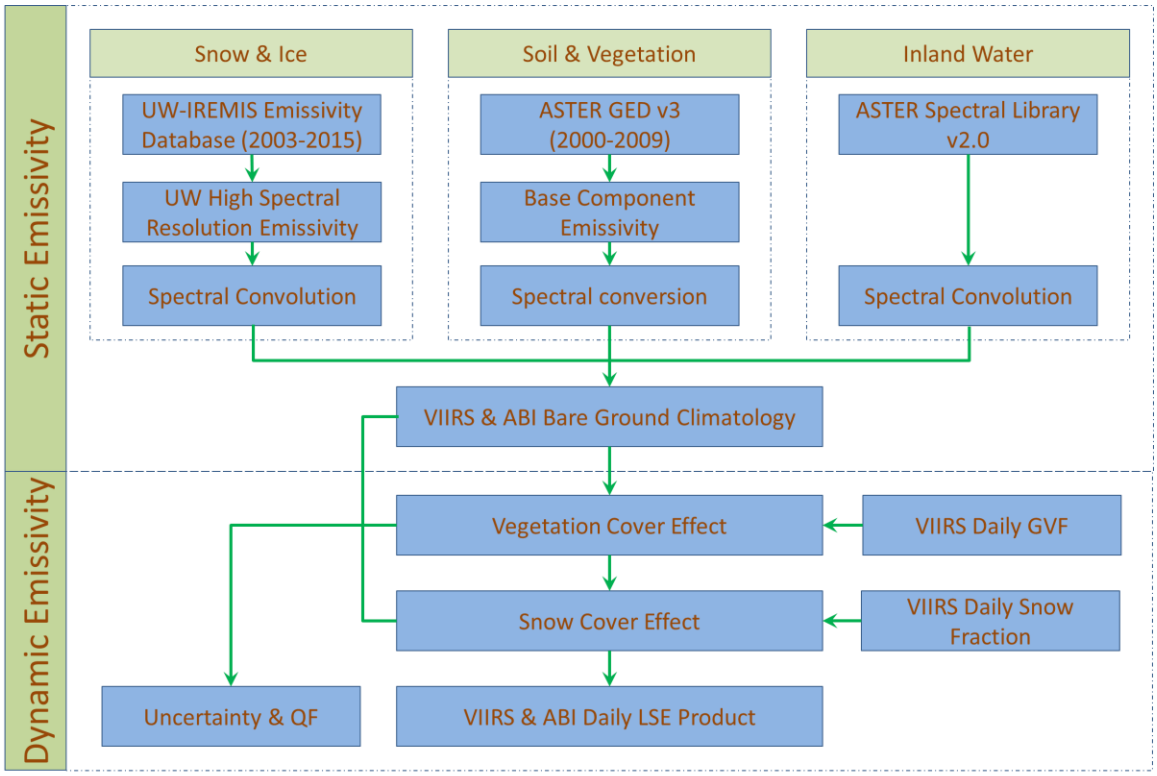

Figure 1-2 - LSE Algorithm Description

In the static emissivity module, land surface is classified into three types with different processing approaches according to their thermal emission characteristics. The permanent snow & ice emissivities are directly converted from the mean value of a long term MODIS emissivity product; the values of the inland water are determined according to Advanced Space borne Thermal Emission and Reflection Radiometer (ASTER) spectral library dataset, and those of soil and vegetation area, also the most common surfaces, are derived from the ASTER Global Emissivity Dataset (GED) dataset and the corresponding mean normalized difference vegetation index (NDVI) using the reverse form of the VCM.

VIIRS/ABI bare ground emissivity climatology, VIIRS GVF and snow fraction are used in the dynamic emissivity module to derive the daily emissivity and its associated quality flags to determine the data quality. The dynamic LSE processing flowchart is demonstrated in Figure 1-3.

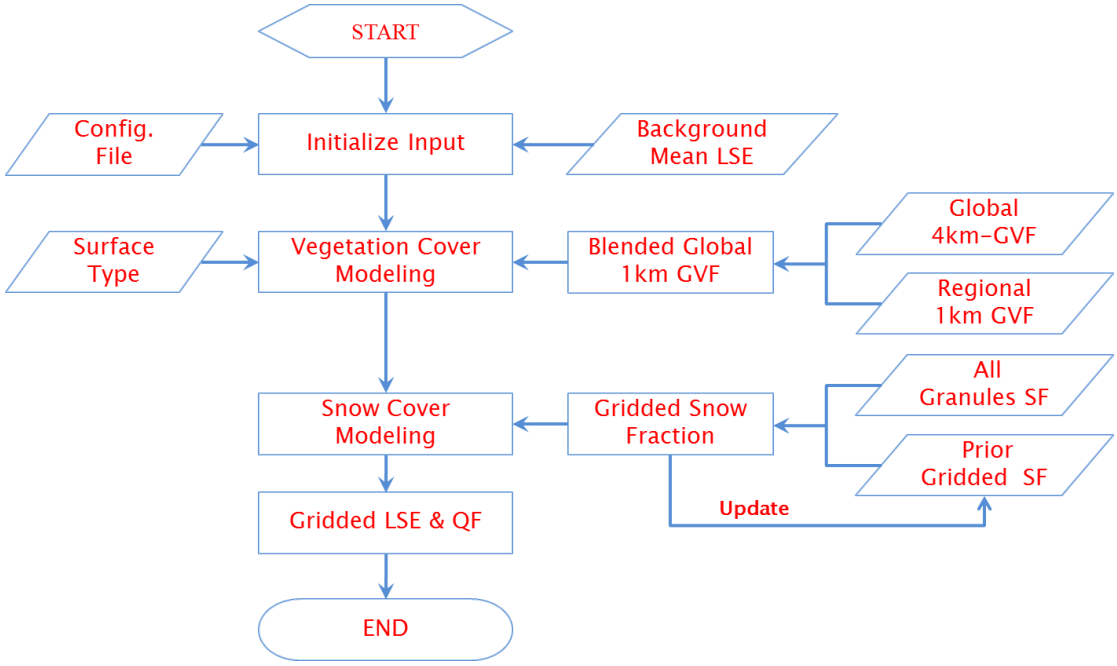

Figure 1-3 - LSE Processing Flowchart

1.3. Interfaces Overview

Before reviewing this System Maintenance Manual (SMM), please request the live master NCCF SMM (refer to *System Maintenance Manual - NESDIS Common Cloud Framework (NCCF)*) from the OSPO PALs in Table 1-2. The NCCF system overview is described in the master NCCF SMM: NCCF Description and Overview (or Document Object: 4,5).

2. Hardware

2.1. Hardware Description

The hardware is described in the master NCCF SMM: NCCF Description and Overview (or Document Object: 4,5): Infrastructure.

2.2. Operating System

The NCCF operating system description can be found in the master NCCF SMM: Operating System (or Document Object: 64).

2.3. System Requirements

The system requirements and timing information are listed in the following table.

Table 2-1 - Land Surface System Requirements

Processing Unit	Required Memory	Required CPUs	Run time
Online LST	3 GB	1 CPU	30 seconds
Offline LSE	3 GB	1 CPU	2-3 hours

2.3.1. Storage Requirements

Unpacking the tar files included with the delivery will require about 8.5 GB of storage space. Sizes of individual files used or created by Offline/Online LSE/LST are shown in Table 2-1. LSE is run once per day and LST is run for every L1b granule (1012-1013 granules per day per satellite) when all L1B and upstream product files are available.

Table 2-2 - Offline LSE and Online LST File Sizes

File	Approximate Size per File
JRR AOD	10 MB
JRR Cloud Mask	33 MB
JRR Snow Cover	23 MB
Offline LSE	320 MB
Online LST	8 MB
L1b VIIRS GMTCO	29 MB
L1b VIIRS SVM [15,16]	~5-10 MB
GVF Global	13 MB
GVF Regional	71 MB
Log	< 5 KB

2.3.2. Computer Resource Requirements

The following libraries and utilities are required by LST:

1. Listing of Files, Libraries, and Off-The-Shelf (OTS) Software Present in the Docker Image
    a. HDF5
    b. netCDF
    c. zlib
    d. szlib
    e. ecCodes
    f. wgrib2
    g. CRTM
    h. Python 3.9 with the following libraries

i. Numpy
j. Urllib3
k. netCDF4
l. h5py
m. pykdtree
n. pyyaml
o. pyresample
p. sortedcontainers

2. Docker Image information
    a. Base Image: assistt-centos7-python3-fw2
    b. Image Size: 1.05 GB
    c. Repository Name: 754153872510.dkr.ecr.us-east-1.amazonaws.com
    d. Tag Name: v4.0

Additionally, the builder requirements are as follows:

- GCC 8.3.0 or Intel 19.0.5 C/C++/Fortran compilers
- Python 3.7+
- CRTM version 2.3.0
- HDF5 version 1.10.5
- NetCDF version 4.7.0 (c)
- NetCDF version 4.4.5 (fortran)
- NetCDF version 4.3.0 (C++)
- zlib version 1.2.11
- Boost version 1.70 (header only)
- ecCodes 2.12.5

These libraries and utilities must be present within the package to allow processing to run successfully. Without these libraries and utilities, the expected output files cannot be produced properly.

2.3.3. Communication Needs

The target system for the Land Surface Temperature package is the NESDIS Common Cloud Framework (NCCF). There are no special bandwidth or communication issues associated with LST. The communication needs of the processing system must be sufficient to meet the processing requirements described throughout this document.

3. Software

3.1. Software Description

There are two main components of JRR LST processing – online LST and offline LSE. This is done because of time constraints – the offline component takes a significant amount of time to execute and is run only once per day. The Offline LSE component generates a dynamic global emissivity map which is used as ancillary data for Online LST. This map is created by the daily offline LSE algorithm

that also produces a daily global gridded Snow Fraction product for subsequent LSE runs. Online LST generates the output product described in this document.

3.2. Directory Description

The CCAP consists of 3 gzip'd tar files:

- JPSSLST_CCAP_v2-1_20240627_CODE.tar.gz
- JPSSLST_CCAP_v2-1_20240627_DATA.tar.gz
- JPSSLST_CCAP_v2-1_20240627_DOCS.tar.gz

One way to unpack the algorithm package involves running the following command:

tar -xvzf < tarfile_name >

where tarfile_name is the name of the tarfile you wish to unpack.

After unpacking the tar files, the following directory trees will appear:

```
├── CODE
│   ├── docker
│   │   ├── Dockerfile
│   │   ├── eccodes_definitions.tgz
│   │   └── fw.tgz
│   ├── LSE_src
│   │   ├── build_alg
│   │   ├── bin
│   │   ├── setEnvLib
│   │   └── source
│   ├── LST_src
│   │   ├── build_alg
│   │   ├── bin
│   │   ├── Config
│   │   ├── doc
│   │   ├── Python
│   │   ├── Setup
│   │   └── src
│   └── scripts
├── DATA
```

```
│   ├── ancillary_data
│   │   ├── algorithm_ancillary
│   │   ├── framework_ancillary
│   │   └── LSE
│   ├── sample_yamls
│   └── test_cases
│     ├── gfs
│     ├── logs
│     ├── LSE_NOAA20
│     ├── LSE_NOAA21
│     ├── LSE_NPP
│     ├── LST_NOAA20
│     ├── LST_NOAA21
│     ├── LST_NPP
│     └── output
└── DOCS
      ├── ATBD_EPS_Land_LSE_v1.2.pdf
      ├── EnterpriseAlgorithm_ATBD_LST_V1.3_nov2021.docx
      ├── JPSS-LST_v2-1_DeliveryMemo_20240627.docx
      ├── JPSS-LST_v2-1_ProductionRules_20240627.docx
      └── JPSS-LST_v2-1_README_20240627.docx
```

Note it will require about 46 GB to store the unpacked data.

Docker Information:

1. Files, Libraries, and Off-The-Shelf (OTS) Software Present in the Docker Image
    a. HDF5
    b. netCDF
    c. zlib
    d. szlib
    e. ecCodes
    f. wgrib2
    g. CRTM
    h. Python 3.9 with the following libraries
        i. Numpy
        ii. Urllib3

iii. netCDF4
iv. h5py
v. pykdtree
vi. pyyaml
vii. pyresample
viii. sortedcontainers

2. Docker Image information
    a. Base Image: assistt-centos7-python3-fw2
    b. Image Size: 1.05 GB
    c. Repository Name: 754153872510.dkr.ecr.us-east-1.amazonaws.com
    d. Tag Name: v4.0

3.3. Source Code Description

The source code associated with the Land Surface Temperature system is similar to the source code that supports the JPSS Risk Reduction (RR) system. The framework processing system is currently the AIT-Framework system version 2, or the GOES-R Algorithm Working Group (AWG) Product Processing System Framework. The JPSS RR system contains many programs, functions, subroutines, and scripts. These files are written multiple coding languages including Fortran 77/90/95, C/C++, and Python. The source code can be found in the Framework Algorithms subdirectory.

Due to the size and content of the processing system, details about the source code are not described in this document.

4. Normal Operations

4.1. System Control

Several YAML and python scripts are responsible for the execution of both Offline LSE and Online LST. The scripts begin the process by setting up the parameters, deciding what gets processed, and runs the algorithm.

4.1.1. System Control Files

Online LST processing and offline LSE processing are performed by running the following command from the scripts directory:

- python launch_docker.py /path/to/docker_config.yaml

This script launches a Docker container then executes either LST processing via the script jpssrr_application.py or LSE processing via the script run_offline_lse.py.

4.1.2. Processing Controls

There must be at least one YAML file present in the algorithm package that contains information concerning the setup of the Docker run command. For convenience, all items in the file that have a possibility of variation or are system specific will be located at the top of the YAML file as anchors. Each of these arguments are listed below.

Table 4-1 - YAML Argument Listings

Key	Description
application_yaml	Full path of the application YAML
log_dir	Full path of the directory where log files are placed at the end of processing
output_dir	Full path of the directory where output files are placed at the end of processing
scripts_dir	Full path of the 'scripts' directory included in the package
lse_exe	Full path to the LSE executable offline_lse.exe
gvf_dir	Full path to the directory containing GVF files
snow_cover_dir	Full path to the directory containing FSC files
rolling_snow_dir	Full path to the directory containing Rolling Snow Cover files
ancillary_dir	Full path of the ancillary data directory included in the package
memory	Memory allocated to the Docker container. '4GB' is recommended.
docker_image	Full path/name of the Docker image
sat	These values are only used to generate a name for the Docker container and are not used by the algorithm. Any value may be used, but the following are suggested values: • sat - 'multi' • project - 'trap' • caseid - The runtime of the job

Two configuration files in YAML format are used for LST and LSE processing each (four YAML files total). One contains information used to launch the Docker container, and the other contains information to run the processing.

LSE Docker YAML Files

The Docker config YAML is organized so that all directories are specified in the "arguments" section at the top of the file. Some directories are used for other JPSS algorithms besides LST and have a fixed value of "none" for this package. This file contains the following entries:

- version – Set to 1.0
- arguments
    - scripts_dir – Location of the scripts directory included in this package
    - application_yaml – Full path to the application config YAML, described below
    - output_dir – Directory where Aerosols product files are saved
    - log_dir – Directory where log files are saved
    - gvf_dir - Directory containing global and regional GVF files. Set to "none" if both files are not available.
    - prev_snow - Full path to the previous day's Rolling Snow Fraction file. Set to "none" if not available.

    - snow_dir - Directory containing the previous day's Fractional Snow Cover products
    - ancillary_dir - Location of the offline_lse_ancillary directory included with this package
- docker_info:
    - docker_args:
        - memory: Memory allocated to the Docker container. At least 3 GB is required.
        - ulimit:
            - stack: Set to -1
        - cpus: set to 1
- container_name – The values below are only used to construct the container name.
    - sat: Satellite ID, such as J01, NPP, or N21. A specific value is not required
    - proj: Recommended to be "lse" or a similar value
    - caseid: Recommended to be the start timestamp of the granule being processed
- image: Full name of the Docker image being used
- python_info – Do not modify this section
    - python: python
    - script: /home/WORKING_DIR/ccap/scripts/jpssrr_application.py
    - application_yaml: /home/WORKING_DIR/app.yaml
- mount_pairs – Do not modify this section

Example: **LSE_NPP_docker_info.yaml**

version: 1.0

arguments:

- &scripts_dir /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/scripts

- &application_yaml /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/sample_yamls/LSE_NPP_application_info.yaml

- &lse_exe /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/offline_src/OFFLINE-LAND_LSE_VIIRS/bin/offline_LSE.exe

- &output_dir /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/test_data/output

- &logs_dir /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/test_data/logs

- &gvf_dir /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/test_data/lse_npp_20200420/GVF

- &prev_snow /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/test_data/lse_npp_20200420/S-NPP.VIIRS.Daily.Rolling.SnowFraction.20200419.nc

- &snow_dir /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/test_data/lse_npp_20200420/snow

- &ancillary_dir /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/ancillary/offline_lse_ancillary

```
  - &sat          NPP
  - &caseid       d20200420t000000b
docker_info:
  image: git.star.nesdis.noaa.gov:4567/assistt/docker_image/assistt-centos7-python3-fw2:v4.0
  python_info:
    python: python
    script: /home/WORKING_DIR/scripts/run_offline_lse.py
    application_yaml: /home/WORKING_DIR/app.yaml
  mount_pairs:
    - src: *scripts_dir
      dst: /home/WORKING_DIR/scripts
    - src: *application_yaml
      dst: /home/WORKING_DIR/app.yaml
    - src: *lse_exe
      dst: /home/WORKING_DIR/offline_LSE.exe
    - src: *output_dir
      dst: /home/WORKING_DIR/output
    - src: *logs_dir
      dst: /home/WORKING_DIR/logs
    - src: *ancillary_dir
      dst: /home/WORKING_DIR/ancillary
    - src: *gvf_dir
      dst: /home/WORKING_DIR/gvf
    - src: *prev_snow
      dst: /home/WORKING_DIR/Daily.Rolling.SnowFraction.nc
    - src: *snow_dir
      dst: /home/WORKING_DIR/snow
  docker_args:
    memory: 4Gb
    tmpfs: /home/WORKING_DIR/work/:exec
  container_name:
```

sat: *sat

proj: LSE

caseid: *caseid

LSE Application YAML Files

The application config YAML is organized so that settings that are routinely modified by the end user are specified in the "arguments" section at the top of the file. The application yaml contains several paths that are internal to the Docker container and should not be modified. In addition, it contains config settings passed to the LSE executable that do not need to be modified under normal usage.

- arguments
    - logging_level - Desired logging level: DEBUG, INFO, WARNING, or ERROR
    - date - Date of the data being processed in YYYYMMDD format
    - satellite - One of NOAA20, NOAA21, or NPP

Example: **LST_NOAA20_application_info.yaml**

production_site: 'T4'

production_environment: 'DEV'

satellite: 'NOAA20'

gfs_pattern: gfs.t@AN@z.pgrb2.0p25.f0@FF@.@YYYY@@MM@@DD@

use_ims_ssmi_snow_mask: false

output_products:

- LAND_LST_VIIRS

working_dir: /home/WORKING_DIR

l1b_dir: /home/WORKING_DIR/l1b

fw_input_dir: /home/WORKING_DIR/input

fw_output_dir: /home/WORKING_DIR/output

cmc_sst_dir: /home/WORKING_DIR/cmc_sst

snow_mask_dir: /home/WORKING_DIR/snow_mask

sea_ice_dir: /home/WORKING_DIR/sea_ice

gfs_dir: /home/WORKING_DIR/gfs

algorithm_ancillary_dir: /home/WORKING_DIR/algorithm_ancillary

framework_ancillary_dir: /home/WORKING_DIR/framework_ancillary

framework_exe: /home/WORKING_DIR/SOURCE/algorithms.exe

framework_config_dir: /home/WORKING_DIR/SOURCE/Config

eccodes_dir: /usr/share/assistt/ots/eccodes/definitions

outdir: /home/WORKING_DIR/docker_output

logdir: /home/WORKING_DIR/docker_logs

lse_file: /home/WORKING_DIR/lse_file.nc

LST Docker YAML Files

The Docker config YAML is organized so that all directories are specified in the "arguments" section at the top of the file. Some directories are used for other JPSS algorithms besides LST and have a fixed value of "none" for this package. This file contains the following entries:

- version – Set to 1.0
- arguments
    - application_yaml – Full path to the application config YAML, described below
    - log_dir – Directory where log files are saved
    - output_dir – Directory where Aerosols product files are saved
    - scripts_dir – Location of the scripts directory included in this package
    - algorithm_ancillary – Location of the algorithm ancillary directory included with this package
    - framework_ancillary – Location of the Framework ancillary directory included with this package
    - gfs_dir – Directory containing GFS files
    - cmc_sst_dir – Set to "none", used for other algorithms
    - snow_mask_dir – Set to "none", used for other algorithms
    - sea_ice_dir – Set to "none", used for other algorithms
    - l1b_dir – Directory containing input L1B data
    - upstream_dir – Directory containing product files from upstream algorithms (Cloud Mask, AOD, Fractional Snow Cover)
    - framework_exe – Location of the Framework executable algorithms.exe
    - framework_config_dir – Location of the Framework Config directory
    - lse_file - Daily LSE file generated by offline LSE processing
- docker_info:
    - docker_args:
        - memory: Memory allocated to the Docker container. At least 3 GB is required.
        - ulimit:
            - stack: Set to -1
        - cpus: set to 1
    - container_name – The values below are only used to construct the container name.
        - sat: Satellite ID, such as J01, NPP, or N21. A specific value is not required
        - proj: Recommended to be "jpss-lst" or a similar value
        - caseid: Recommended to be the start timestamp of the granule being processed
    - image: Full name of the Docker image being used
    - python_info – Do not modify this section
        - python: python
        - script: /home/WORKING_DIR/ccap/scripts/jpssrr_application.py
        - application_yaml: /home/WORKING_DIR/app.yaml

- mount_pairs – Do not modify this section

Example: **LST_NOAA20_docker_info.yaml**

version: 1.0

arguments:

- &application_yaml /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/sample_yamls/LST_NOAA20_application_info.yaml

- &log_dir /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/test_data/logs

- &output_dir /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/test_data/output

- &scripts_dir /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/scripts

- &algorithm_ancillary /data/data312/michael.butler/DAP_JRR_NDEv3r2/AIT_TEST_DATA/algorithm_ancillary

- &framework_ancillary /data/data312/michael.butler/DAP_JRR_NDEv3r2/AIT_TEST_DATA/framework_ancillary

- &gfs_dir /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/test_data/lst_j01_DAY_d20191105_t0506557/gfs

- &cmc_sst_dir none

- &snow_mask_dir none

- &sea_ice_dir none

- &l1b_dir /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/test_data/lst_j01_DAY_d20191105_t0506557/l1b

- &upstream_dir /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/test_data/lst_j01_DAY_d20191105_t0506557/input

- &framework_exe /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/framework/algorithms.exe

- &framework_config_dir /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/framework/Config

- &lse_file /data/data047/eric.buzan/working/JPSS_DAPs/package_lst/test_data/lst_j01_DAY_d20191105_t0506557/NOAA20.VIIRS.LSE.20191103.v2p1.nc

docker_info:

docker_args:

```
  memory: 4GB
  ulimit:
    stack: -1
  cpus: 1
container_name:
  sat: J01
  proj: jpssrr-lst
  caseid: d20191105t0506557
image: git.star.nesdis.noaa.gov:4567/assistt/docker_image/assistt-centos7-python3-fw2:v4.0
python_info:
  python: python
  script: /home/WORKING_DIR/scripts/jpssrr_application.py
  application_yaml: /home/WORKING_DIR/app.yaml
mount_pairs:
  - src: *application_yaml
    dst: /home/WORKING_DIR/app.yaml
  - src: *log_dir
    dst: /home/WORKING_DIR/docker_logs
  - src: *output_dir
    dst: /home/WORKING_DIR/docker_output
  - src: *scripts_dir
    dst: /home/WORKING_DIR/scripts
  - src: *algorithm_ancillary
    dst: /home/WORKING_DIR/algorithm_ancillary
  - src: *framework_ancillary
    dst: /home/WORKING_DIR/framework_ancillary
  - src: *framework_exe
    dst: /home/WORKING_DIR/SOURCE/algorithms.exe
  - src: *framework_config_dir
    dst: /home/WORKING_DIR/SOURCE/Config
  - src: *gfs_dir
```

dst: /home/WORKING_DIR/gfs

- src: *cmc_sst_dir

dst: /home/WORKING_DIR/cmc_sst

- src: *snow_mask_dir

dst: /home/WORKING_DIR/snow_mask

- src: *sea_ice_dir

dst: /home/WORKING_DIR/sea_ice

- src: *l1b_dir

dst: /home/WORKING_DIR/l1b

- src: *upstream_dir

dst: /home/WORKING_DIR/input

- src: *lse_file

dst: /home/WORKING_DIR/lse_file.nc

LST Application YAML Files

The application yaml contains several paths that are internal to the Docker container and should not be modified, along with some values used for other JPSS algorithms that should be set to "false". The application config YAML contains the following entries:

- production_site – Production Site included in the output file metadata
- production_environment – Production environment included in the output file metadata
- satellite – One of the following values depending on the satellite being processed:
    - J01, J1, NOAA-20, NOAA20
    - J02, J2, NOAA-21, NOAA21
    - NPP, SNPP, S-NPP
- gfs_pattern: Allows the naming scheme of the GFS files to be customized. The following special values are used in the name:
    - @AN@ - Model time: 00, 06, 12, or 18
    - @FF@ - 2 digit forecast time
    - @YYYY@ - Year
    - @MM@ - Month
    - @DD@ - Day

The default and recommended value is
"gfs.t@AN@z.pgrb2.0p25.f0@FF@.@YYYY@@MM@@DD@"

- use_ims_ssmi_snow_mask – Set to false
- output products: A list used to set the desired output products. Set to "LAND_LST_VIIRS".
- working_dir – Do not modify
- l1b_dir – Do not modify
- fw_input_dir – Do not modify

- fw_output_dir – Do not modify
- cmc_sst_dir – Do not modify
- snow_mask_dir – Do not modify
- sea_ice_dir – Do not modify
- gfs_dir – Do not modify
- algorithm_ancillary_dir – Do not modify
- framework_ancillary_dir – Do not modify
- framework_exe – Do not modify
- framework_config_dir – Do not modify
- eccodes_dir – Do not modify
- outdir: do not modify
- logdir: do not modify

Example: **LST_NOAA20_application_info.yaml**

production_site: 'T4'

production_environment: 'DEV'

satellite: 'NOAA20'

gfs_pattern: gfs.t@AN@z.pgrb2.0p25.f0@FF@.@YYYY@@MM@@DD@

use_ims_ssmi_snow_mask: false

output_products:

- LAND_LST_VIIRS

working_dir: /home/WORKING_DIR

l1b_dir: /home/WORKING_DIR/l1b

fw_input_dir: /home/WORKING_DIR/input

fw_output_dir: /home/WORKING_DIR/output

cmc_sst_dir: /home/WORKING_DIR/cmc_sst

snow_mask_dir: /home/WORKING_DIR/snow_mask

sea_ice_dir: /home/WORKING_DIR/sea_ice

gfs_dir: /home/WORKING_DIR/gfs

algorithm_ancillary_dir: /home/WORKING_DIR/algorithm_ancillary

framework_ancillary_dir: /home/WORKING_DIR/framework_ancillary

framework_exe: /home/WORKING_DIR/SOURCE/algorithms.exe

framework_config_dir: /home/WORKING_DIR/SOURCE/Config

eccodes_dir: /usr/share/assistt/ots/eccodes/definitions

outdir: /home/WORKING_DIR/docker_output

logdir: /home/WORKING_DIR/docker_logs

lse_file: /home/WORKING_DIR/lse_file.nc

4.2. Installation

4.2.1. Installation Items

For more information concerning the installation items associated with the LST processing system, please refer to section 3.2.

4.2.2. Compilation Procedures

1. Ensure that GCC 8.3 or the Intel 19.0.5 compilers are available in your PATH.
2. Set the location of the required libraries using environment variables. By default, OTS_HOME should be set to the root directory of these libraries, and each library is contained in the following subdirectories:
    a. zlib - $OTS_HOME
    b. szip = $OTS_HOME
    c. hdf5 - $OTS_HOME/hdf5
    d. netCDF - $OTS_HOME/netcdf4
    e. Jasper - $OTS_HOME/jasper
    f. ecCodes - $OTS_HOME/eccodes
    g. CRTM - $OTS_HOME/crtm_v2.3.0
    h. Boost - $OTS_HOME/boost

    The location of each library can be customized with the variables ZLIB_HOME, SZIP_HOME, HDF5_HOME, NETCDF4_HOME, JASPER_HOME, ECCODES_HOME, CRTM_HOME, or BOOST_HOME.
3. Enter the $dap_root/framework directory and run the following commands to build the Framework executable:
    a. ./build_alg clean
    b. ./build_alg -j12
4. Enter the $root/CODE/offline_lse_src directory and run the following commands to build the Framework executable:
    a. ./build_alg clean
    b. ./build_alg

    If compilation is successful, algorithms.exe will be present in the CODE/framework directory, and offline_LSE.exe will be present in the CODE/offline_lse_src/bin directory.

4.2.3. Installation Procedures

4.3. Configuration Procedures

Please refer to the README document included with this delivery package.

See the master NCCF SMM: Installation Procedures (or Document Object: 77).

4.3.1. Production Rules

There must be at least one YAML file present in the algorithm package responsible for guiding the inner script as it chooses what processing or preprocessing steps will occur, which granule will be processed, and which span of time will be observed. For convenience, all items in the file that have a possibility of variation or are system specific will be located at the top of the YAML file as anchors. Each of these arguments are listed below.

4.4. Operations Procedures

4.4.1. Normal Operations

Please refer to the master NCCF SMM: Procedures for Normal Operations (or Document Object 10).

4.5. Distribution

4.5.1. Data Transfer/Communications

Please refer to the master NCCF SMM: Data Transfer/Communications and Data Preparation (or Document Object 73, 81).

4.5.2. Distribution Restrictions

There are no restrictions on the distribution of LST products.

4.5.3. Product Retention Requirements

No specific requirement for this product.

4.5.4. External Product Tools

There are no external product tools supplied with the LST package. The LST output files are all in NetCDF4 format. External users can choose their own tools to display and analyze these output files.

5. Monitoring and Maintenance

5.1. Job Monitoring

5.1.1. Product Monitoring and Visualization

Product quality is monitored using the NCCF Product Monitoring Tool at https://nccf.espc.nesdis.noaa.gov/mtool/index.html.

Users can use this page to monitor summaries of the LST quality based on parameter thresholds determined by the PAL.

The NCCF Products Visualization Page is located at https://origin-east-01-www-ospo.woc.noaa.gov/products/land/lst/.

LST products are generated daily.

5.2. Data Signal Monitoring

5.3. Product Monitoring

5.3.1. Unit Test Plans

5.3.2. Internal Product Tools

There are no internal product tools supplied with the LST processing system. The output product files are NetCDF4 files. External users can choose their own tools to display and analyze these output files.

5.3.3. Performance Statistics

5.3.4. Product Monitoring

5.3.5. Product Criticality

5.4. Maintenance

5.4.1. Monitoring and Maintenance

See the master NCCF SMM: Maintenance Utilities (or Document Object: 84).

5.4.1.1. Ingest Monitoring

See the master NCCF SMM: Data Transfer/Communications and Data Preparation (or Document Object: 73).

5.4.1.2. Production Job Monitoring

5.4.1.3. Product Distribution Monitoring

5.4.2. Science Maintenance

Product quality monitoring is performed by the OSPO Product Quality Monitoring System and the OCS developers. OCS and OSPO personnel communicate regularly to discuss any potential data quality issues, formulate updates to the code, and schedule updates to the package's science code.

5.4.3. Library Maintenance

See the master NCCF SMM: Library Maintenance (or Document Object: 71).

5.4.4. Special Maintenance Procedures

No special maintenance procedures are required for the algorithm.

See the master NCCF SMM: Special Maintenance Procedures (or Document Object: 72).

5.4.5. Maintenance Utilities

5.5. Program Backup Procedures

See the master NCCF SMM: Data Recovery Procedures and Program Recovery Procedures (or Document Object: 89, 90).

6. Troubleshooting

6.1. Program Diagnosis and Recovery

See the master NCCF SMM: Problem Diagnosis and Recovery Procedures (or Document Object: 82).

6.1.1. Quality Control Output

Quality flags are expected to be zero, which means no error. Each failure is associated with a unique "flag" value that is saved in the Land Surface Temperature output file. The output file has a number of failure codes. Table 6-1 describes the control codes of the quality flags for the LST product. The Quality Control (QC) variable names and corresponding data types are also listed in the captions.

Table 6-1 - LST Quality Flags (LST_Quality_Flag output file variable)

bit	Flag	Source	Description
1-0	LST quality	LST	00=high quality 10=medium 01=low 11=no retrieval due to missing data (other than SDR) or ocean pixel
3-2	Cloud condition	Cloud Mask	00=confidently clear 01=probably clear 10=probably cloudy 11=confidently cloudy
4	SDR quality	SDR	0=normal, 1=out of space, missing or bad SDR data
5	Aerosol Optical Thickness at 550 nm (slant path)	AOD	0=within range (aod<=1.0) 1=outside range (aod>1.0)
7-6	Land surface cover	Land/sea mask snow/ice mask	00=land 01=snow/ice 10=inland water/shoreline 11=coastal/seawater
9-8	Water vapor condition	TPW Input	00=very dry (wv<1.5g/cm2) 01=dry (1.5g/cm2<=wv<3.0g/cm2) 10=moist (3.0g/cm2<=wv<4.5g/cm2) 11=very moist (wv>=4.5g/cm2)
10	Emissivity quality	Emissivity	0=emissivity uncertainty within range 1=out of range
11	Degradation by large viewing angle	SDR	0=no degradation 1=large view degradation (LZA>40deg).
12	Day/night flag	SDR	0=nighttime (SolZen>85) 1=daytime (SolZen<=85)
13	Thin cirrus	Cloud Mask	thin cirrus 0=no, 1=yes
14	Fire contamination flag	Cloud Mask	0=no, 1=yes
15	Reserved for future use		

The LST output file contains a quality flag variable for the Offline LSE processing results called Emis_Quality_Flag, where: Bit 0-1: Overall Quality: 00=mean error 0-0.005 01=mean error 0.005-0.010 10=mean error 0.010-0.015 11=mean error >0.015. Bit 2-3: Surface Type: 00=Land 01=Permanent snow/ice 10=Ocean 11=Inland water. Bit 4: GVF: 0=Original 1km GVF 1=Resampled 1km GVF from 4km data. Bit 5: Snow: 0=Instantaneous VIIRS snow fraction 1=non-instantaneous snow fraction. Bit 6-7: For future use.

6.1.2. Error Correction

See the master NCCF SMM: Error Correction – Warnings and Messages for Systems and Error

Codes, Menus and Navigation (or Document Object: 43,44,45).

6.1.3. Problem Diagnosis and Recovery Procedures

See the master NCCF SMM: Problem Diagnosis and Recovery Procedures (or Document Object: 82).

6.1.3.1. High-Level Errors

6.1.3.2. Low-Level Errors

6.1.4. Data Recovery Procedures

See the master NCCF SMM: Data Recovery Procedures (or Document Object: 89).

6.1.5. Program Recovery Procedures

6.2. Application Shutdown and Restart

See the master NCCF SMM: Program Recovery Procedures (or Document Object: 90).

6.2.1. Application Shutdown Procedures

See the master NCCF SMM: Application Shutdown Procedures (or Document Object: 94).

6.2.2. Application Restart Procedures

See the master NCCF SMM: Application Restart Procedures (or Document Object: 92).

6.3. System Shutdown and Restart

See the master NCCF SMM: Reboot Procedures, Restart Procedures and Shutdown Procedures (or Document Object: 83, 93, 95).

6.3.1. System Shutdown Procedures

6.3.2. System Restart Procedures

6.3.3. System Reboot Procedures

7. Appendix

7.1. Data Flow

The various levels of the LST data flow are presented in the figure, and sections, below. Section 7.2 presents the various types of input data files needed for processing. Section 7.3 shows the system level ancillary files used in the calculation of the products. A description of the output files is shown in Section 7.5 for the various file formats.

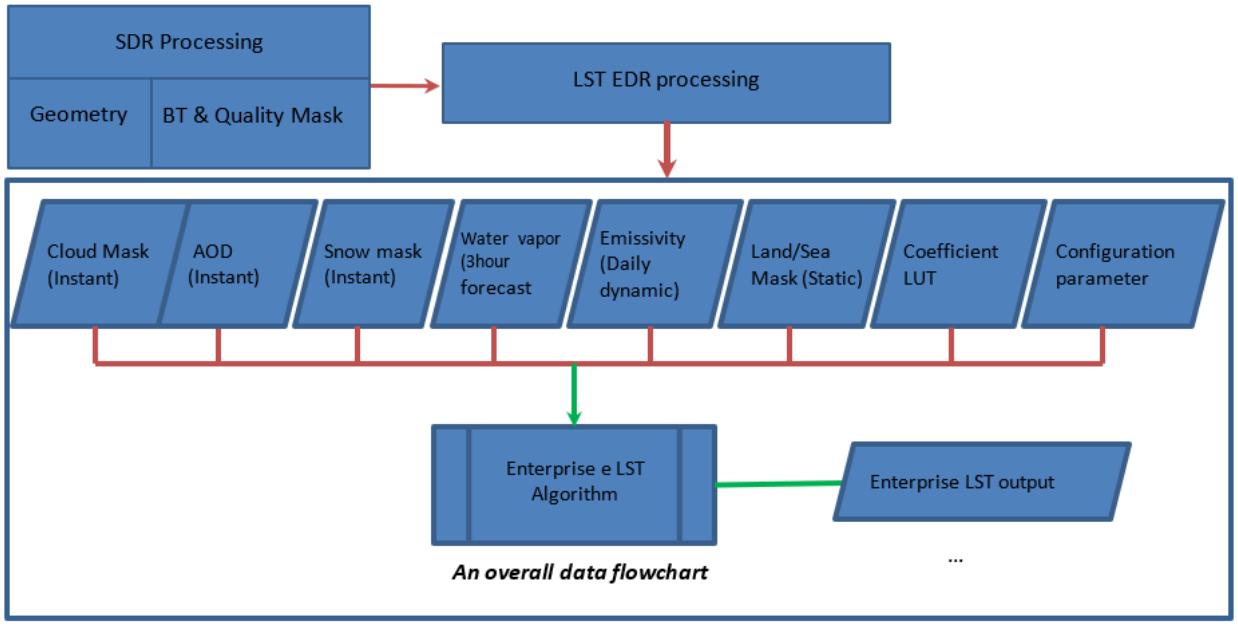

Figure 7-1 - High level data flow of the enterprise LST algorithm

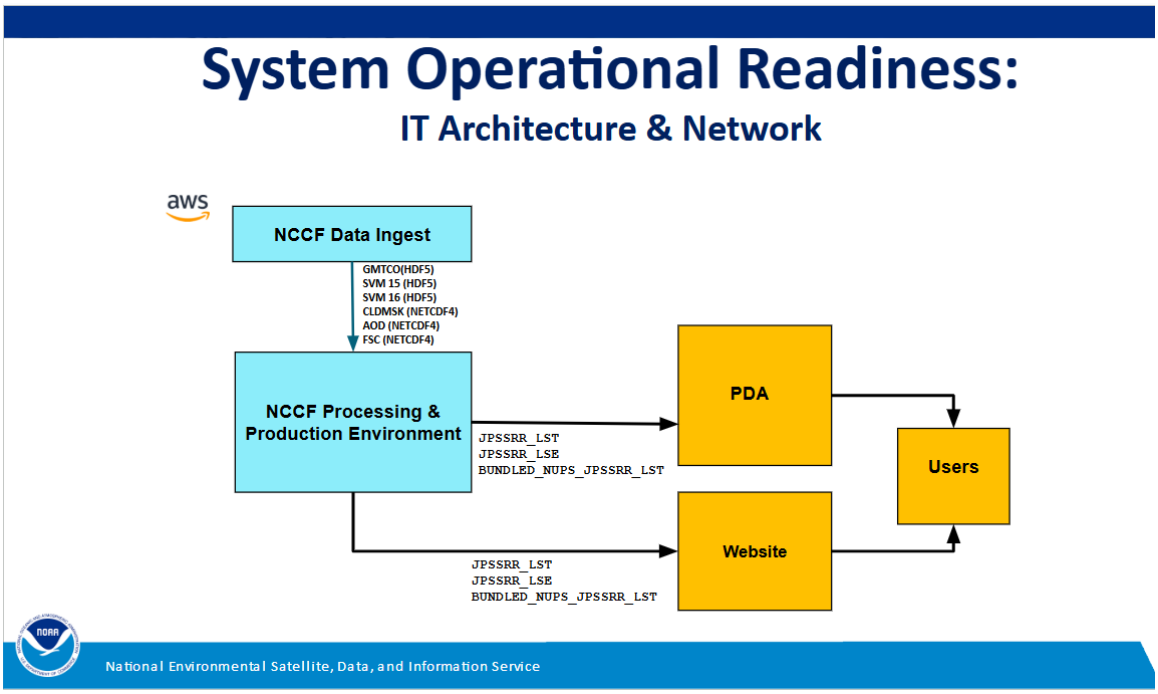

Figure 7-2 - IT Architecture & Network

The enterprise LST product is based on a split-window technique that corrects for atmospheric absorption, and applies surface emissivity explicitly in the retrieval. Coefficients of the LST algorithm, which were derived using an atmospheric radiative transfer model (RTM), are stratified by daytime and nighttime conditions, multiple levels of viewing geometry and dry and moist atmospheric conditions. The algorithm is then verified using an RTM simulation dataset and evaluated using S-NPP VIIRS dataset and ground measurements.

7.2. Input Data Files

The input data sources differ between the Offline LSE and Online LST components. The Offline output files are technically intermediate files, as their purpose is to provide the input for Online LST. This section lists the necessary input files for LSE and LST.

Online LST

The JRR Land Surface Temperature processing system ingests three M-Band channels - GMTCO and SVM [15-16] data files captured by the VIIRS instrument (see Table 7-1).

Table 7-1 - LST Input Satellite Data

File Handle Name in PCF	File Name
GMTCO, SVM15, SVM16	<band>_<sat>_d<YYYYmmdd>_t<HHMMSSs>_ e<HHMMSSs>_b<orbit>_c<YYYYmmddHHMMSSssssss>_<source>.h5
Cloud Mask. AOD, FSC	JRR-<algorithm>_<version>_<sat>_s<YYYYmmddHHMMSSs>_ e<YYYYmmddHHMMSSs> _c<YYYYmmddHHMMSSs>.nc
GFS Data	gfs.t<AN>z.pgrb2.0p25.f0<FF>.<YYYYmmdd>

Where:

<band>	→	GMTCO, SVM15, or SVM16
<algorithm>	→	CloudMask, AOD, SnowCover
<version>	→	Filename version (current version for the JRR files is v3r2 and for LST is v2r2)
<sat>	→	Satellite (j01, j02*, n21**, or npp) *for daily LSE and L1b **for Cloud Mask, AOD, FSC, and LST
s<YYYYmmddHHMMSSs>	→	Start date and time of the granule with four-digit year, two-digit month, two-digit day, two-digit hour, two digit second, and one digit tenth of a second
e<YYYYmmddHHMMSSs>	→	End date and time of the granule with four-digit year, two-digit month, two-digit day, two-digit hour, two digit second, and one digit tenth of a second
c<YYYYmmddHHMMSSs>	→	Creation date and time of the granule with four-digit year, two-digit month, two-digit day, two-digit hour, two digit second, and one digit tenth of a second
d<YYYYmmdd>	→	Start date with four-digit year, two-digit month, and two-digit day
t<HHMMSSs>	→	Start time of the granule with two-digit hour, two digit second, and one digit tenth of a second
e<HHMMSSs>	→	End time of the granule with two-digit hour, two digit second, and one digit tenth of a second
b<orbit>	→	Satellite orbit number
<source>	→	Source of the file, including, but not limited to, noac_ops or oeac_ops
<AN>	→	GFS Model run time (00, 06, 12, or 18)
<FF>	→	GFS Model forecast time (03, 06, or 09)
<YYYYmmdd>	→	GFS Model run date with four-digit year, two-digit month, two-digit day

Land Surface Temperature also relies on corresponding products from the NOAA JPSS Risk Reduction System. These products include the Cloud Mask, Aerosol Optical Depth (AOD), and Snow Cover. Note that AOD is only used as input in the daytime.

Offline LSE

The following satellite data is required or optional to perform offline LSE processing:

- Fractional Snow Cover (FSC) in netCDF format. All FSC output for the corresponding satellite from the previous day should be provided. The expected format of the filename is as follows:

Table 7-2 - LSE Input Satellite Data

File Handle Name in PCF	File Name
JRR Snow Cover	JRR-SnowCover_<version>_<sat>_s<YYYYmmddHHMMSSs>_ e<YYYYmmddHHMMSSs> _c<YYYYmmddHHMMSSs>.nc
Snow Fraction	<sat_sf>.VIIRS.Daily.Rolling.SnowFraction.<YYYYmmdd>.nc
GVF Weekly	GVF-WKL-<loc>_<version>_<sat>_s<YYYYmmddHHMMSSs>_ e<YYYYmmddHHMMSSs>_c<YYYYmmddHHMMSSs>.nc

Where:

<version>	→	Filename version (current version is v3r2 for JRR Snow Cover, v3r0 for GVF Weekly, v2r2 for Daily LSE)
<sat>	→	Satellite (j01, n21, or npp)
<sat_sf>	→	Satellite (NOAA20, NOAA21, S-NPP)
s<YYYYmmddHHMMSSs>	→	Start date and time of the granule with four-digit year, two-digit month, two-digit day, two-digit hour, two digit second, and one digit tenth of a second
e<YYYYmmddHHMMSSs>	→	End date and time of the granule with four-digit year, two-digit month, two-digit day, two-digit hour, two digit second, and one digit tenth of a second
c<YYYYmmddHHMMSSs>	→	Creation date and time of the granule with four-digit year, two-digit month, two-digit day, two-digit hour, two digit second, and one digit tenth of a second

7.3. Ancillary Data Files

Similar to the input files, the dynamic ancillary files differ between Offline LSE and Online LST. This section lists the necessary ancillary files for LSE and LST.

Static and dynamic ancillary files are in NetCDF format. Static ancillary data includes data sets such as the land mask, elevation, surface type, etc, and are included with the algorithm product package. Dynamic data includes GFS model data at 0.25-degree resolution, and daily Land Surface Emissivity

(LSE) data generated by the offline LSE product system. The second and main unit generates the Land Surface Temperature (LST) product.

The static ancillary data needed to generate the LSE products is located in the following directory:

- DATA/ancillary_data/LSE

The static ancillary data needed to generate the LST products are located in the following directories:

- DATA/ancillary_data/algorithm_ancillary
- DATA/ancillary_data/framework_ancillary

Two grib2-formatted files with specific model and forecast time are required. The file name convention of the unformatted GFS files can be seen in Table 7-3.

Table 7-3 - Land Surface Dynamic Ancillary Data Files

File Type	File Name Convention
GFS File	gfs.t<AN>z.pgrb2.0p25.f0<FF>.<YYYYmmdd>
Daily LSE File	<sat_d>.VIIRS.LSE.<YYYYmmdd>.<version>.nc
GVF Weekly	GVF-WKL-<loc>_<version>_<sat>_s<YYYYmmddHHMMSSs>_ e<YYYYmmddHHMMSSs>_c<YYYYmmddHHMMSSs>.nc

Where:

<AN>	→	Model run time, either 00, 06, 12, or 18
<FF>	→	Model forecase time, either 03, 06, or 09
<YYYYMMDD>	→	the date in 4-digit year, 2-digit month, and 2-digit day format
<version>	→	Filename version (current version is v3r2 for JRR Snow Cover, v3r0 for GVF Weekly, v2r2 for Daily LSE)
<sat_d>	→	Satellite (NOAA20, NOAA21, SNPP)
<yyyymmdd>	→	Start date with four-digit year, two-digit month, and two-digit day
<loc>	→	Location (GLB or REG)
<version>	→	Filename version (current version is v3r2 for JRR Snow Cover, v3r0 for GVF Weekly, v2r2 for Daily LSE)

<sat>	→	Satellite (j01, n21, or npp)
s<YYYYmmddHHMMSSs>	→	Start date and time of the granule with four-digit year, two-digit month, two-digit day, two-digit hour, two digit second, and one digit tenth of a second
e<YYYYmmddHHMMSSs>	→	End date and time of the granule with four-digit year, two-digit month, two-digit day, two-digit hour, two digit second, and one digit tenth of a second
c<YYYYmmddHHMMSSs>	→	Creation date and time of the granule with four-digit year, two-digit month, two-digit day, two-digit hour, two-digit second, and one-digit tenth of a second

The specific GFS files needed for LST processing are based on the start hour of the granule. The table below lists the required times for each hour. YYYYMMDD corresponds to the date of the granule.

Table 7-4 - LST Granule Item List

Granule start time hour	Model hour	Forecast hours	Example GFS Files
00, 01, 02	18	06, 09	gfs.t18z.pgrb2.0p25.f006.YYYYMMDD* gfs.t18z.pgrb2.0p25.f009.YYYYMMDD* *These files are from the previous day.
03, 04, 05	00	03, 06	gfs.t00z.pgrb2.0p25.f003.YYYYMMDD gfs.t00z.pgrb2.0p25.f006.YYYYMMDD
06, 07, 08	00	06, 09	gfs.t00z.pgrb2.0p25.f006.YYYYMMDD gfs.t00z.pgrb2.0p25.f009.YYYYMMDD
09, 10, 11	06	03, 06	gfs.t06z.pgrb2.0p25.f003.YYYYMMDD gfs.t06z.pgrb2.0p25.f006.YYYYMMDD
12, 13, 14	06	06, 09	gfs.t06z.pgrb2.0p25.f006.YYYYMMDD gfs.t06z.pgrb2.0p25.f009.YYYYMMDD
15, 16, 17	12	03, 06	gfs.t12z.pgrb2.0p25.f003.YYYYMMDD gfs.t12z.pgrb2.0p25.f006.YYYYMMDD
18, 19, 20	12	06, 09	gfs.t12z.pgrb2.0p25.f006.YYYYMMDD gfs.t12z.pgrb2.0p25.f009.YYYYMMDD
21, 22, 23	18	03, 06	gfs.t18z.pgrb2.0p25.f003.YYYYMMDD gfs.t18z.pgrb2.0p25.f006.YYYYMMDD

If these GFS files are not available, files with a model time 6 hours prior and a forecast time 6 hours later may be used. For example, if gfs.t06z.pgrb2.0p25.f006.20200108 is not available, gfs.t00z.pgrb2.0p25.f012.20200108 may be substituted. GFS files from 6, 12, or 18 hours prior using this pattern can be used.

7.4. Look Up Tables (Static Ancillary Data)

Additional static ancillary data for the algorithm and software processing is included with the delivery package.

7.5. Intermediate Data Set Description

LSE produces two intermediate files: the Daily LSE product that is passed on to the LST unit, and an updated Daily Gridded Snow Fraction file that is used as an input for the following day. The files produced will follow this file name format:

Table 7-5 - Land Surface Intermediate File Product Naming Convention

File	Naming Convention
Daily LSE File	<sat>.VIIRS.LSE.<YYYYmmdd>.<version>.nc

Where:

<version>	→	Filename version (current version is v3r2 for JRR Snow Cover, v3r0 for GVF Weekly, v2r2 for Daily LSE)
<sat_d>	→	Satellite (NOAA20, NOAA21, SNPP)
<yyyymmdd>	→	Start date with four-digit year, two-digit month, and two digit day

7.6. Output Data Set Description

The Land Surface Temperature output file contains the estimation of the surface temperature on land. This file will be archived at NCEI. The LST products use VIIRS L1B data for granules from the S-NPP, NOAA-20, and NOAA-21 satellites.

The products are in NetCDF format and undergo compression while being processed. Table 7-6 lists the LST output file and its format.

Table 7-6 - Land Surface Temperature Output File Naming Convention

Type of File	Naming Convention
Land Surface Temperature	LST_<version>_<sat>_s<YYYYmmddHHMMSSs>_e<YYYYmmddHHMMSSs> _c<YYYYmmddHHMMSSs>.nc

Where:

<version>	→	Filename version (current version for the JRR files is v3r2 and for LST is v2r2)

<sat>	→	Satellite (j01, j02*, n21**, or npp) *for daily LSE and L1b **for Cloud Mask, AOD, FSC, and LST
s<YYYYmmddHHMMSSs>	→	Start date and time of the granule with four-digit year, two-digit month, two-digit day, two-digit hour, two digit second, and one digit tenth of a second
e<YYYYmmddHHMMSSs>	→	End date and time of the granule with four-digit year, two-digit month, two-digit day, two-digit hour, two digit second, and one digit tenth of a second
c<YYYYmmddHHMMSSs>	→	Creation date and time of the granule with four-digit year, two-digit month, two-digit day, two-digit hour, two digit second, and one digit tenth of a second
<YYYYMMDD>>	→	Date with year/month/day

Table 7-7 - LST Output File Metadata

Variable	Type	Description	Dim	Units	Range
AOD_Availability	Short	AOD Availability check: 0=no, 1=yes	0	1	N/A
DataQualityFlag	Byte	VIIRS LST 2-bits Quality Flag: 00-high quality; 10-medium quality; 01-low quality; 11-no retrieval	2	1	0,3
Emis_Quality_Flag**	Byte	VIIRS Emissivity Product Quality Flag	2	1	0,63
LST	short	VIIRS Land Surface Temperature (LST = LST data * LST_ScaleFact + LST_Offset)	2	Kelvin	2600,28600
LST_Err	byte	Land Surface Temperature Uncertainty	2	Kelvin	
LST_Quality_Flag*	short	LST Quality Flag	2	1	-32768,-1
Latitude	Float	Pixel latitude in field latitude	2	Degrees north	-90, 90.
Longitude	Float	Pixel longitude in field longitude	2	Degrees east	-180, 180.

Variable	Type	Description	Dim	Units	Range
Max_LST	Float	Maximum LST within valid LST range	0	K	213.0, 343.0
Mean_LST	Float	Mean LST within valid LST range	0	K	213.0, 343.0
Min_LST	Float	Minimum LST within valid LST range	0	K	213.0, 343.0
Percent_Clear	Float	Percent of pixels under confidently clear or probably clear	0	percent	0.0, 100.0
Percent_High_Quality	Float	Percent of high quality retrievals	0	percent	0.0, 100.0
Percent_Invalid	Float	Percent of pixels out of valid LST range	0	percent	0.0, 100.0
Percent_Land	Float	Percent of land pixels	0	percent	0.0, 100.0
Percent_Large_Angle	Float	Percent of pixels with large view angle	0	percent	0.0, 100.0
Percent_Low_Quality	Float	Percent of low quality retrievals	0	percent	0.0, 100.0
Percent_Medium_Quality	Float	Percent of medium quality retrievals	0	percent	0.0, 100.0
Percent_Retrieved	Float	Percent of pixels with LST retrieval	0	percent	0.0, 100.0
Satellite_Azimuth_Angle	Byte	Scaled Satellite Azimuth Angle (Azi = Azi_Data * AZI_Scalefact + AZI_Offset)	2	Degrees	-120, 120
Satellite_Zenith_Angle	Byte	Satellite Zenith Angle	2	Degrees	0, 90
StartColumn	Integer	Start column index	0	1	N/A
StartRow	Integer	Start row index	0	1	N/A
StdDev_LST	Float	Standard deviation of LST within valid LST range	0	Kelvin	N/A
emis_bbe	Byte	Broadband Emissivity between 8.0 and 13.5 micron (LSE = LSE data * LSE_ScaleFact + LSE_Offset)	2	1	-100, 100

Variable	Type	Description	Dim	Units	Range
emis_m15	Byte	VIIRS Band M15 Emissivity (LSE = LSE data * LSE_ScaleFact + LSE_Offset)	2	1	-100, 100
emis_m16	Byte	VIIRS Band M16 Emissivity (LSE = LSE data * LSE_ScaleFact + LSE_Offset)	2	1	-100, 100
quality_information	N/A	Total number of retrievals, percentage of optimal retrievals, percentage of sub-obtimal_retrievals, percentage of bad retrievals	1	1	N/A

Table 7-8 - LST Output File Metadata

Attribute	Description	Type	Array Size
_NCProperties	NetCDF and HDF version numbers, will be automatically generated	String	Scalar
Conventions	Conventions used here	String	Scalar
Metadata_Conventions	Metadata conventions used here	String	Scalar
Metadata_Link	Contains a URL where detailed metadata or a product information page is located	String	Scalar
ascend_descend_data_flag	Flag indicate whether satellite ascending or descending	Integer	Scalar
cdm_data_type	States the geographic category the product represents	String	Scalar
creator_email	Includes an email for the algorithm development team.	String	Scalar
creator_name	This attribute should include STAR and the name of the algorithm team responsible for development of the product	String	Scalar
creator_url	All product projects should have a website that users can access.	String	Scalar
date_created	UTC time the product file was created in 4-digit year, 2-digit month, 2-digit day, 2-digit hour, 2-digit minute, 2-digit second format	String	Scalar
day_night_data_flag	flag which indicates whether it is day or night	String	Scalar
end_orbit_number	The ending number of the satellite orbit	Integeer	Scalar
geospatial_bounds	Describes the shape and bounding corner locations of the domain	string	1
geospatial_first_scanline_first_fov_lat	The first latitude at the first scanline	float	scalar
geospatial_first_scanline_first_fov_lon	The first longitude at the first scanline	float	scalar

Attribute	Description	Type	Array Size
geospatial_first_scanline_last_fov_lat	The last latitude at the first scanline	float	scalar
geospatial_first_scanline_last_fov_lon	The last longitude at the first scanline	float	scalar
geospatial_last_scanline_first_fov_lat	The first latitude at the last scanline	float	scalar
geospatial_last_scanline_first_fov_lon	The first longitude at the last scanline	float	scalar
geospatial_last_scanline_last_fov_lat	The last latitude at the last scanline	float	scalar
geospatial_last_scanline_last_fov_lon	The last longitude at the last scanline	float	scalar
geospatial_lat_max	Describes a simple upper latitude limit; may be part of a 2- or 3-dimensional bounding region. Geospatial_lat_max specifies the northernmost latitude covered by the dataset.	32-bit floating-point	1
geospatial_lat_min	Describes a simple lower latitude limit; may be part of a 2- or 3-dimensional bounding region. Geospatial_lat_min specifies the southernmost latitude covered by the dataset.	32-bit floating-point	1
geospatial_lat_resolution	Information about the targeted spacing of points in latitude. Recommend describing resolution as a number value followed by the units. Examples: '100 meters', '0.1 degree'	String	Scalar
geospatial_lat_units	Indicates unit associated with geospatial latitude	String	Scalar
geospatial_lon_max	Describes a simple longitude limit; may be part of a 2- or 3-dimensional bounding region. geospatial_lon_max specifies the easternmost longitude covered by the dataset.	32-bit floating-point	1
geospatial_lon_min	Describes a simple longitude limit; may be part of a 2- or 3-dimensional bounding region. geospatial_lon_min specifies the westernmost longitude covered by the dataset.	32-bit floating-point	1
geospatial_lon_resolution	Information about the targeted spacing of points in longitude. Recommend describing resolution as a number value followed by units. Examples: '100 meters', '0.1 degree'	String	Scalar
geospatial_lon_units	Indicates unit associated with geospatial longitude	String	Scalar
history	Indicates algorithm name and version responsible for creating the file	String	Scalar
history_package	The delivery package version number	string	scalar
id	Unique identifier for the product	String	Scalar
institution	Indicates institution responsible for product file	String	Scalar
instrument	Name of the relevant satellite instrument	string	scalar

Attribute	Description	Type	Array Size
keywords	A comma-separated list of key words and/or phrases. Keywords may be common words or phrases, terms from a controlled vocabulary (GCMD is often used), or URIs for terms from a controlled vocabulary (see also 'keywords_vocabulary' attribute).	String	Scalar
metadata_link	A URL that gives the location of more complete metadata or a product information web page. A persistent URL is recommended for this attribute. If the only available metadata is contained in this file, then the name of this file is sufficient.	String	Scalar
naming_authority	Organization responsible for providing the "id" attribute	String	Scalar
platform	This attribute should reflect the satellite(s) used to derive the product	String	Scalar
processing_level	Level of processing associated with product file	String	Scalar
production_environment	Processing string responsible for generating the product	String	Scalar
production_site	Processing site for the product	String	Scalar
project	Indicates the name(s) of the project(s) responsible for generating the original data used as input to the processing system	String	Scalar
publisher_email	This attribute should be the OSPO ESPC Help Desk email "espcoperations@noaa.gov." or The email address of the person (or other entity specified by the publisher_type attribute) responsible for publishing the data file or product to users, with its current metadata and format.	string	Scalar
publisher_name	This attribute should be "DOC/NOAA/NESDIS/ESPDS > ESPDS, NESDIS, NOAA, U.S. Department of Commerce." or The name of the person (or other entity specified by the publisher_type attribute) responsible for publishing the data file or product to users, with its'' current metadata and format.	string	scalar
publisher_url	This attribute should be the OSPO Web site "http://www.ospo.noaa.gov." or The URL of the person (or other entity specified by the publisher_type attribute) responsible for publishing the data file or product to users	String	Scalar
references	Contact info	String	Scalar

Attribute	Description	Type	Array Size
resolution	horizontal resolution	string	Scalar
satellite_name	Name of the satellite	string	scalar
sensor_band_central_radiation_wavelength	Central wavelength of the satellite instrument	string	1
sensor_band_identifier	Band number of the satellite instrument	string	1
source	This attribute should list all major input files as a comma delimited list	string	Scalar
standard_name_vocabulary	Provides the name and corresponding version number of the controlled vocabulary used	String	Scalar
start_orbit_number	The starting number of the satellite orbit	integer	scalar
summary	Provides a brief summary of the product	String	Scalar
time_coverage_end	Indicates the end time of the observation associated with the file in 4-digit year, 2-digit month, 2-digit day, 2-digit hour, 2-digit minute, 2-digit second format	String	Scalar
time_coverage_start	Indicates the start time of the observation associated with the file in 4-digit year, 2-digit month, 2-digit day, 2-digit hour, 2-digit minute, 2-digit second format	String	Scalar
title	Provides the short name for the product	String	Scalar

7.7. Archive Data Files

The NetCDF output files for LST will be archived at NCEI.

7.8. References

NESDIS/STAR (2023), README and setup and installation text files included with the JRR Land Surface Temperature CCAP (Cloud Containerized Algorithm Package)

NESDIS/STAR (2023), The JRR Land Surface Temperature External Users Manual

NESDIS/STAR (2022), The NOAA JPSS Risk Reduction (JPSSRR) System - External Users Manual (EUM) v3.1

NESDIS/STAR (2022), The NOAA JPSS Risk Reduction (JPSSRR) System - System Maintenance Manual (SMM) v3.1

NESDIS/STAR (2022), FY23 Enterprise Algorithm Project Plan. NLR Land & Surface Hydrology: Land Surface Temperature (LST L2) / Land Surface Emissivity (LSE L2)

END OF DOCUMENT

8. Acronyms

Acronym	Definition
ABI	Advanced Baseline Imager
AOD	Aerosol Optical Depth
ASSISTT	Algorithm Scientific Software Integration and System Transition Team
ASTER	Advanced Space borne Thermal Emission and Reflection Radiometer
ATBD	Algorithm Theoretical Basis Document
AVHRR	Advanced Very High Resolution Radiometer
AWG	Algorithm Working Group
DMSP	Defense Meteorological Satellite Program
DNB	Day-Night Band
DOA	Department of Agriculture
DOC	Department of Commerce
DOD	Department of Defense
DOI	Department of the Interior
EPA	Environmental Protection Agency
ERT	Earth Resources Technology, Inc.
ESPC	Environmental Satellite Processing Center
EUM	External Users' Manual
FSC	Fractional Snow Cover
GED	Global Emissivity Dataset
GEO	Geostationary Orbit
GMTCO	Terrain Corrected Geolocation
GOES-R	Geostationary Operational Environmental Satellites R Series
GVF	Green Vegetation Fraction
JPSS	Joint Polar Satellite System
JPSSRR	JPSS Risk Reduction
JRR	JPSS Risk Reduction
LEO	Low Earth Orbit
LSE	Land Surface Emissivity
LST	Land Surface Temperature
MODIS	Moderate Resolution Imaging Spectroradiometer
NCCF	NESDIS Common Cloud Framework
NCEI	National Centers for Environmental Information
NDVI	Normalized Difference Vegetation Index
NESDIS	National Environmental Satellite, Data, and Information Service
NetCDF	Network Common Data Form
NOAA	National Oceanic and Atmospheric Administration
NWP	Numerical Weather Prediction
OCS	Office of Common Services
OLS	Operational Linescan System
OMS	Operations, Maintenance, and Sustainment
OSPO	Office of Satellite and Product Operations
PAL	Product Area Lead
PDA	Product Distribution and Access
PIB	Product Implementation Branch
PPM	Project Portfolio Management
QC	Quality Control
RAD	Requirements Allocation Document
SDR	Science Data Record

Acronym	Definition
SMM	System Maintenance Manual
S-NPP	Suomi National Polar-orbiting Partnership
SVM	Support Vector Machines
SW	Split Window
TIR	Thermal Infrared
V&V	Verification and Validation
VIIRS	Visible Infrared Imaging Radiometer Suite