NOAA-Unique VIIRS Polar Winds Product System

Algorithm Readiness Review

March 27, 2014

Prepared By: Walter Wolf[1], Jeff Key[1], Jaime Daniels[1], Hongming Qi[2], Shanna Sampson[3], Aiwu Li[3], Tom King[3], and Larisa Koval[3]

[1] NOAA/NESDIS/STAR
[2] NOAA/NESDIS/OSPO
[3] IMSG

Review Agenda

• Introduction	9:00 am – 9:15 am	Wolf
• Risks	9:15 am – 9:30 am	Wolf
• Requirements	9:30 am – 9:45 am	Wolf
• Algorithm Readiness	9:45 am – 10:05 am	Daniels
• Software Architecture	10:05 am – 11:05 am	Li
• Delivered Algorithm Package	11:05 am – 11:15 am	Li
• Risks and Actions	11:15 am – 11:20 am	Wolf
• Summary and Conclusions	11:20 am – 11:25 am	Wolf
• Appendix – VIIRS Polar Winds Unit Tests		

Agenda

- Introduction
- Risks
- Requirements
- Algorithm Readiness
- Software Architecture
- Delivered Algorithm Package
- Risks and Actions
- Summary and Conclusions
- Appendix – VIIRS Polar Winds Unit Tests



Introduction

Presented by

Walt er Wolf

Introduction

- Project Background
  - » Terminology
  - » IJPS
  - » NPP/JPSS
  - » VIIRS
  - » Capabilities/Impacts
- Project Objectives
- Project Stakeholders
- Project Plan
- Project Risk Management

Terminology

- Sensor Data Records (SDR, used by the Winds Team)
  - » Data record produced when an algorithm is used to convert RDR to geolocated, calibrated detected fluxes with associated ephemeris data. Calibration, ephemeris, and any other ancillary data necessary to convert the sensor units back to sensor raw data (counts) are included. For ATMS these are averaged to a CrIS field-of-regard.
- Environmental Data Records (EDR, used by the winds team)
  - » Data record produced when an algorithm is used to convert SDR to geophysical parameters (including ancillary parameters, e.g., cloud clear radiation, etc.).

IJPS

- IJPS -- Initial Joint Polar-Orbiting Operational Satellite System
  - » Mission: It is a cooperative effort between NOAA and EUMETSAT to provide and improve the operational meteorological and environmental forecasting and global climate monitoring services worldwide.

- IJPS is started with NOAA N, covers the MetOp series, and continues with S-NPP and the JPSS series.

S-NPP/JPSS

- S-NPP and JPSS, a joint NOAA/NASA effort, is the next series of polar-orbiting satellites dedicated to among other things, operational meteorology. The objective of the JPSS mission is to ensure continuity, improvement and availability of operational observations from an afternoon polar orbit (13:30 pm).

- NPP is the first of three missions with launch dates of 2011, ≈2017, ≈ 2021, respectively.

VIIRS Description

- VIIRS: Visible/Infrared Imager Radiometer Suite
  - » Combines the radiometric accuracy of the Advanced Very High Resolution Radiometer (AVHRR) currently flown on the NOAA polar orbiters with the high (0.65 kilometer) spatial resolution of the Operational Linescan System (OLS) flown on DMSP.
  - » 22 channels (0.3 – 14 microns)

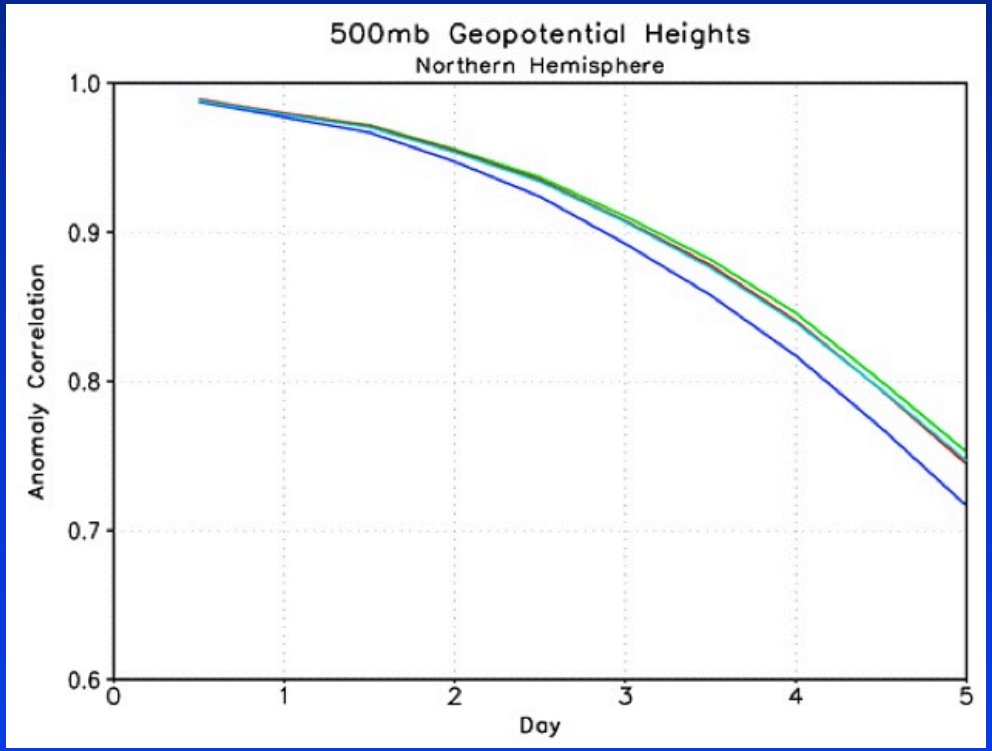
VPW Objectives

- Provide VIIRS Polar Winds (VPW) NOAA Unique products within three hours of observation (or 30 minutes of data receipt from IDPS) to NWS and DOD:
  - » Polar Winds derived from Channel M15 (10.763um)
  - » Product Quality Flags (Expected Error, QI)
  - » Metadata
  - » Diagnostics
- Provide NOAA Unique VPW Products to CLASS.

Capabilities & Impacts

- Polar wind data add skill to weather forecasts, not just locally but globally.

500 hPa anomaly correlation coefficients as a function of forecast time for a summer 2004 experiment, Northern hemisphere (20 to 80 deg lat, left panel. Control (no MODIS winds) in blue, Control + MODIS WV winds in aqua color, Control + MODIS IR winds in red (overlapping aqua), Control + all MODIS winds in green.



- As with AVHRR, VIIRS IR (cloud track) winds are expected to have a positive impact on forecasts.

Project Stakeholders – Customers and Users

- U.S. Users:
  - NCEP (Dennis Keyser)
  - GMAO/JCSDA (Lars Peter Riishojgaard)
  - NRL/FNMOC (Randy Pauley)
  - STAR (Jaime Daniels)
  - CLASS (John Bates)
- Foreign Users:
  - UK Met Office (Mary Forsythe)
  - JMA (Masahiro Kazumori)
  - ECMWF (Jean-Noel Thepaut)
  - DWD (Alexandar Cress)
  - Meteo-France (Bruno Lacroix)
  - CMC (Real Sarrazin)
  - BOM (John LeMarshall)
  - EUMETSAT (Simon Elliott)
  - Russian Hydrometcenter (Mikhail Tsyrulnikov)

Project Stakeholders – Operations and Maintenance

- **NDE – NPOESS Data Exploitation (Rick Vizbulis)**
- **NDE System Development Team (Dylan Powell)**
- **STAR-NDE Liason (Pryanka Roy)**
- **OSPO (H. Qi)**
  - » Run the VIIRS Polar Winds product processing system at OSPO after delivery
  - » Distribute the Winds data and products

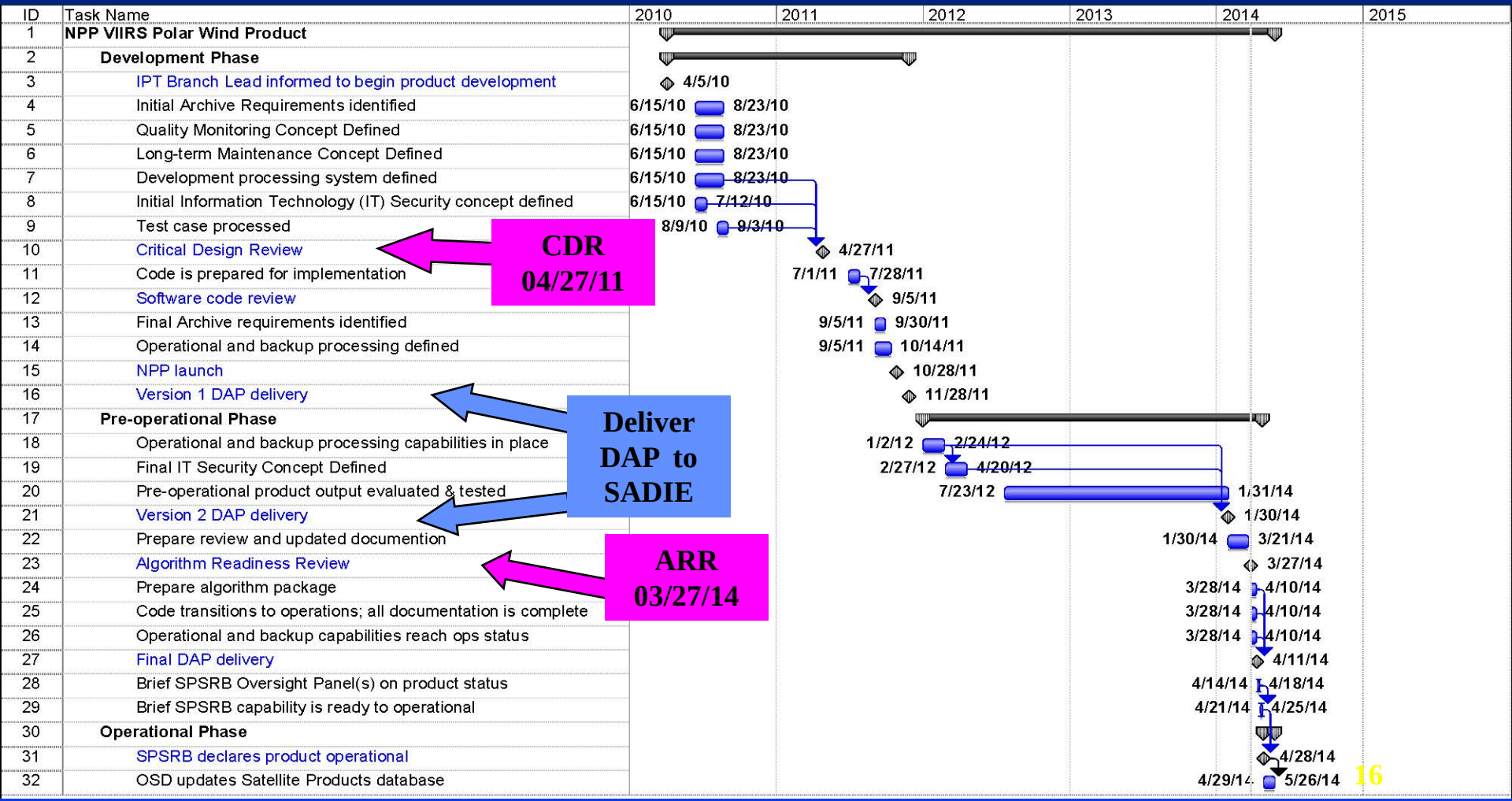
Project Stakeholders – Development Team

- **STAR (Jeff Key-P.I., Jaime Daniels, Walter Wolf, Steve Wangzong, Andrew Bailey, Shanna Sampson, Aiwu Li, Thomas King, Tianxu Yu, Larisa Koval, Yunhui Zhao)**
  - » Develop the VPW package

Project Plan: Task and Schedules

- Schedule (key milestones):
  - Preliminary Design Review – Apr 27, 2011 (Combined with CDR)
  - Critical Design Review – Apr 27, 2011
  - **System Readiness Review – Mar 27, 2014** (Oct 15, 2013)
  - SPSRB Briefing – April 2014 (Nov 2013)
  - Operations Commence – Apr 2014 (Nov 2013)

VPW Project Timeline: Algorithm Development



VPW ARR – Entry Criteria

- VIIRS Polar Winds CDR Report
- Review of VIIRS Polar Winds ARR
  - » Risks
  - » Requirements
  - » Software Architecture
  - » Unit Tests
  - » System Tests
  - » Delivered Algorithm Package
  - » New Risks and Actions

VPW Team ARR – Exit Criteria

- Algorithm Readiness Review Report
  - » The ARR Report (ARRR), a standard artifact of the STAR Enterprise Process Lifecycle (EPL), will be compiled after the ARR
  - » The report will contain:
    - – Actions
    - – Comments
    - – SRR presentation

Agenda

- Introduction
- Risks
- Requirements
- Algorithm Readiness
- Software Architecture
- Delivered Algorithm Package
- Risks and Actions
- Summary and Conclusions
- Appendix – VIIRS Polar Winds Unit Tests

CDR Review Report and Actions

Presented by

Walter Wolf

Pre-CDR Risks and Actions

VPW Pre-CDR
Open Risks and Action

- None

CDR Risks and Actions

CDR Risks and Actions For VPW

- **CDR Risk 1:** Lack of funding for product monitoring of the VPW algorithm -- not scoped in the project plan
- **Risk Assessment:** High
- **Impact:** The VPW algorithm will not be monitored by OSPO without the required funding
- **Likelihood:** High
- **Mitigation:**
  - STAR will provide OSPO their offline product validation software and documentation
- **Status:** Closed

CDR Risks and Actions For VPW

- **CDR Risk 2:** Data stitching -- the algorithm uses stitched granules at the poles to cover the polar region
- **Risk Assessment:** Low
- **Impact:** The VPW algorithm will not be able to run without the required input
- **Likelihood:** Low
- **Mitigation:**
  - The granules over the poles need to be stitched together and provided to the regridding algorithm by NDE.
  - NDE will provide all the granules over the poles and the STAR software puts them together.
- **Status:** Closed

CDR Risks and Actions For VPW

- **CDR Risk 3:** Data regridding -- The data in the stitched granules need to be regridded to a polar projection (note: the projected image i & j values are required as well as the satellite latitude and longitude)
- **Risk Assessment:** Low
- **Impact:** The VPW algorithm will not be able to run without the required input
- **Likelihood:** Low
- **Mitigation:**
  - Data is regridded to a polar projection and has been implemented
- **Status:** Closed

CDR Risks and Actions For VPW

- **CDR Risk 4:** The AIT framework currently runs on linux compiled using the INTEL compiler (32 and 64 bit)
- **Risk Assessment:** Medium
- **Impact:** Delays in porting the AIT framework to the AIX will cause delays in the delivery of the VPW to operations
- **Likelihood:** Low
- **Mitigation:**
  - » Polar Winds system will be run on Linux. The system has been compiled using both the INTEL and GNU compilers.
- **Status:** Closed

CDR Risks and Actions For VPW

- **CDR Risk 6:** Lack of funding for integration of the VPW algorithm into the AIT framework -- not scoped in the project plan
- **Risk Assessment:** High
- **Impact:** Without funding for the AIT framework development, the VPW will not be delivered to NDE
- **Likelihood:** High
- **Mitigation:**
  - The VPW algorithm has been integrated into the AIT framework and the system is being delivered
- **Status:** Closed

CDR Risks and Actions For VPW

- **CDR Risk 7:** The CRTM with the framework has not been run for the VIIRS instrument
- **Risk Assessment:** Low
- **Impact:** Without a functional CRTM product for VIIRS within the AIT framework, the VPW products can not be produced
- **Likelihood:** Low
- **Mitigation:**
  - The AIT framework interfaces to the CRTM have updated to include VIIRS data
- **Status:** Closed

CDR Risks and Actions For VPW

- **CDR Risk 8:** Schedule delays
- **Risk Assessment:** High
- **Impact:** Schedule delays will cause an increased in cost to the project
- **Likelihood:** High
- **Mitigation:**
  - The schedule has slipped. The VPW system has been delivered to go operational.
- **Status:** Closed

CDR Risks and Actions For VPW

- **CDR Risk 9:** SPSRB documentation is not within the NDE contract
- **Risk Assessment:** High
- **Impact:** Without funding for the SPSRB documents, the VPW system will be delayed in it delivery to NDE
- **Likelihood:** High
- **Mitigation:**
  - OSD has worked with STAR to update the VPW SPSRB documentation
- **Status:** Closed

Review Items Summary

- **8 risks and actions from the CDR were reported, and all of them are closed.**

Agenda

- Introduction
- Risks
- Requirements
- Algorithm Readiness
- Software Architecture
- Delivered Algorithm Package
- Risks and Actions
- Summary and Conclusions
- Appendix – VIIRS Polar Winds Unit Tests

VIIRS Polar Winds Requirements

Presented by

Walter Wolf

VIIRS Polar Winds - Requirements

Product name	Vertical Coverage	Horizontal Resolution	Vertical Reporting Interval	Mapping Uncertainty	Measurement Range	Measurement Precision	Measurement Accuracy	Latency	Refresh Rate
VIIRS Polar Winds	Surface to tropopause	~19 km	At cloud tops	0.4 km (at NADIR); 1.5km (Edge of Scan)	Speed: 3 to 100 m s^{-1}; Direction: 0 to 360 degrees	Mean vector difference: 4.2 m/s	Mean vector difference: 7.5 m/s	30 min	100 min

Project Requirements Have Been Established

- Established for Critical Design Review (CDR)
  - » Requirements Allocation Document (RAD) v1r0
- Established for Algorithm Readiness Review (ARR)
  - » Requirements Allocation Document (RAD) v2r0

Requirements Allocation Document

- **Requirements Allocation Document (RAD) v2r0**
  - » RAD v2r0 is a ARR artifact.
  - » Contains the basic and derived requirements for the work products
  - » Contains the allocation of the requirements to system components and product components

VPW Requirements – Basic Requirement 0.0

- Basic Requirement 0.0 - *The VPW project shall adopt the standard practices of the SPSRB Proccess*

  - **Derived Requirement 0.1 - *The checklist items for the VPW reviews shall be tailored. The tailored checklist items shall be established in the Review Item Disposition (RID) spreadsheet.***

  - **Derived Requirement 0.2 - *The VPW RID shall be established and maintained under CM in the VPW project artifact repository.***

VPW Requirements – Basic Requirement 1.0

- **Basic Requirement 1.0***: The VPW shall generate VPW products for U.S NWP center users.* (product, functional)

- **Derived Requirement 1.1**: *For NCEP, VPW shall generate winds over the poles created from stitched together granules from an orbit, from three consecutive orbits. (*system, functional*)*

Basic Requirement 1.0

- **JPSS-PS-R 1.1.1:** *The VPW product shall have a vertical coverage from the surface to the tropopause*
- **JPSS-PS-R 1.1.2:** *The VPW product shall have a horizontal resolution of approximately 10 km*
  - » *The requirement is incorrect and the Winds Team will work with NJO to update. It should be 19 KM*
- **JPSS-PS-R 1.1.3:** *The VPW product vertical reporting interval shall be at the cloud top*
- **JPSS-PS-R 1.1.4:** *The VPW product mapping uncertainty shall be 5 km*

Basic Requirement 1.0

- **JPSS-PS-R 1.1.5:** *The VPW measurement range shall be 3 to 100 m s-1 for speed and 0 to 360 degrees for direction*

- **JPSS-PS-R 1.1.6:** *The VPW product shall have mean vector difference measurement precision of 3.8 m/s*

- **JPSS-PS-R 1.1.7:** *The VPW product shall have a mean vector difference measurement accuracy of 7.5 m/s*

Basic Requirement 1.0

- **JPSS-PS-R 1.1.8:** *The VPW product shall have a latency of 204 minutes*

- **JPSS-PS-R 1.1.9:** *The VPW product shall have a refresh rate of N/A*

VPW Requirements – Basic Requirement 1.0

- **Derived Requirement 1.1.10**: *The VPW shall develop the BUFR table for the polar winds.* (system, functional)
  - **Derived Requirement 1.1.10.1**: *The VPW polar winds data for NCEP shall be represented as scaled integers* (system, functional)
  - **Derived Requirement 1.1.10.2**: *The polar winds shall be represented by 16 bit words in the BUFR format.* (system, functional)
  - **Derived Requirement 1.1.10.3**: *The VPW shall supply the NDE System Development Team (SDT) with BUFR table as well as the frequency list.* (product, operational)
  - **Derived Requirement 1.1.10.4**: *The BUFR table shall use delayed replication for writing the winds for different channels.* (product)
  - **Derived Requirement 1.1.10.5**: *The BUFR must contain the VIIRS derived cloud fraction and cloud height.* (performance)

VPW Requirements – Basic Requirement 1.0

- **Derived Requirement 1.1.10.6:** *The BUFR table shall contain the following variables. Some variables are used multiple times due to the different channels.* (product)

Satellite ID
ID of originating center
Satellite instrument
Satellite classification
Segment size at nadir in X direction
Segment size at nadir in Y direction
Year
Month
Day
Hour
Minute
Second
Satellite instrument data used I
Satellite derived wind computation
Latitude
Pressure
Winds direction
Orbit Number
Satellite channel centre frequency
Satellite channel band width
Coldest cluster temperature
Height assignment method
Tracer correlation method
Land/Sea qualifier
Origin of first quess information
Time Significance
Time period of displacement
Temperature/Dry bulb temperature
Backward reference bit map
Data present indicator
Percent confidence
Longitude
Satellite Zenith
Wind speed

VPW Requirements – Basic Requirement 1.0

- **Derived Requirement 1.1.11**: *The product for NCEP shall be available within three hours of observation.*

- **Derived Requirement 1.1.12**: *The VPW shall write polar winds for NCEP into netCDF4 format so tailoring can convert it into BUFR format. Therefore, the contents of the BUFR table defined in section 1.1.2 are, at least, a subset of the netCDF4 output files.*

VPW Requirements – Basic Requirement 1.0

- **Derived Requirement 1.2**: *The VPW processing system shall be a stand alone system.*
  - » **Derived Requirement 1.2.1**: *The VPW processing system shall be written in C, C++, and/or Fortran90/95/2000 programming languages*

VPW Requirements – Basic Requirement 1.0

- **Derived Requirement 1.3**: *The VPW Integrated Product Team (IPT) shall perform validation and verification of the VPW products.*
  - » **Derived Requirement 1.3.1**: *The VPW IPT shall verify that the winds products in the output netCDF4 files are generated correctly and documented.*

VPW Requirements – Basic Requirement 2.0

- **Basic Requirement 2.0**: *The VPW shall use cloud products as inputs for the winds algorithm*
  - » **Derived Requirement 2.1**: *The VPW shall use the GOES-R cloud height product as an input to the algorithm*
    - ▪ **Derived Requirement 2.1.1**: *The cloud height algorithm shall use the NPP cloud mask product as an input*
      - ▪ **Derived Requirement 2.1.1.1**: *NDE shall provide the NPP cloud mask product to the VPW system*

VPW Requirements – Basic Requirement 2.0

- **Derived Requirement 2.1.2**: *The cloud height algorithm shall use the GOES-R cloud phase product as an input.*
  - **Derived Requirement 2.1.2.1:** ***The GOES-R cloud phase algorithm will be delivered within the VPW system***
- **Derived Requirement 2.1.3:** ***The GOES-R cloud height algorithm shall be delivered within the VPW system***

VPW Requirements – Basic Requirement 2.0

- **Derived Requirement 2.1.4**: *The cloud height algorithm shall require IMS snow mask information as inputs*
- **Derived Requirement 2.1.5**: *The cloud height algorithm shall require NCEP model forecast information as input*
- **Derived Requirement 2.1.6**: *The cloud height algorithm shall require CRTM radiances and transmittances as inputs*

» **Derived Requirement 2.2**: *The VPW shall use the NCEP forecast products as inputs to the algorithm*

VPW Requirements – Basic Requirement 3.0

- **Basic Requirement 3.0**: *The VPW output files shall be archived* (product, functional)
  - **Derived Requirement 3.1**: *The VPW shall generate for CLASS an associated metadata file for each product file.* (product, quality)
  - **Derived Requirement 3.2**: *NDE shall create a Data Submission Agreement (DSA) and an ICD with CLASS. The DSA shall include all information regarding the archival of product files.* (product, quality)

VPW Requirements – Basic Requirement 4.0

- **Basic Requirement 4.0**: *The VPW shall generate daily matchups for system validation, maintenance, and development.*
  - » **Derived Requirement 4.1**: ***The VPW shall generate matchup datasets between satellite measurements and other existing correlated measurements. These products are for STAR.***
    - **Derived Requirement 4.1.1**: ***The VPW shall generate daily radiosonde matchups with the polar winds***

VPW Requirements – Basic Requirement 5.0

- **Basic Requirement 5.0**: *The VPW system shall be delivered to NDE for integration into the NDE Data Handling System (DHS). The following is required as part of this delivery.*
  - » **Derived Requirement 5.1**: *The VPW system shall be delivered according to the specifications identified in the Standards for NDE Algorithm Development (2007) for a Delivered Algorithm Package (DAP)*

VPW Requirements – Basic Requirement 5.0

- **Derived Requirement 5.1.1**: *DAP item 3 shall be the delivery of all science algorithm source code including make files and scripts*
- **Derived Requirement 5.1.2**: *DAP item 4 shall be the delivery of test input data, temporary files, and expected output data*
- **Derived Requirement 5.1.3**: *DAP item 5 shall be the delivery of coefficient files and look-up tables (in SMM with DAP)*
- **Derived Requirement 5.1.4:** *DAP item 6 shall be the delivery of processing steps as data flow diagrams (in SMM with DAP)*
- **Derived Requirement 5.1.5**: *DAP item 7 shall be the description of all input data files (including PCF) and any intermediate files (format, content) (in SMM with DAP)*
- **Derived Requirement 5.1.6**: *DAP Item 8 shall be the description of all output data files (format, content) and requirements (in SMM and RAD with DAP)*

VPW Requirements – Basic Requirement 5.0

- **Derived Requirement 5.1.7**: *DAP item 9 shall be the description of product quality monitoring information (quality flags, quality flag values in SMM with DAP)*
- **Derived Requirement 5.1.8**: *DAP item 10 shall be the delivery of test plans, test description, test procedures, and detailed performance testing results*
- **Derived Requirement 5.1.9**: *DAP item 11 shall be the delivery of estimates of resources required for execution (run time, disk space, memory)*
- **Derived Requirement 5.1.10**: *DAP item 12 shall be the delivery of an Algorithm Theoretical Basis Document (ATBD)*

VPW Requirements – Basic Requirement 5.0

- **Derived Requirement 5.1.14:** ***DAP item 1 shall be the delivery of a delivery memo.*** **(system, delivery)**

  - **Derived Requirement 5.1.14.1:** ***The delivery memo shall include the following:***

    (1) *Point(s) of contact for questions specific to the algorithm (include name, telephone, and e-mail address). At a minimum, include algorithm developer and PAL.*
    (2) *Date of Current DAP Release*
    (3) *Purpose of the delivery (e.g. an initial release, modification)*
    (4) *Brief description of package (e.g. "DAP Version X represents the initial delivery of the…", see Appendix B for a more detailed example)*
    (5) *List of DAP contents*
    (6) *Description of problem(s) resolved, if any, and method of resolution.*
    (7) *Description of significant changes from previous version, if any.*
    (8) *List of known remaining updates/issues.*

VPW Requirements – Basic Requirement 5.0

- **Derived Requirement 5.1.15:** ***DAP item 2 shall be the delivery of a README text file*** **(system, delivery)**
  - **Derived Requirement 5.1.15.1:** ***The README text file shall include the following:***

(1) *DAP version number.*
(2) *Date of current DAP release*
(3) *Target configuration for SA source code execution.*
(4) *List of all pointers (or hyperlinks) to reference documents*
(5) *Production Rule Definitions*
(6) *List of expected compiler warnings*
(7) *Other pertinent information as judged by the algorithm developer(s) (compiler settings, etc.).*

VPW Requirements – Basic Requirement 5.0

- **Derived Requirement 5.2**: *The VPW shall be able to run on the NDE target platform (Linux)*
- **Derived Requirement 5.3**: *The VPW unit driver scripts shall be able to read and handle the content from a Process Control File (PCF) passed to them from the NDE Product Generation Manager (PGM) for each run of the script)*
- **Derived Requirement 5.4**: *The VPW unit driver scripts shall each produce a Process Status File (PSF) after each run of the script in the format required by the NDE Data Handling System (DHS)*
- **Derived Requirement 5.5**: *The VPW code shall adhere to the STAR/NDE coding and security standards)*
- **Derived Requirement 5.6**: *The VPW software shall be able to write all output product into netCDF4 format*

VPW Requirements – Basic Requirement .0

- **Basic Requirement 6.0**: *The delivered VPW system shall be able to handle input and output data.*
  - » **Derived Requirement 6.1**: *All VPW software units shall be able to perform data range checks on the input NetCDF4 files provided by the NDE DHS*
  - » **Derived Requirement 6.2**: *The VPW software shall be able to read the VIIRS radiance, VIIRS geolocation, and VIIRS Cloud Mask netCDF4 input data supplied by NDE*

Basic Requirement 7.0

- **JPSS-RRPS-R 7.0:** *STAR shall deliver a VPW document package to OSPO.*
  - » **Driver:** This basic requirement is traced to an OSPO need for documentation to support operations, maintenance, and distribution.

Basic Requirement 7.0

- **JPSS-RRPS-R 7.1:** *The VPW document package shall include a README text file.*

- **JPSS-RRPS-R 7.1.1:** *The README file shall list each item in the final pre-operational system baseline, including code, test data, and documentation.*

  - » All required deliverable items must be correctly identified

Basic Requirement 11.0

- **JPSS-RRPS-R 7.2:** *The VPW document package shall include a Review Item Disposition (RID) document.*

- **JPSS-RRPS-R 7.2.1:** *The RID shall describe the final status of all development project tasks, work products, and risks.*

Basic Requirement 7.0

- **JPSS-RRPS-R 7.3:** *The VPW document package shall include an Algorithm Theoretical Basis Document (ATBD).*
  - » The ATBD will follow SPSRB Version 2 document standards
- **JPSS-RRPS-R 7.4:** *The VPW document package include a Requirements Allocation Document (RAD).*
  - » The RAD will follow document standards stated in EPL v3 process asset DG-6.2

Basic Requirement 7.0

- **JPSS-RRPS-R 7.5:** *The VPW document package shall include a System Maintenance Manual (SMM).*
  - » The SMM will follow SPSRB Version 2 document standards.
- **JPSS-RRPS-R 7.6:** *The VPW document package shall include an External Users Manual (EUM).*
  - » The EUM will follow SPSRB Version 2 document standards.
- **JPSS-RRPS-R 7.7:** *The VPW document package shall include a Critical Design Document (CDD).*
  - » The CDD will follow STAR EPL document standards in DG-8.2 and DG-8.2.A.

Basic Requirement 7.0

- **JPSS-RRPS-R 7.8:** *The VPW document package shall include a Code Test Document (CTD).*
  - » The CTD will follow STAR EPL document standards in DG-10.3 and DG-10.3.A.
- **JPSS-RRPS-R 7.9:** *The VPW document package shall include a System Readiness Document (SRD).*
  - » The SRD will follow STAR EPL document standards in DG-11.5 and DG-11.5.A.

Basic Requirement 7.0

- **JPSS-RRPS-R 7.10:** *The VPW document package shall include a System Readiness Review Report (SRRR).*
  - » The SRRR will follow document standards stated in EPL v3 process asset DG-11.6
- **JPSS-RRPS-R 7.10.1:** *The SRRR shall document the approved readiness of the VPW system for transition to operations.*
  - » This is an SRR exit criteria item

Basic Requirement 8.0

- **JPSS-RRPS-R 8.0:** *The VPW system shall undergo an OSPO Code Review Security for security compliance*
  - » **Driver:** OSPO Security

Basic Requirement 8.0

- **JPSS-RRPS-R 8.1:** *The VPW system shall comply with OSPO data integrity check list.*
  - » OSPO data integrity check list is part of the OSPO Code Review Security check lists.
- **JPSS-RRPS-R 8.2:** *The VPW system shall comply with OSPO development security check list.*
  - » OSPO development security check list is part of the OSPO Code Review Security check lists.

Basic Requirement 8.0

- **JPSS-RRPS-R 8.3:** *The VPW system shall comply with OSPO code check list.*
  - » OSPO code check list is part of the OSPO Code Review Security check lists.

Basic Requirement 9.0

- **JPSS-RRPS-R 9.0:** *The VPW developers shall specify IT resource needs for operations.*
  - » **Driver:** OSPO IT Capacity Planning

Basic Requirement 9.0

- **JPSS-RRPS-R 9.1:** *The JPSS RRPS system shall run on Redhat Linux.*
  - » Servers are available.
- **JPSS-RRPS-R 9.2:** *Operational server shall have 3 TB of disk space.*
  - » Available servers have this capability.
- **JPSS-RRPS-R 9.3:** *Each operational server shall have 2 GB of RAM for each core.*
  - » Available servers have this capability.

VPW Requirements – Summary

- The VIIRS Polar Winds Requirements have been established.
- The VPW Requirements have been documented in the Requirements Allocation Document (RAD).
- The VPW Requirements are traceable to drivers (customer needs or expectations) and other requirements.

Agenda

- Introduction
- Risks
- Requirements
- Algorithm Readiness
- Software Architecture
- Delivered Algorithm Package
- Risks and Actions
- Summary and Conclusions
- Appendix – VIIRS Polar Winds Unit Tests

VIIRS Polar Winds Algorithm Readiness

Presented by

Jeff Key

NOAA/NESDIS/STAR

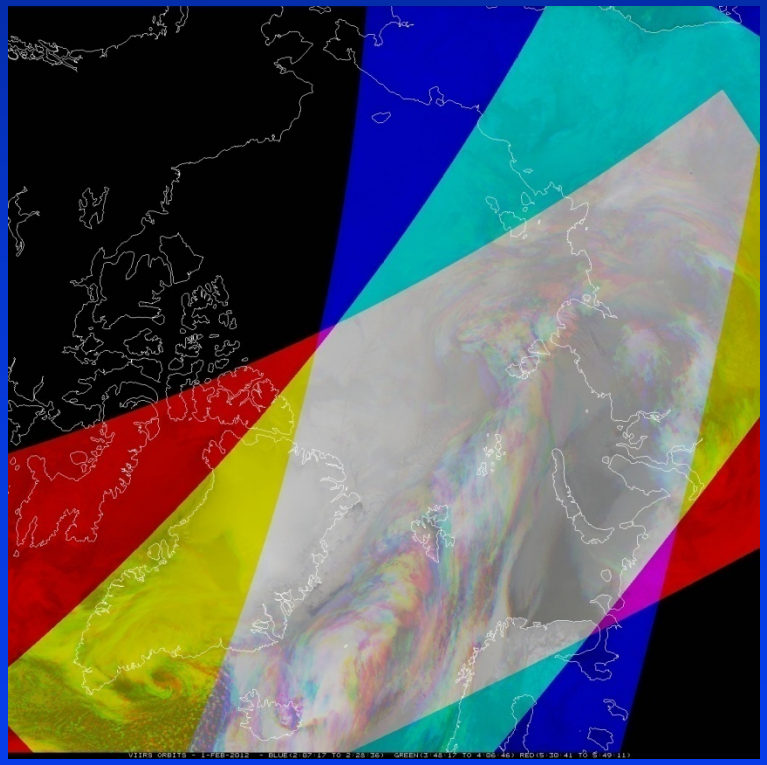
Outline

- Product Overview and Requirements
- Product Examples
- Validation Strategy
- Validation Results

Goal: Demonstrate science quality of the VIIRS wind product.

VIIRS Winds Product Overview & Requirements

- VIIRS Polar Winds will be derived by tracking clouds features in the VIIRS longwave infrared channel (Band M15, 10.8um)
  - » Wind speed, direction, and height are measured throughout the troposphere, poleward of approximately 65 degrees latitude, in cloudy areas only
  - » VIIRS Polar Winds will be generated in both Arctic and Antarctic Pole regions
  - » The product will include observation time, data quality flags, and metadata



VIIRS Winds Product Overview & Requirements

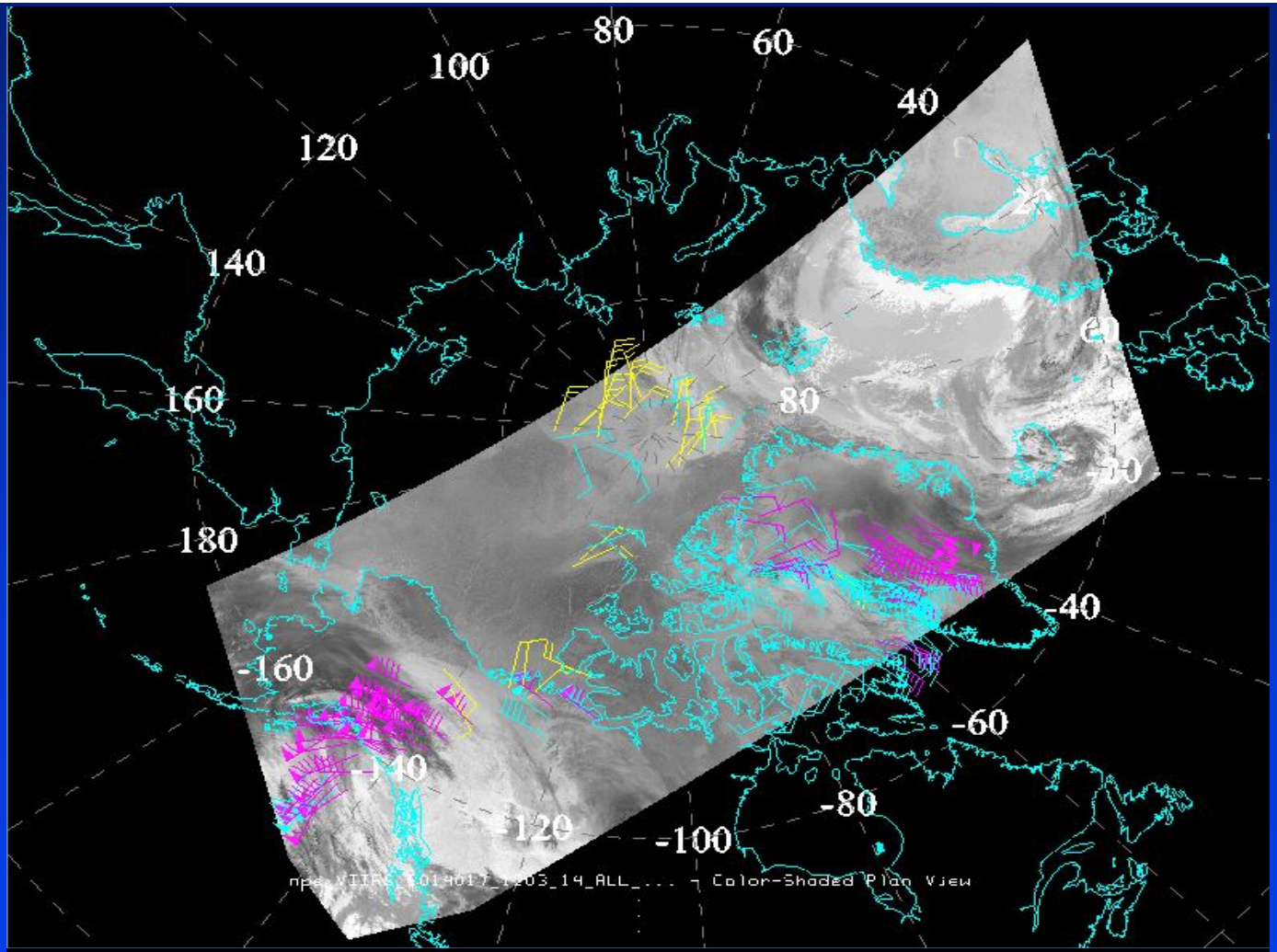
- VIIRS Polar Winds will be derived by tracking clouds features in the VIIRS longwave infrared channel (Band M15, 10.8um)
  - » Wind speed, direction, and height are measured throughout the troposphere, poleward of approximately 65 degrees latitude, in cloudy areas only
  - » VIIRS Polar Winds will be generated in both Arctic and Antarctic Pole regions
  - » The product will include observation time, data quality flags, and metadata

System Capability	Objective
Vertical Coverage	Surface to Tropopause
Horizontal Resolution	~10 km (Should be 19 km, will work with NJO)
Vertical Reporting Interval	At cloud tops
Mapping Uncertainty	5 km
Measurement Range	Speed: 3 to 100 m/s Direction: 0 to 360 degrees
Measurement Precision	3.8 m/s
Measurement Accuracy	7.5 m/s (Mean Vector Difference)
Latency	204 minutes
Refresh	N/A

Algorithm Output

Arctic

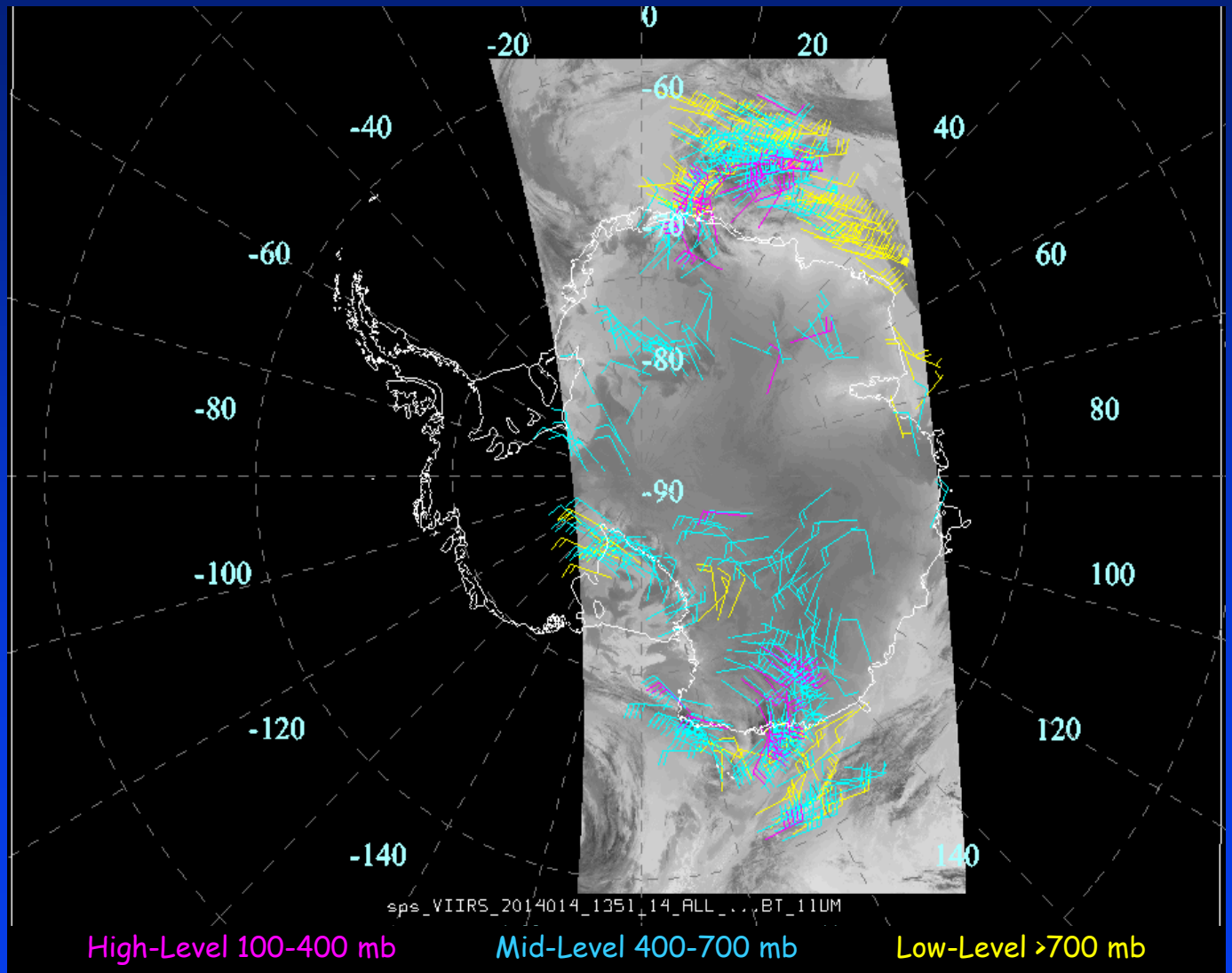
NPP/VIIRS LWIR Winds derived from overlap region of three consecutive M15 band (10.8μm) images on 01 17 2014 12:03UTC



Algorithm Output

Antarctica

NPP/VIIRS LWIR Winds derived from overlap region of three consecutive M15 band (10.8μm) images on 01 14 2014 13:51UTC

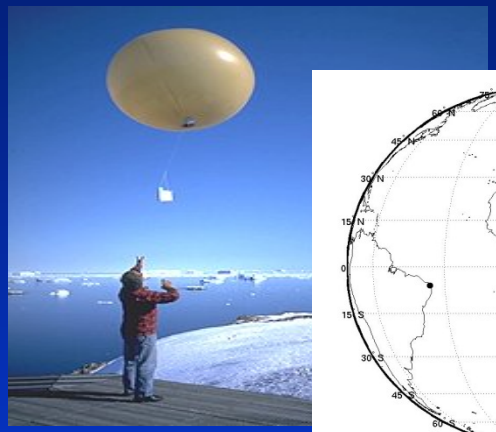


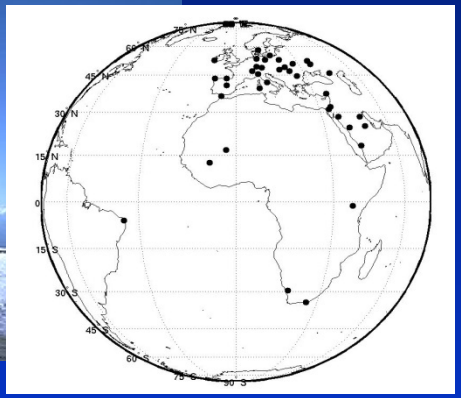
Validation Strategy

- Derive winds over both poles using overlapping NPP/VIIRS orbits
- Derive winds with full product precedence in place
  - » Official NPP cloud mask product is used
  - » Cloud Products (cloud-top temp, pressure, phase, type) are generated as part of the product precedence chain
- Collocate (in space and time) derived satellite winds with reference ("truth") winds
  - » Radiosonde wind observations (Land)
  - » Aircraft wind observations (Land & Ocean)
  - » GFS analysis winds (Ocean)
- Generate comparative statistics (Satellite winds – Reference Winds)
  - » Accuracy
  - » Precision

ABI Derived Motion Winds Test Plan – Offline Validation: Truth Data

- Radiosonde wind observations serve as a key validation data source for derived motion wind products
  - » Used by all operational satellite processing centers that generate satellite derived motion winds

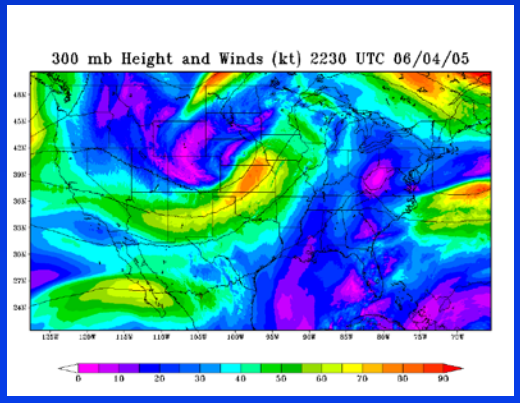




Radiosonde stations over land within MSG/SEVIRI domain

- Aircraft wind observations

- GFS Model Analysis Wind Fields



WRF Model Forecast Wind Fields

Comparisons to Radiosondes

September, 2013 – January, 2014

Comparison statistics of VPW product computed using the M15 band (10.76um),

All Levels (100-1000 hPa)	*VIIRS Polar Wind vs. Radiosonde Winds (m/s)*		*GFS Forecast Winds vs. Radiosonde Winds (m/s)*	
	NHEM	SHEM	NHEM	SHEM
Accuracy	5.67	5.71	4.54	4.77
Precision	3.41	3.25	3.06	2.99
Speed bias	0.38	-0.04	-0.30	-0.57
Speed	17.61	14.22	16.93	13.69
Sample	9650	866	9650	866
High Level (100-400 hPa)	**NHEM**	**SHEM**	**NHEM**	**SHEM**
Accuracy	6.21	6.81	5.08	5.56
Precision	3.55	3.36	3.23	3.14
Speed bias	-0.06	-0.23	-0.69	-0.55
Speed	23.62	18.05	22.99	17.73
Sample	3054	301	3054	301
Mid Level (400-700 hPa)	**NHEM**	**SHEM**	**NHEM**	**SHEM**
Accuracy	5.65	5.24	4.48	4.48
Precision	3.40	3.12	3.04	2.87
Speed bias	0.56	0.07	-0.32	-0.75
Speed	16.69	12.51	15.81	11.69
Sample	4468	471	4468	471
Low Level (700-1000 hPa)	**NHEM**	**SHEM**	**NHEM**	**SHEM**
Accuracy	4.95	4.55	3.90	3.70
Precision	3.08	2.39	2.69	2.40
Speed bias	0.64	0.04	0.32	0.28
Speed	10.91	10.52	10.58	10.76
Sample	2128	94	2128	94

Specifications:

Measurement Accuracy	7.5 m/s
Measurement Precision	3.8 m/s

Comparisons to Aircraft
January 29, 2014 – February 23, 2014

Additional comparison statistics of the VPW product computed using the M15 band (10.76um),

All Levels (100-1000 hPa)	*VIIRS Polar Wind vs. Aircraft Winds (m/s)*
	NHEM
Accuracy	6.10
Precision	3.27
Speed bias	0.36
Speed	18.76
Sample	3225
High Level (100-400 hPa)	**NHEM**
Accuracy	6.56
Precision	3.36
Speed bias	-0.34
Speed	20.24
Sample	2082
Mid Level (400-700 hPa)	**NHEM**
Accuracy	5.37
Precision	3.22
Speed bias	1.57
Speed	17.37
Sample	945
Low Level (700-1000 hPa)	**NHEM**
Accuracy	4.77
Precision	2.00
Speed bias	1.93
Speed	9.81
Sample	198

Specifications:

Measurement Accuracy	7.5 m/s
Measurement Precision	3.8 m/s

Summary of Validation Results for VIIRS Winds Product

- NPP/VIIRS wind product has been validated against radiosonde wind observations, aircraft wind observations, and GFS analysis winds

- The accuracy and precision of the VIIRS winds fall well within the accuracy and precision specifications

Agenda

- Introduction
- Risks
- Requirements
- Algorithm Readiness
- Software Architecture
- Delivered Algorithm Package
- Risks and Actions
- Summary and Conclusions
- Appendix – VIIRS Polar Winds Unit Tests

VIIRS Polar Winds Product System (VPWPS) Software Architecture

Meizhu Fan, Shanna Sampson, Aiwu Li,
Peter Keehn, Tianxu Yu, Jaime Daniels,
Wayne Bresky, Walter Wolf

Presented by

Aiwu Li

VPWPS
Software Architecture

Outline

- VPWPS System Overview
- VPWPS Detailed Design
- AIT Framework
- Transition to Operations

VPWPS
Software Architecture

- Hardware Environment
  - » STAR Science Development
  - » System Development and Unit Testing
  - » Test and Production
- Software Description
  - » External-Level Flow
  - » System Level Flow
  - » Unit Level Flow
  - » Sub-Unit Flow

VPWPS
Software Architecture

- Data Files
  - » Input Files
  - » Static/Ancillary Files
  - » Output Files
  - » Log Files
  - » Resource Files
  - » File Formats

VPWPS
Development Hardware

- VPW science development occurs in the STAR Collaborative Environment.

- Science development hardware:
  - Dell Intel
  - OS Version: Red Hat Enterprise Linux 5.9
  - Intel Fortran compiler: Version 13
  - Intel C/C++ Compiler: Version 13

- STAR machines
  - rhw1044 – Algorithm Integration
  - rhw1045 – Testing
  - orbit171l – Clouds product retrieval
  - orbit183l – Winds retrieval development
  - orbit211l – SDR decoder
  - orbit242l – SDR reformatting
  - orbit241l – Validation

VPWPS
Development Hardware

- VPWPS system development occurs on rhw1044 located in the STAR Collaborative Environment.

- The current rhw1044 configuration is:
  - Dell Intel
  - OS Version: Red Hat Enterprise Linux 6.4
  - Disk space: 2 TB
  - Number of Processors: 16
  - Total Memory: 32 GB
  - Processor Clockspeed: 3.20 GHz
  - Intel Fortran Compiler: Version 13.0.1
  - Intel C/C++ Compiler: Version 13.0.1

- Upgrades: No plans for upgrades at this time.

VPWPS
Test Hardware at STAR

- VPWPS system testing occurs on rhw1045 located in the STAR Collaborative Environment.

- The current rhw1045 configuration is:
  - Dell Intel
  - OS Version: Red Hat Enterprise Linux 6.4
  - Diskspace: 2 TB
  - Number of Processors: 16
  - Total Memory: 32 GB
  - Processor Clockspeed: 3.20 GHz
  - (No compiler)

- Upgrades: No plans for upgrades at this time.

VPWPS
NDE Hardware

- The VPW official test machine is located within the NDE environment
- The VPW operational machine is located within the NDE environment

VPWPS
Software Package Summary

- The VPWPS is an integrated system built on top of the GOES-R AIT framework.
- The system is written in C/C++, FORTRAN, PERL
- Three third-party software applications are used.

Description	File and Language	Number of Files
System processing control scripts	Perl scripts	13
AIT Framework (including SDR reader, CRTM, Clouds and Winds algorithm, etc.)	C/C++, FORTRAN 90, Makefiles, XML, etc.	1264
SDR Gap-filling*	Python scripts (python 2.7.3)	2
JPSS HDF5 to netCDF4 Convertor*	C/, Makefiles	8
Swatch to Grid Remapping*	C/C++, FORTRAN 90, Makefiles	219
Granule data handling programs(for extracting selected variables, merge granules, NetCDF formatter, etc)	C/C++, FORTRAN 90, Perl scripts.	6

* third-party software

VPWPS
Software Package Summary (cont.)

- Third-party software applications*:
  - SDR Gap-filling utility tool
    - Python package for mending bowtie removed pixels in a VIIRS SDR image.
    - *ftp://ftp.ssec.wisc.edu/pub/shellb3/ShellB3-Linux-x86_64-20120621-r713-core.tar.gz*
  - JPSS HDF5 to NetCDF4 conversion utility ***(h5augjpss**, v1.1.0)*
    - Command line tool that modifies JPSS HDF5 files by adding associated data or metadata or by hiding HDF5 elements in order to make the file accessible to NetCDF4 based applications and tools.
    - *http://www.hdfgroup.org/projects/jpss/h5augjpss_index.html*
  - Swath-to-Grid Toolbox (***MS2GT**, v0.24)*
    - A set of software tools that was originally developed for extracting data from HDF-EOS files containing MODIS swath data into flat binary files containing gridded data in a variety of map projections.
    - *http://cires.colorado.edu/~tharan/ms2gt/ms2gt0.24.tar.gz*

* Source code of the 3rd-party tools will be included in the VPWPS DAP

VPWPS
Software Package Summary (cont.)

- **Libraries and Utilities**:
  - » **NetCDF4 Library version 4.1.3 (available from Unidata website)**
    - – Compiled as 64 bit
    - – Libraries supplied by NDE
  - » **HDF5 Library version 1.8.10 (available from HDF Group website)**
    - – Compiled as 64 bit
    - – Requires szip 2.1 and zlib zlib-1.2.5
    - – Libraries supplied by NDE
  - » **CRTM Library version 2.0.2 (available from Joint Center for Satellite Data Assimilation (JCSDA)**
    - – Fortran 90 code
    - – Compiled as 64 bit
    - – Source code supplied within the VPWPS DAP
  - » **wgrib2 version 0.1.8.2 (available from NCEP CPC website)**
    - – C and Fortran 90 code
    - – Compiled as 64 bit
    - – Utility tool supplied by NDE
  - » **Libxml2 devel version 2.7.6 (available from website xmlsoft.org or gnome.org)**
    - – C code
    - – Compiled as 64 bit
    - – Utility tool supplied by NDE

VPWPS
Data Storage Information

- The storage usage for all VPWPS primary and ancillary input data, intermediate data, and output data for one day.

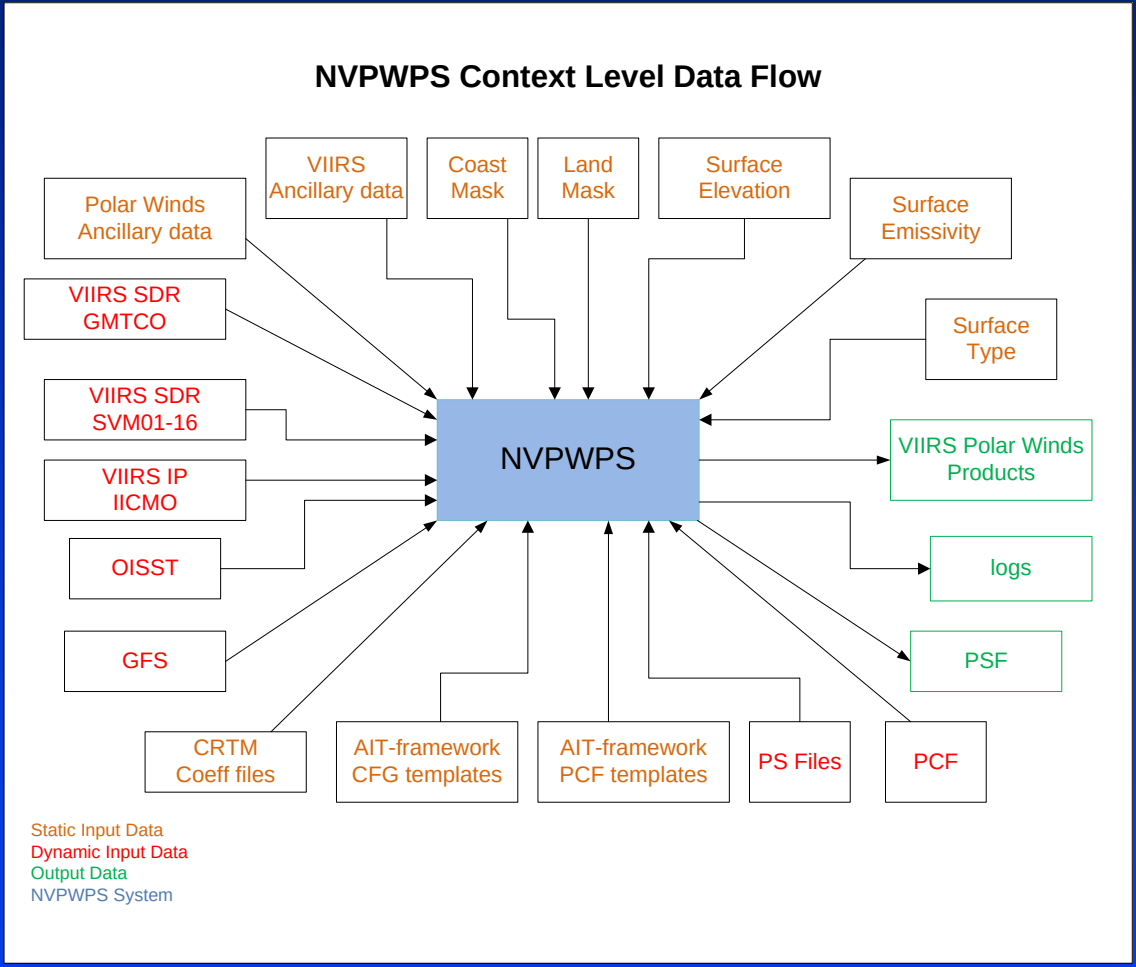
Storage Items	Data Size* *(Per/day)*
Input VIIRS granule data files (SVM01-SVM16, GMTCO, IICMO)	120 GB
Product Data Files	140 MB
Intermediate data Files	640 GB
Static Ancillary	13 GB
GFS Forecasts	800 MB
OISST Daily	8 MB
System Code (executable)	150 MB
System scripts and Resource Files	936 KB
Third-party utility tools (h5augjpss, ms2gt, ShellB3, CRTM, etc)	568 MB
Total	**~780 GB**

* The information in the table is based on 14 orbits per day with 15 granules on each polar image.

VPWPS
External-Level Interface

- VPWPS external interfaces are shown on the following 4 slides.
  - » 1 slide showing the context level input/output data flow and specs.
  - » 2 slides showing the external-level interfaces in NDE DHS environment
  - » 1 slide showing production and user interface flow chart

VPWPS
External-Level Data Flow

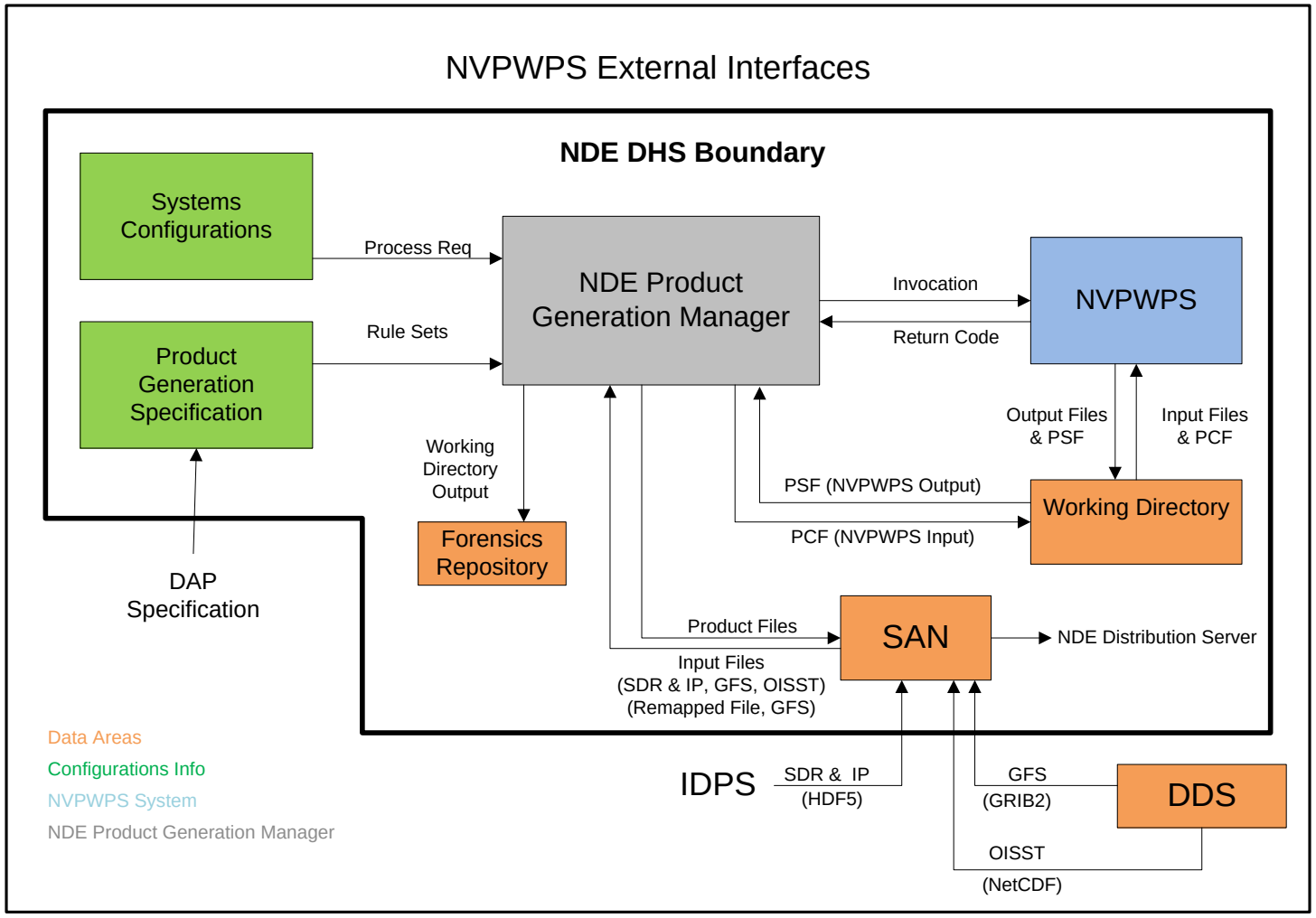


Note: The VPWPS is also referred to as the NOAA VIIRS Polar Winds Processing System (NVPWPS) in VPW documents(i.e. SMM, EM.)

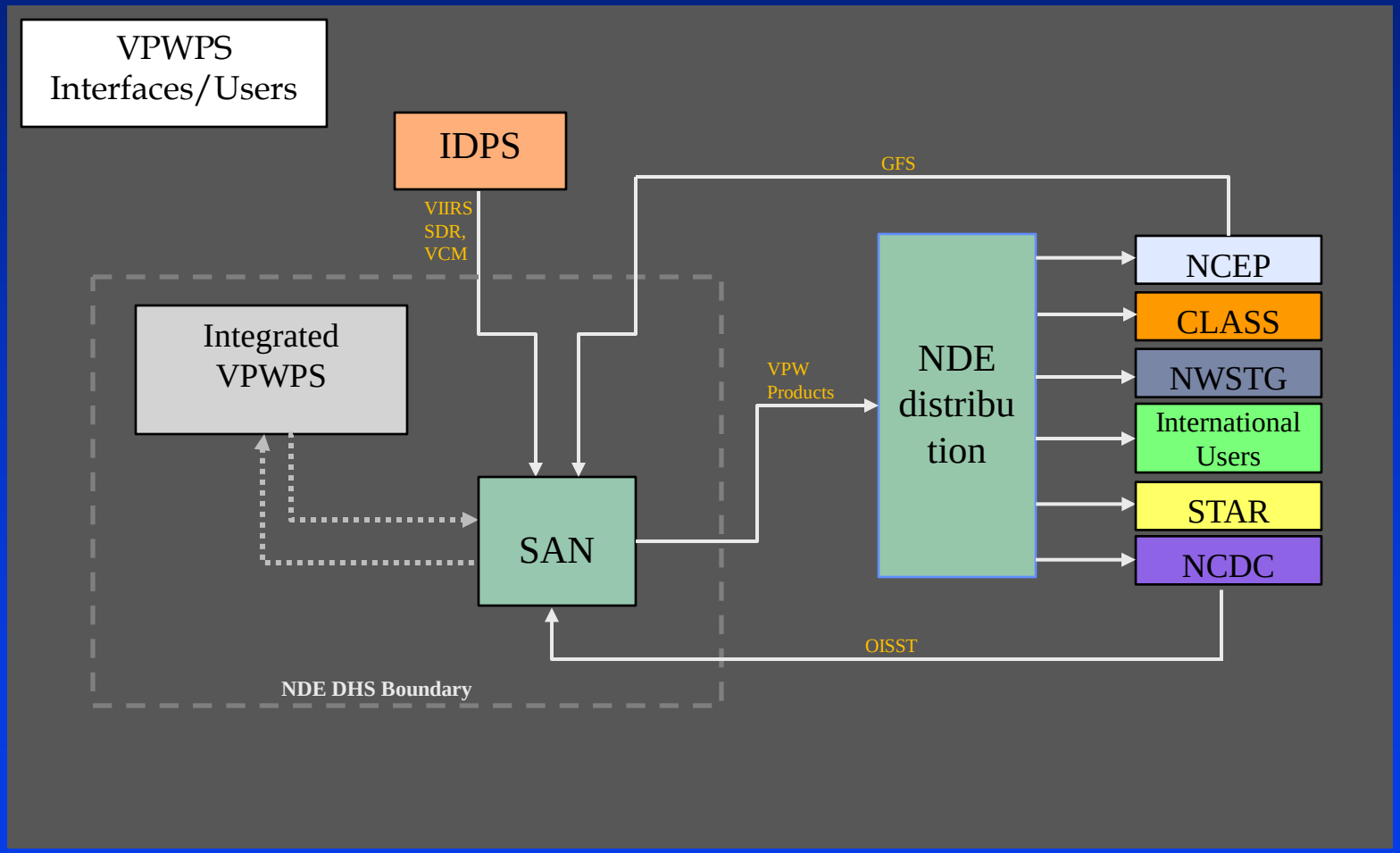
VPWPS
External Interfaces

- The VPWPS will be run via the execution of top level driver scripts that will be invoked, monitored, and managed by the NDE Data Handling System(DHS).
- Execution of the scripts will be event-driven (i.e. the arrival of a list of input data files which qualify the time and geo-coverage criteria provided by the VPW product science team).
- Each VPWPS driver script is designed to run in a single working directory. All unit output will be produced in this same working directory.
- Each processing run will get a Processing Control File (PCF) from the NDE Product Generation Manager(PGM) containing all the input file locations and parameters and other information needed by VPW product processing.
- Each processing run will produce a PSF file containing information about product files.

VPWPS
External Interfaces



VPWPS
Interfaces/Users



Outline

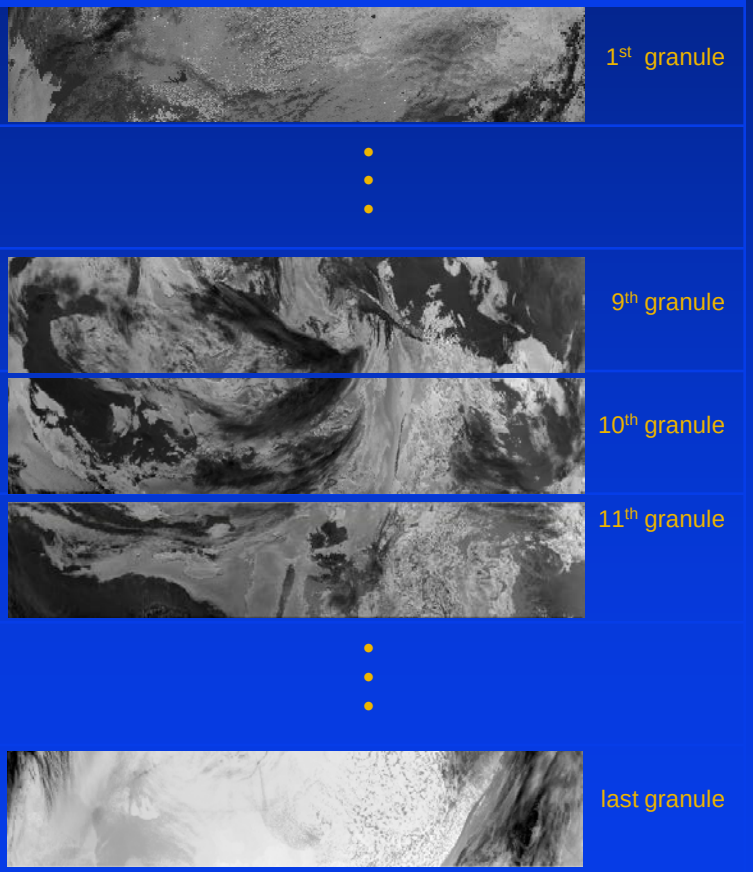
- VPWPS System Overview
- VPWPS Detailed Design
- AIT Framework
- Transition to Operations

VPWPS
General Background

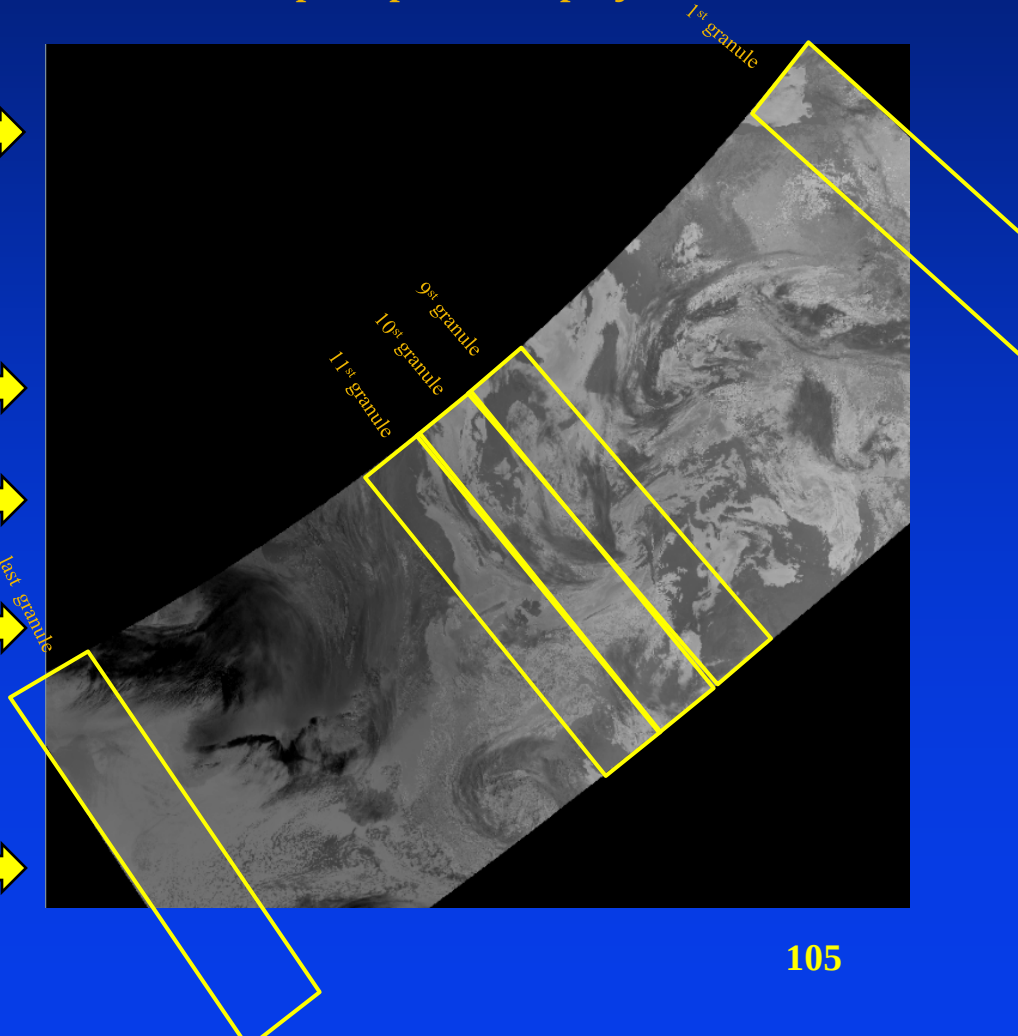
- The VPW products are generated from NPP VIIRS SDR granules over the north polar region (> 50 N) and the south polar region (> 50S).
- Three overlapped polar passes/images are required for Winds product retrieval.
- Each image contains 14~18 aggregated 86-seconds VIIRS SDR granules over the polar region.
- All input data and output data need to be in NetCDF format for AIT framework cloud and winds product processing.
- All input data (SDR, clouds products, ancillary data, etc) required by VPW algorithm needs to be remapped to standard polar stereographic(PS) projection for product processing.

VPWPS
General Background

List of VIIRS granules over polar region

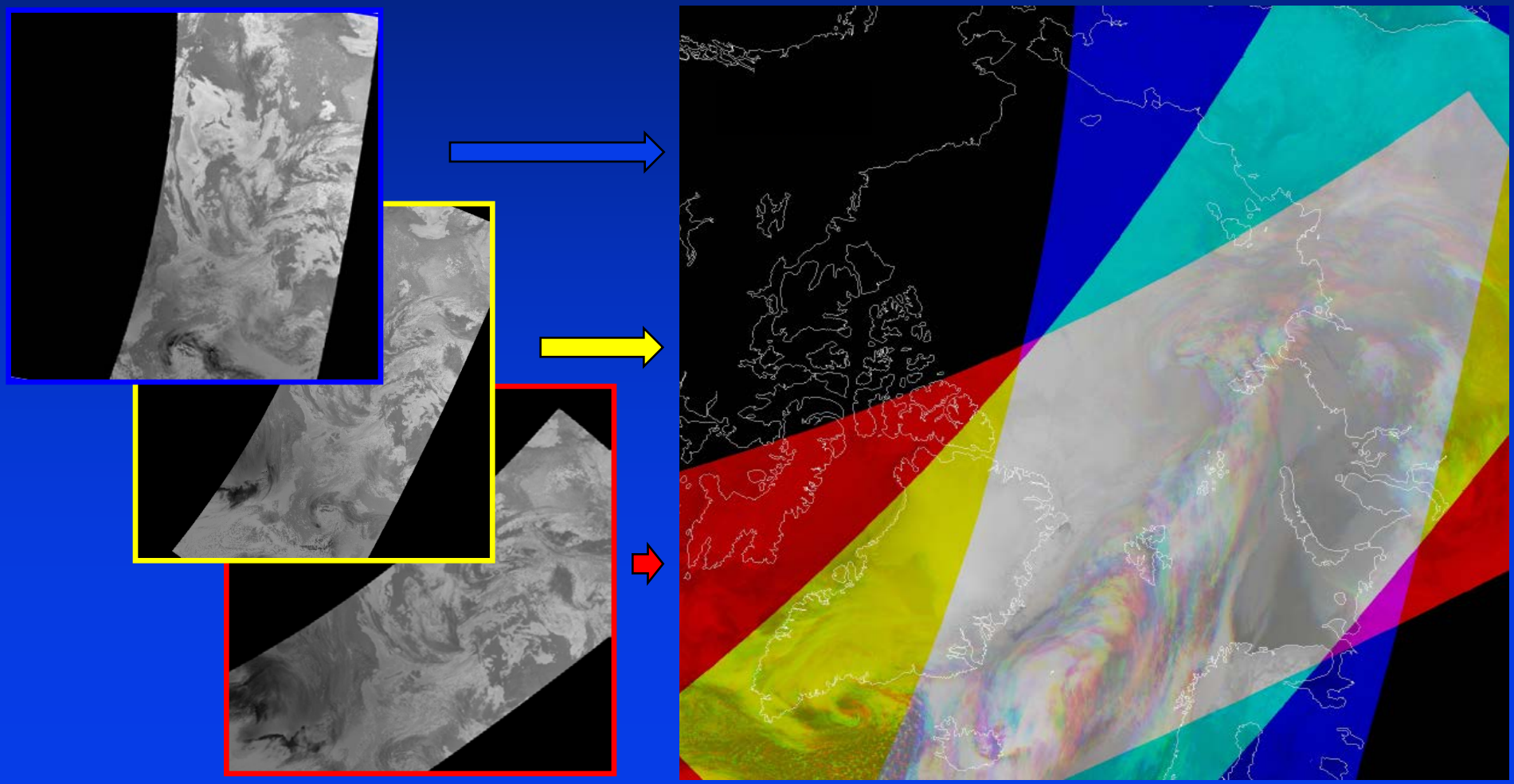


VIIRS polar pass in PS projection



VPWPS
General Background

Three overlapped polar passes are required for the VPW retrieval



VPWPS
System Level Interfaces

- The VPWPS consists of two stand alone production units (or subsystems)
  - **Preprocessor** - generates remapped data from granules on a polar pass
  - **Product Generator** - generates winds products from a set of remapped triplet polar passes

- The system layer is where NDE invokes the VPWPS.

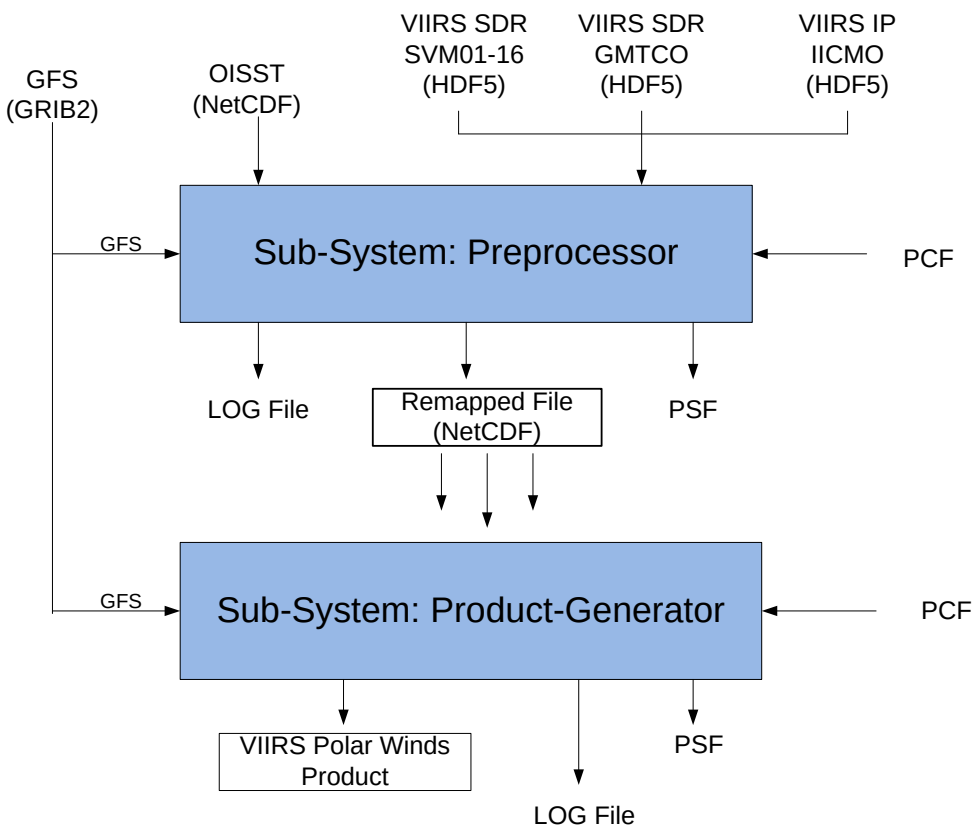
- Each production unit has a Perl wrapper script that drives the processing algorithms.

- The Preprocessor unit processes a list of granules from a polar pass to generate the remapped data over the polar region.

VPWPS
System Level Interfaces(cont.)

- The Product Generator unit retrieves the VPW products from the remapped triplet data set.
- The system PCFs are generated by NDE PGM containing all the input file locations and parameters needed by the VPWPS subsystems.
- Each unit, in turn, produces a PSF file containing information about the system product file status.

VPWPS
System Level Diagram



VPWPS
System Driver Scripts

- The system driver scripts are the top level script called by NDE PGM to invoke the production unit processing of an instance.
  - » There are no hard coded paths in the scripts. All needed information regarding locations of files will come through the system PCFs.
  - » All Perl library function calls and system calls have their return values checked and errors trapped so the exits are graceful and informative.
  - » All standard out and standard error is directed to a single log file that can be read by NDE for obtaining any error or warning messages.
  - » The driver scripts translate the low-level program errors into a high-level numerical value expected by the PGM (0 = success, 1=failure).
  - » The driver scripts set up the local environments (i.e. create unit PCFs, symbolic links, etc.) up front before proceeding to low-level processing.

VPWPS
Process Control File(PCF)

- A system PCF is generated by NDE PMG for each VPWPS unit run. It contains:
  - Job coverage start date and time string (*yyyymmddhhmmsss*), which is also image time of the first granule or polar pass).
  - Job coverage end data and time string (*yyyymmddhhmmsss*), which is also the image time of the last granule or polar pass.
  - Input data file names and locations for current process instance.
  - Ancillary data files and location for framework product generation (template PCF files, algorithm coefficient files, *etc.*)
  - Path to the executables and scripts.
  - Directories for intermediate output files.

- An example of the system PCF will be given in Unit Test section.

VPWPS
Product Status File(PSF)

- A system PSF is produced from each VPWPS unit run.
  - The Preprocessor system PSF contains the successfully generated remapped data file name and location.
  - The Product Generator system PSF contains the successfully generated VPW product file name and location.

- Examples:
  - Preprocessor system PSF content:

```
/data/data046/working/OUT/npp_viirs_remap/nps_VIIRS_2013122_0448_49_ALL_3305_3305.nc
```

  - Product Generator system PSF content:

```
/data/data046/working/OUT/viirs_results_pw/results/PAMV-VIIRS-CD-
NH_v1r1_npp_s201305020126490_e201305020448490_c201312021612210.nc
```

VPWPS
External-Level Input Data

File	Files /Day	Description	Size /File	Status
NPP VIIRS SDR granule data files In HDF5	6272 ~ 8064 (2 poles 14 orbits 16 M-bands 14 ~ 18 granules)	VIIRS SDR granule files containing science data (radiances) for 16 Moderate Resolution Bands over north and south polar region. Each polar pass has 14 ~ 18 granules SVM??_npp_d????????_t???????_e???????_b??????_c??????????????????????_noaa_ops.h5	38KB ~ 23MB	Dynamic input
VIIRS granule Geolocation data for M-bands in HDF5	392 ~ 504 (2 poles 14 orbits 14 ~ 18 granules)	VIIRS granule files containing geolocation data for 16 Moderate Resolution Bands over north and south polar region. GMTCO_npp_d????????_t???????_e???????_b??????_c??????????????????????_noaa_ops.h5	81MB	Dynamic input
VIIRS Cloud mask data in HDF5	392 ~ 504 (2 poles 14 orbits 14 ~ 18 granules)	VIIRS granule files containing cloud data over polar region. IICMO_npp_d????????_t???????_e???????_b?????_c??????????????????????_noaa_ops.h5	15MB	Dynamic ancillary input
Daily OISST in NetCDF	1	The 0.25 degree global AVHRR only Daily OISST. avhrr-only-v2.*YYYYMMDD*.nc	8.3 MB	Dynamic ancillary input
NWP GFS data In GRIB2	12	GFS 6-hour global forecast data at 0.5 degree in GRIB2 format from NCEP gfs.t*HH*z.pgrbfhh	55MB	Dynamic ancillary input

VPWPS
External-Level Output Data

File	Files /Day	Description	Size /File	Status
VIIRS Polar Winds product files	~28 (2 poles 14 orbits)	Framework generated VIIRS Polar Wind product for northern(NH) or southern(SH) polar region in NetCDF format. PAMV-VIIRS-CD-NH_v1r1_npp_s*{starttime}*_e*{endtime}*_c{*creattime*}.nc PAMV-VIIRS-CD-SH_v1r1_npp_s*{starttime}*_e*{endtime}*_c*{creattime}*.nc Where, *{starttime}* is the first image granules list's pole-pass-over time. *{endtime}* is the last image granule list's pole-pass-over time. *{creattime}* is the product generation time. Time string format is *YYYYMMDDhhmmsss*	Vary (5MB ~ 8 MB, depending on the number of targets found)	Output
Product Status files	~28	Product file list contain successfully generated VIIRS Polar Winds product files and the remapped input files(if any).	<1 KB	Output
Processing log files	~28	Processing log containing the standard output and return status of the VPWPS instance run.	Vary (depending on the number of targets found)	Output

Where, ,
{starttime} is the first granule time in the granule list .
{endtime} is the last granule time in the granule list.
{creattime} is the product generation time.
Time string format is *YYYYMMDDhhmmsss*

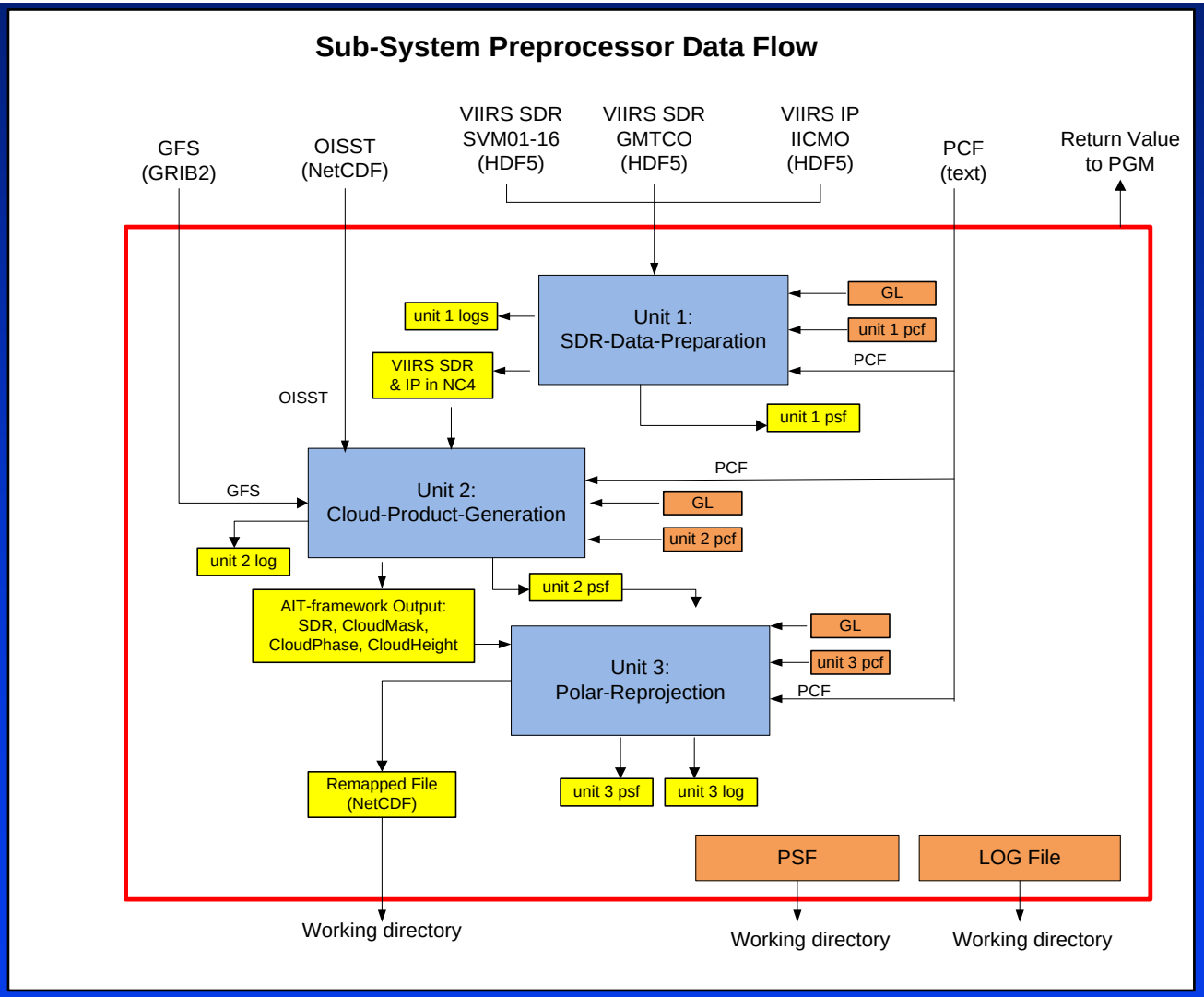
YYYY = 4 digit year (i.e. 2013)
MM = 2 digit month (i.e. 05)
DD = 2 digit day of the month (i.e. 25)
hhmmss = 2 digit of hour, 2 digit minute, and 3 digit second, each

VPWPS
Sub-system: Preprocessor

Three processing units:

1) SDR Data Preparation
2) Cloud Product Generation
3) Polar Re-projection

VPWPS
Sub-system: Preprocessor

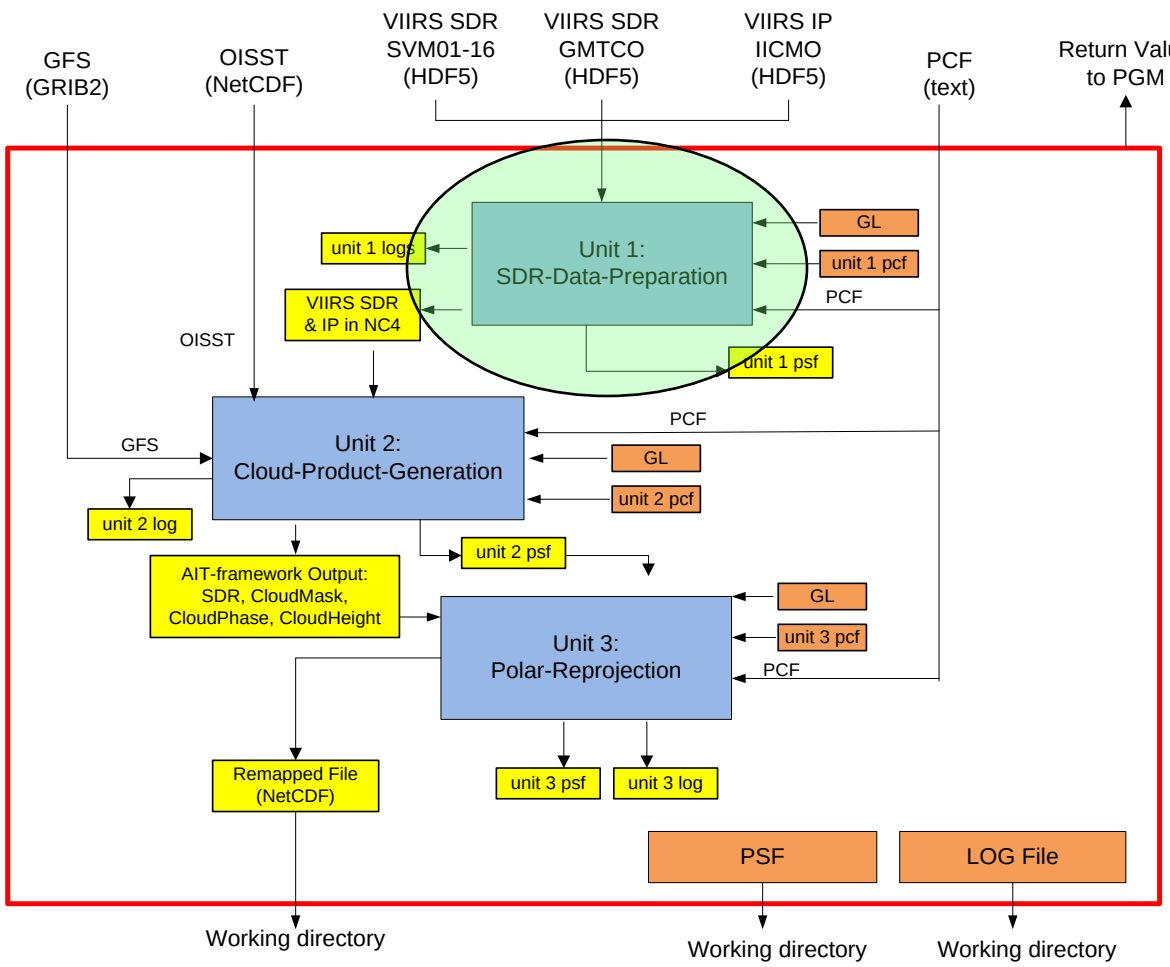


VPWPS
Sub-system: Preprocessor

Script Execution Tree

		VPW_REMAP.pl (System driver script for Preprocessor) ➢ Setup_links_for_static_ancillary.pl
⇩	Unit 1	➢ NVPWPS_Preproc_SRD.pl (SDR Data Preparation Unit) • mender.py (sub-unit SDR Gap-Filling) o viirsmend.py • update_pre_process_status.pl • convert_viirs_hd5_2_nc4.pl (sub-unit SDR Converter) o (h5augjpss) o get_sdr_scan_time.exe
⇩	Unit 2	➢ NVPWPS_Preproc_Clouds.pl (Clouds Product Generation Unit) • update_pre_process_status.pl • get_cfg_pcf_for_clouds.pl (sub-unit Framework CFG-PCF Generator) • pcf_framework.exe (sub-unit Framework Cloud Product Generation) o run_wgrib.pl
⇩	Unit 3	➢NVPWPS_Preproc_Remap.pl (Polar Re-Projection Unit) • get_remap_binary.pl (sub-unit Granule Remapper) o get_data_nc.exe o ms2gt -- fornav, gridsize, ll2cr • convert_remap_bin2nc.pl (sub-unit Granule Combiner) o combine_remap_binary.exe

Preprocessor
Unit 1: SDR Data Preparation



Preprocessor
Unit 1: SDR Data Preparation

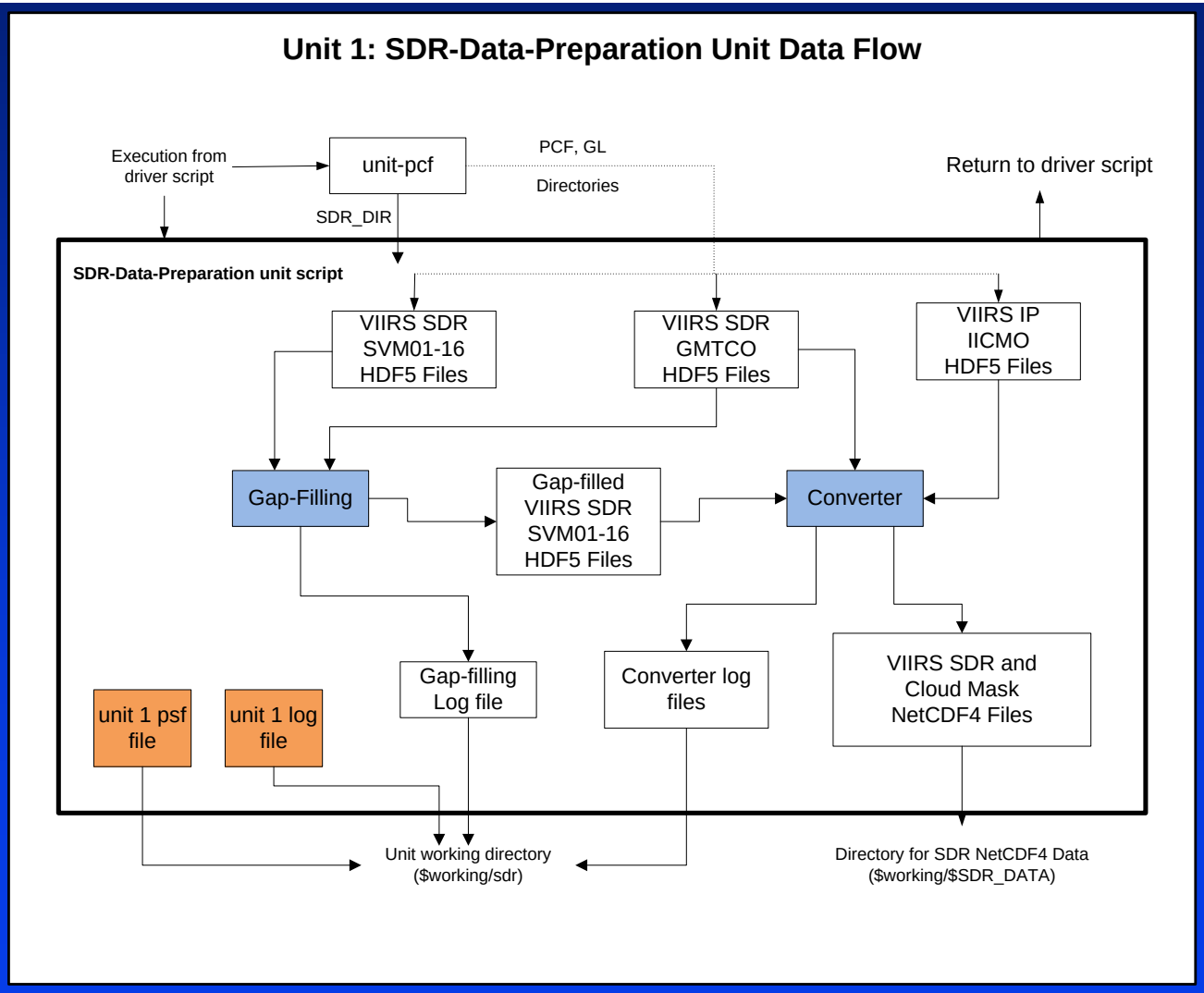
In this unit, all VIIRS SDR granules on the current image are processed with the Gap-filling utility for removing the gaps on the granule edges, followed by the converting of file format from HDF5 to NetCDF4.

Two sub units:

1.1) VIIRS SDR Gap Filling

1.2) HDF5 to NetCDF4 Converter

Preprocessor
Unit 1: SDR Data Preparation



Preprocessor
Unit 1: SDR Data Preparation

Unit PCF

- A unit PCF generated by the system level driver script up front is passed to the unit driver script for preparation processing of the SDR data files for the current image.

- The unit PCF contains location of the system PCF and the granule list file, the directory settings of unit input and output data files

- An example is given below

```
OPS_CONFIG_FILE=/data/data046/mfan/working/VPW_REMAP.pl.PCF
LIST_FILE=/data/data046/mfan/working/GRANULE_LIST/granule_list
SDR_HDF_DIR=/data/data046/mfan/working/OUT/viirs_hdf5
SDR_NC_DIR=/data/data046/mfan/working/OUT/npp_viirs_sdr
```

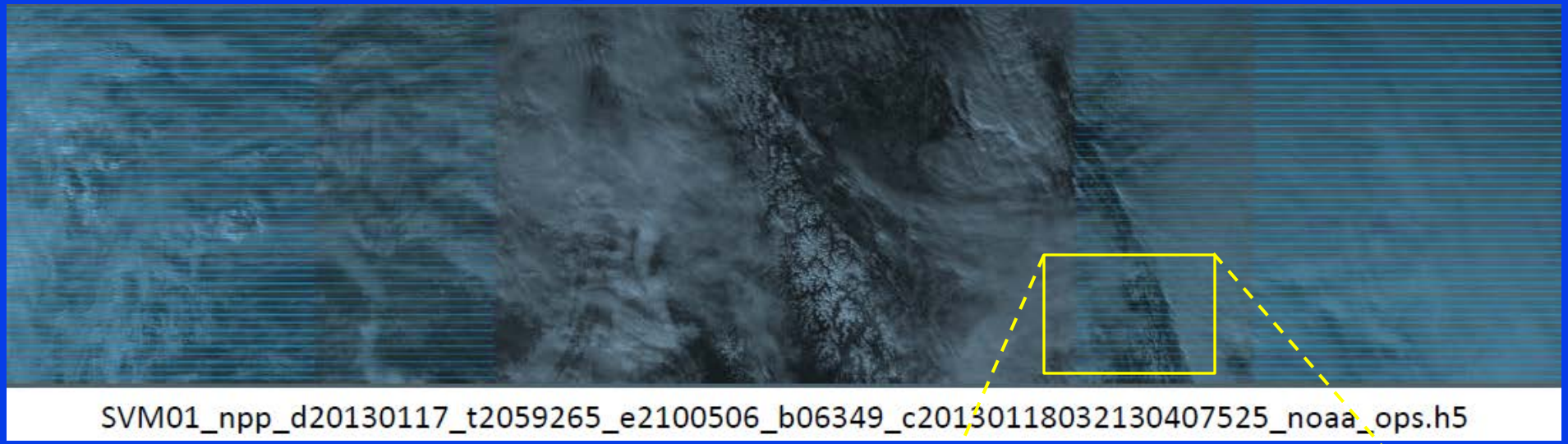
SDR Data Preparation
1.1 SDR Gap Filling

Unit level interfaces

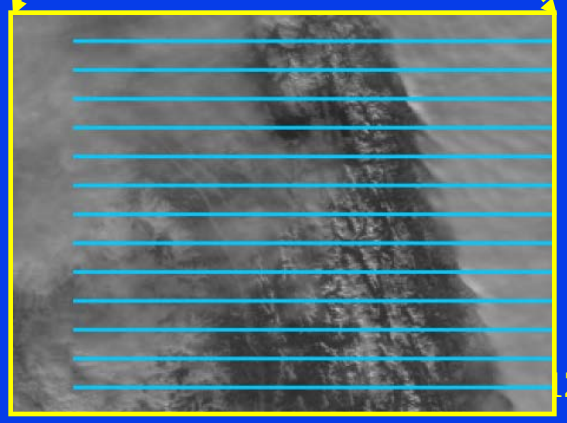
- A unit lever driver script is invoked by the system level script to start the unit processing.
- Information in the system PCF and unit PCF is used to set up the unit configuration, including creating the granule list status file for the current image, creating symbolic links for system files, etc.
- Gap-Filling python run scripts are called to fill the gaps on the VIIRS SDR granules listed in the system PCF. Currently, this includes all 16 M bands.
- A log file is saved for gap-filling processing.

SDR Data Preparation
1.1 SDR Gap Filling

Gaps on SDR granule



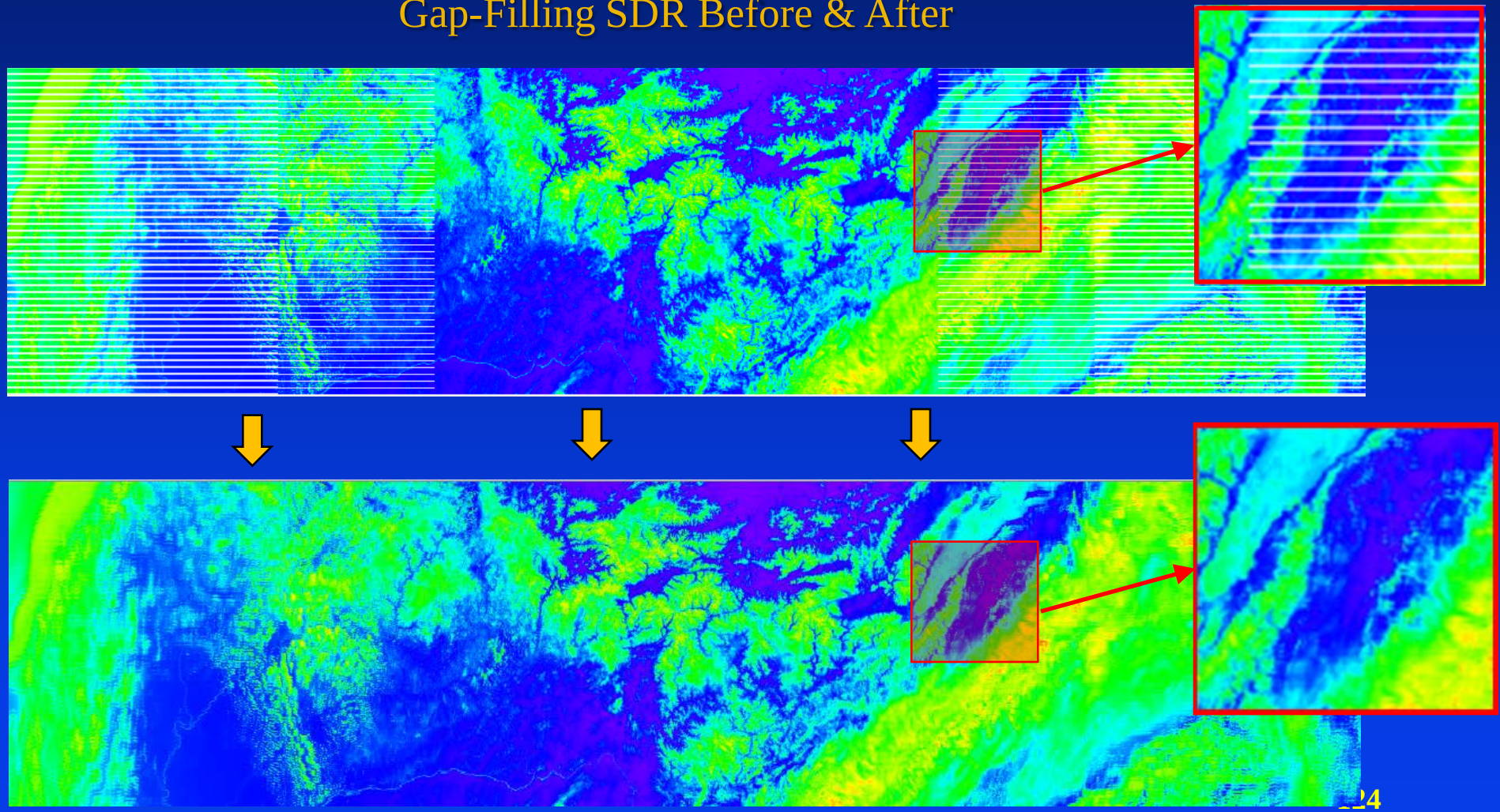
Gaps on the edges of a SDR granule, shows dropped portions of scan to reduce bow-tie effect



SDR Data Preparation

1.1 SDR Gap Filling

Gap-Filling SDR Before & After



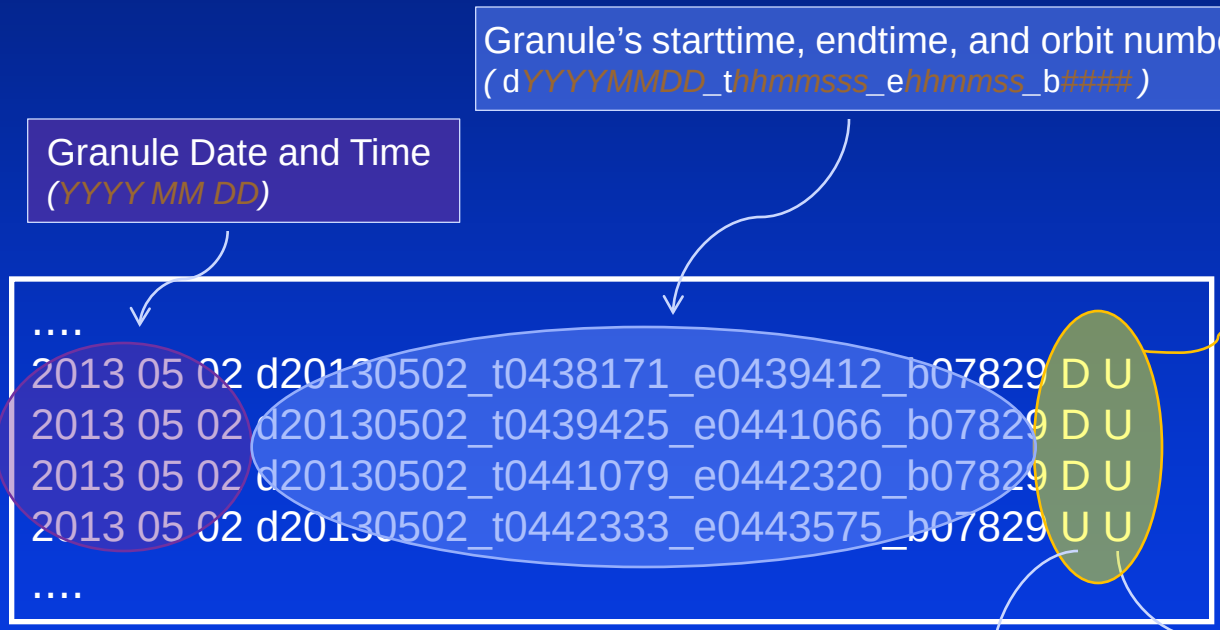
SDR Data Preparation
1.2 HDF5 to NetCDF4 Convertor

Unit level interfaces

- After Gap-filing of all the SDR granules, the JPSS HDF5 converter is invoked to convert the gap-filled SDR HDF5 files (SVM01-16) along with geolocation (GMTCO), VIIRS cloud mask IP (IICMO) files into NetCDF4 format.
- The granule list processing status file is updated after the conversion of each VIIRS granule.
- A PSF file containing the filenames of all converted VIIRS data files is generated.
- A log file is saved for each granule list for the conversion processing.

Preprocessor
SDR Data Preparation Unit

Granule List Processing Status File

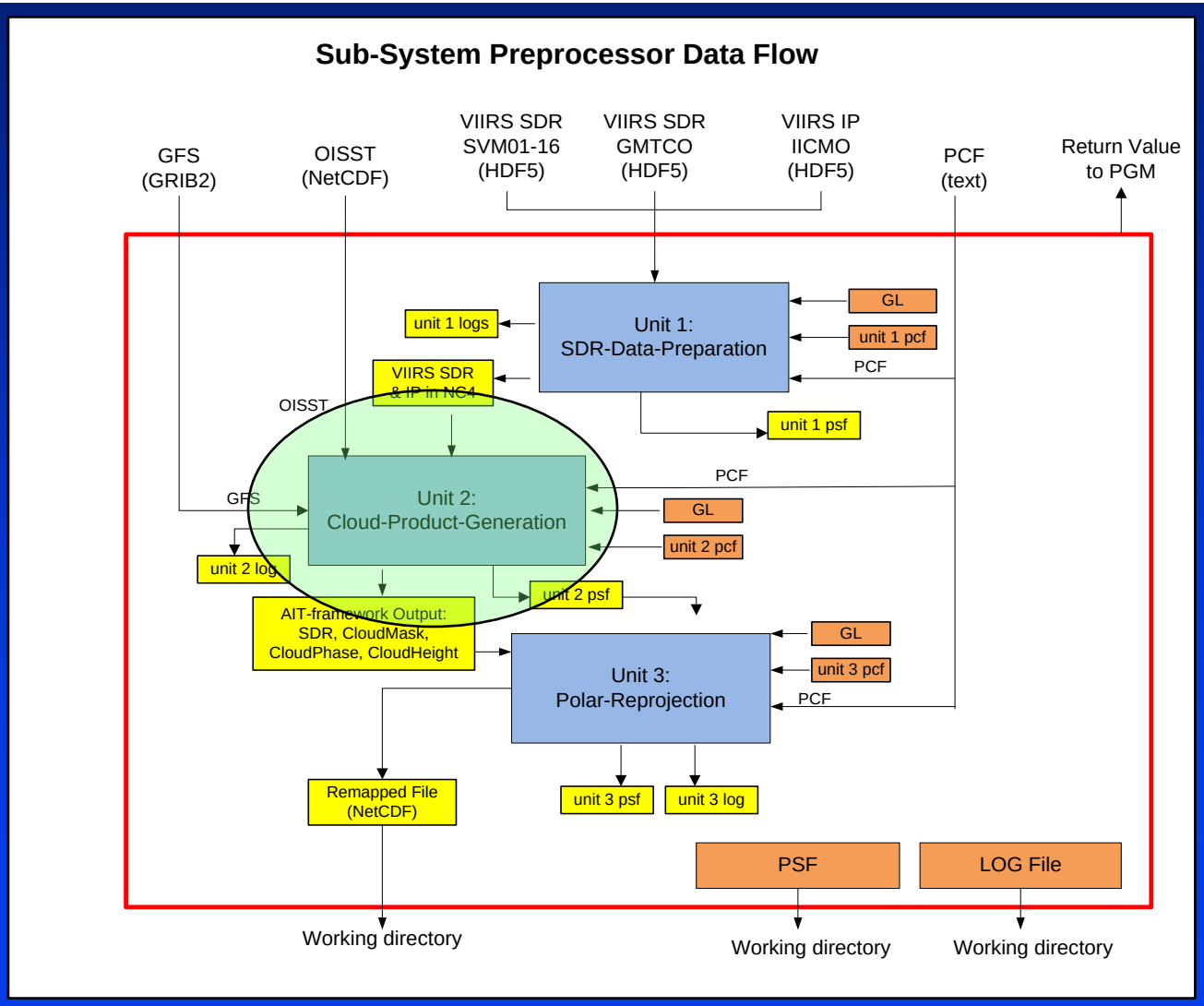


Preprocessor
SDR Data Preparation Unit

Input and Output Data

File	Input /Output	Data description	Format
SDR HDF5 files	Input	VIIRS SDR granule files on the granule list in system PCF. Total of 14~18 files for each VIIRS M-band on a polar pass/image	HDF5
VIIRS granule geolocation file for M-band HDF5 files	Input	Geolocation data files for SDR granule data. Total of 14~18 files	HDF5
Unit PCF	Input	Contains the setup of output directory, log directory, etc.	ASCII
Granule List Process Status file	Input /Output	For each image, a granule list process status file lists all the SDR granule name for the current instant run	ASCII
Gap-filled SDR NetCDF files	output	Gap-filled and converted SDR in NetCDF4. Total of 14~18 files	NetCDF4
Unit PSF	Output	Each file contains the time string of last processed GVAR image of the domain.	ASCII
Log file	Output	Contains processing log for the conversion processing.	ASCII

Preprocessor
Unit2 : Cloud Product Generation



Preprocessor
Cloud Product Generation Unit

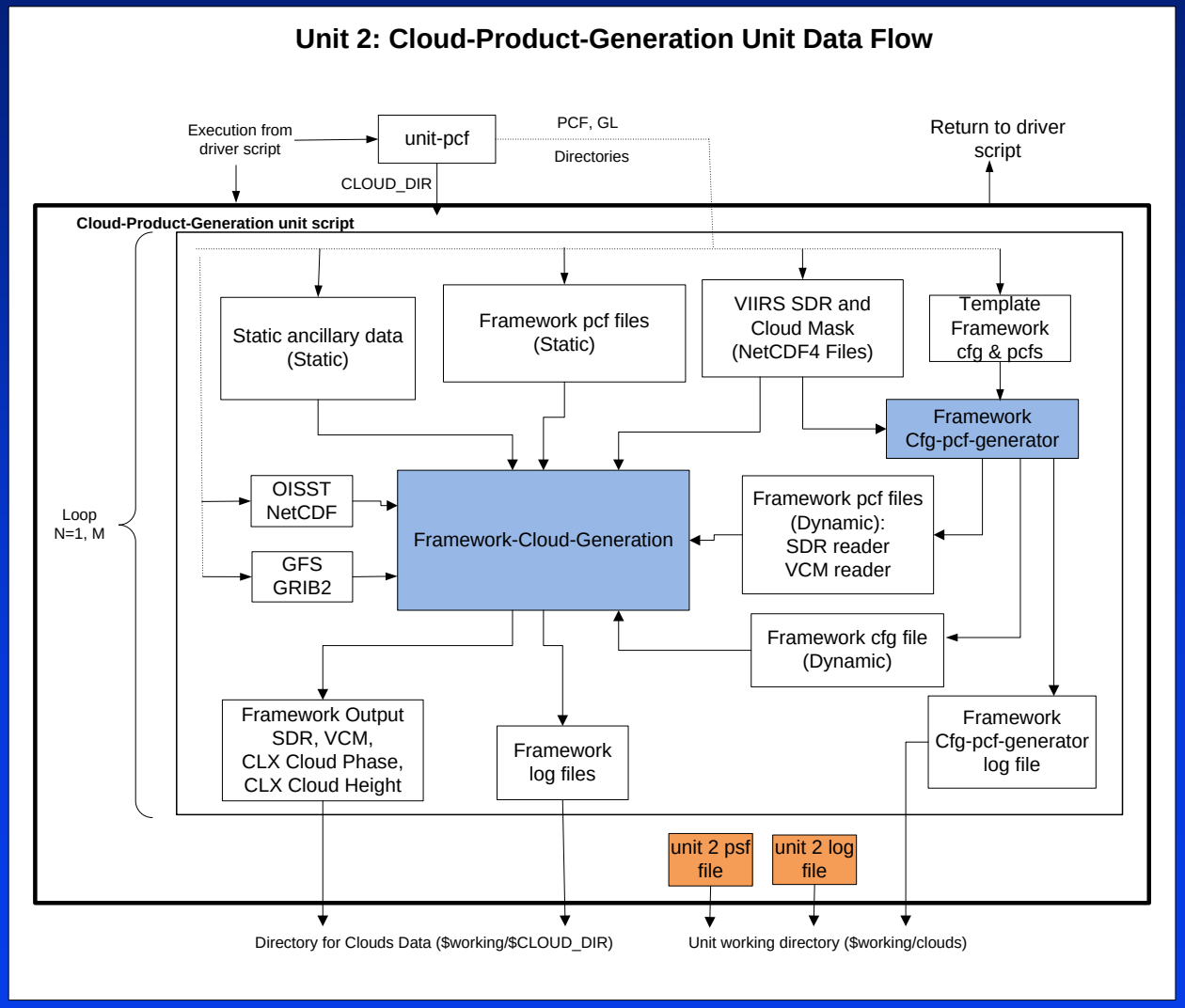
In this unit, the VIIRS cloud products are generated for each granule on the selected polar pass.

Two sub units:

2.1) Framework CFG-PCF Generator

2.2) Framework Cloud Product Generation

Preprocessor
Cloud Product Generation Unit



Preprocessor
Cloud Product Generation Unit

- A unit PCF generated by the system level driver script is passed to the unit driver script in order to process the cloud products for all granules on the selected polar pass.
- The unit PCF contains the location of the system PCF and the directory settings of unit input and output data files
- An example is given below

```
OPS_CONFIG_FILE=/data/data046/mfan/working/VPW_REMAP.pl.PCF
LIST_FILE=/data/data046/mfan/working/GRANULE_LIST/granule_list
SDR_NC_DIR=/data/data046/mfan/working/OUT/npp_viirs_sdr
CLOUD_DIR=/data/data046/mfan/working/OUT/viirs_results_clouds
```

Cloud Product Generation

2.1 Framework CFG-PCF Generator

Unit Level Interfaces

- A unit PCF generated up front at the system level is used for setting up the local environment, including creating symbolic links for AIT framework* ancillary data, input and output directories.
- The AIT framework overwritten PCFs and CFGs are generated from the template files before the Cloud product processing is triggered.
- A log file is generated for each run.

*AIT Framework provides an integrated processing environment for cloud and winds product generation. Details will be presents on slides 71-79.

Cloud Product Generation

2.2 Framework Cloud Product Generation

Unit Level Interfaces

- The AIT framework is invoked to produce the cloud products (Cloud Height, Cloud Phase) from VIIRS SDR data.
- VIIRS SDR data and VIIRS Cloud mask IP data for the granules are output in the framework product file format for the remapping processing unit.
- The cloud product generation status in the granule list status file is updated after the cloud product is generated for each granule.
- A PSF is generated containing all framework cloud product filenames for each granule along with SDR data and VIIRS cloud mask IP data.
- A log file is generated for the unit run.

Cloud Product Generation

2.2 Framework Cloud Product Generation

Update the status file

The cloud product generation status is updated for the processed granule in the granule list status file.

An example is given below.

```
2013 05 02 d20130502_t0438171_e0439412_b07829 D D
2013 05 02 d20130502_t0439425_e0441066_b07829 D D
2013 05 02 d20130502_t0441079_e0442320_b07829 D D
2013 05 02 d20130502_t0442333_e0443575_b07829 D D
2013 05 02 d20130502_t0443587_e0445229_b07829 D D
2013 05 02 d20130502_t0445241_e0446483_b07829 D D
2013 05 02 d20130502_t0446495_e0448137_b07829 D D
2013 05 02 d20130502_t0448149_e0449391_b07829 D U
2013 05 02 d20130502_t0449403_e0451027_b07829 D U
2013 05 02 d20130502_t0451039_e0452281_b07829 D U
2013 05 02 d20130502_t0452293_e0453535_b07829 D U
2013 05 02 d20130502_t0453547_e0455189_b07829 D U
2013 05 02 d20130502_t0455201_e0456443_b07829 D U
2013 05 02 d20130502_t0456455_e0458097_b07829 D U
2013 05 02 d20130502_t0458110_e0459351_b07829 D U
```

granule just being processed

Changed from 'U' to 'D'

Preprocessor
Cloud Product Generation Unit

Input Data Files

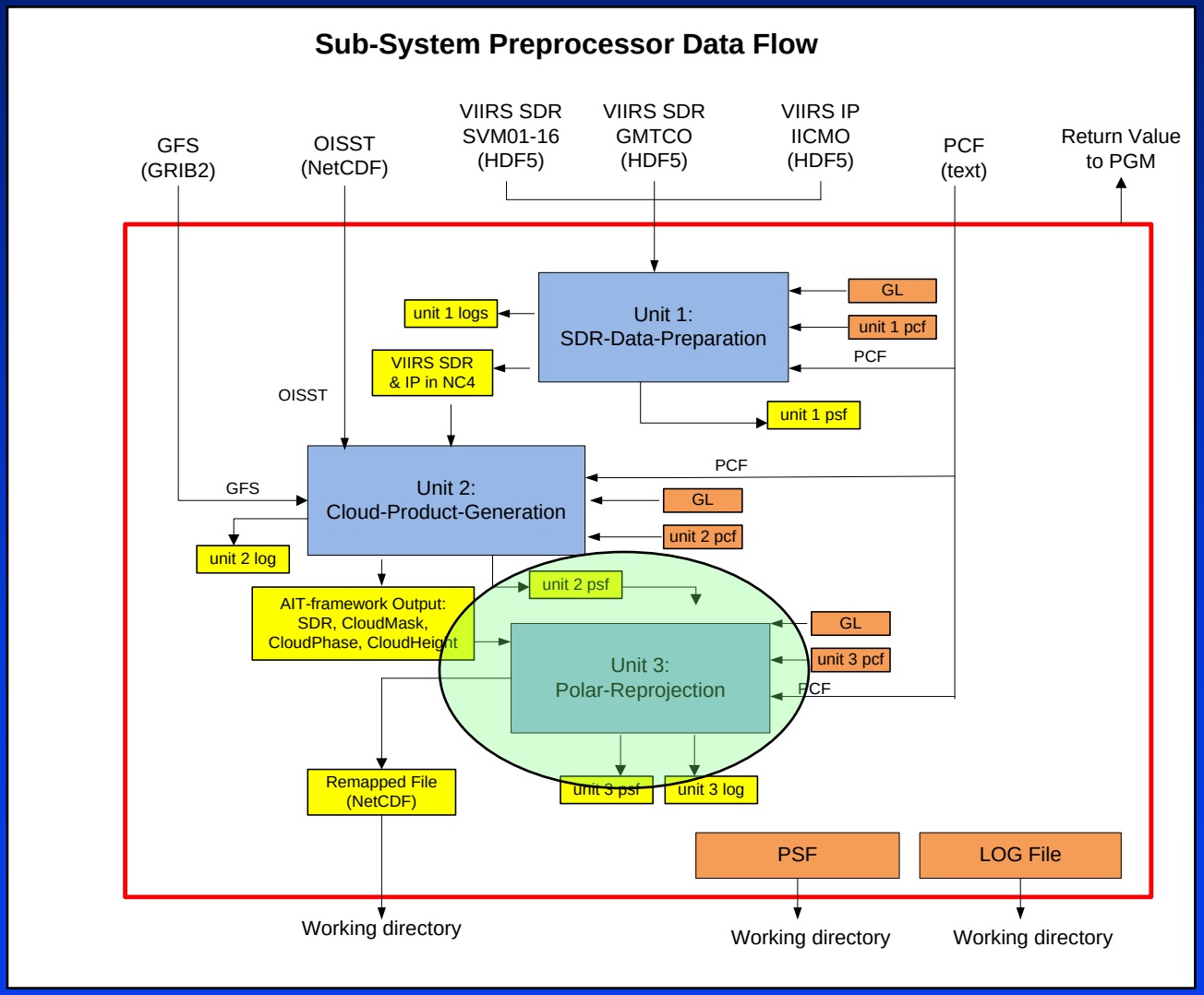
File	Input /Output	Data description	Format
VIIRS SDR granule data files for 16 M-Bands	Input	Gap-filled and converted SDR granule data files for 16 M bands.	NetCDF
VIIRS geolocation data file for M-bands	Input	Geolocation data files for SDR granule data converted from HDF5.	NetCDF
VIIRS Cloud Mask	Input	VIIRS Cloud mask IP data file converted from HDF5.	NetCDF
System PCF	Input	Contains the setup of status file, working directory, log file, etc.	ASCII
Unit PCF	Input	Unit configuration file containing directory setting for input and output data	ASCII
Framework product template files of PCFs and CFGs for cloud product	Input	Template files used to generate overwritten PCFs and CFGs for framework cloud product generation	ASCII
Daily OISST	Input	Contains Reynolds OISST daily analysis at 0.25 degree resolution	NetCDF
GFS data	Input	NCEP GFS model data in grib2 format at 0.5 degree resolution	GRIB2
Framework ancillary data	Input	Static data for framework product run, including land mask, surface emissivity data, CRTM coefficients , etc	NetCDF /Binary
Granule List Process Status file	Input /Output	Process status of each granule for the current processed image	ASCII

Preprocessor
Cloud Product Generation Unit

Output Data Files

File	Input /Output	Data description	Format
VIIRS channel Reflectance/BT/Radiance output from framework	Output	VIIRS SDR data in framework output format for each granule, which contains channel Refl/BT/Radiance and the geolocation data, etc.	NetCDF
VIIRS Cloud Mask output from framework	Output	VIIRS Cloud Mask in framework output format for each granule, which contains framework defined cloud mask and flags, etc.	NetCDF
Cloud Height product files	Output	The CLARX Cloud Height products in framework NetCDF4 for each granule	NetCDF
Cloud Phase product files	Output	The CLARX Cloud Phase products in framework NetCDF4 for each granule	NetCDF
Granule List Process Status file	Input/ Output	Process status of each granule for the current processed image	ASCII
Unit PSF	Output	Contains the all framework output files from Cloud product run, including SDR, VIIRS Cloud Mask, Cloud Height, Cloud Phase files of each granule.	ASCII
Log files	Output	Contain the unit processing log of each run.	ASCII

Preprocessor
Unit3: Polar Re-projection



Preprocessor
Polar Re-projection Unit

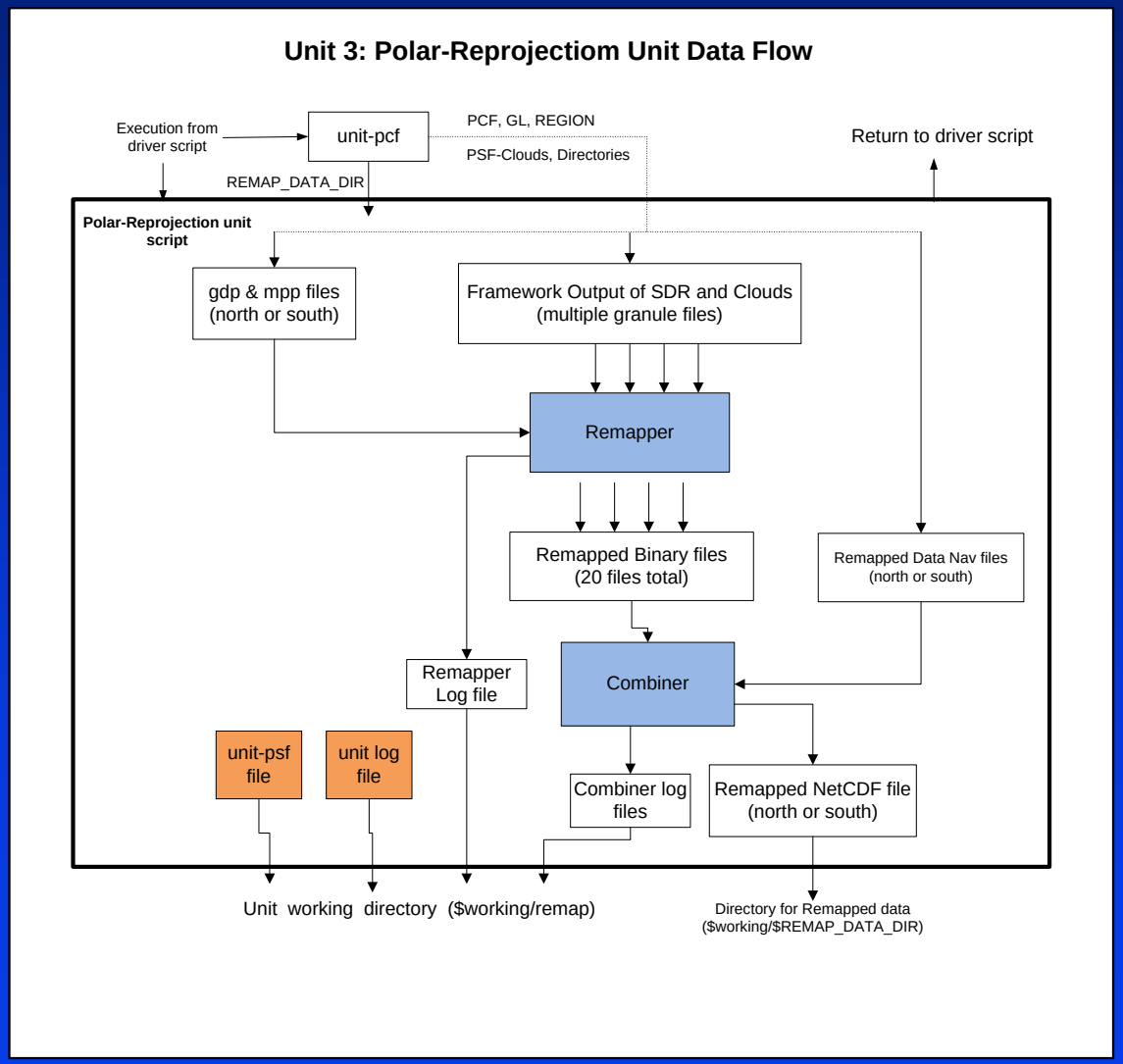
In this unit, all selected variables are extracted from the framework output granule product files, remapped to polar-stereographic projection, and combined into one NetCDF file for current image

Two sub units:

3.1) Remapper

3.2) Combiner

Preprocessor
Polar Re-projection Unit



Preprocessor
Polar Re-projection Unit

- A unit PCF is generated by the system lever driver script and passed to the unit driver script.
- The PCF contains the setup of the unit environment
  - System PCF file and location
  - Granule data file lists of current polar pass
  - Output directory for the remapped data
  - Date and Time over pole for current pass

```
OPS_CONFIG_FILE=/data/data046/mfan/working/VPW_REMAP.pl.PCF
LIST_FILE=/data/data046/mfan/working/GRANULE_LIST/granule_list
REGION=NORTH
REMAP_DATA_DIR=/data/data046/mfan/working/OUT/npp_viirs_remap
PSF_Clouds=/data/data046/mfan/working/clouds/NVPWPS_Preproc_Clouds.pl.PSF
Time_Over_Pole=/data/data046/mfan/working/remap/time_over_pole.txt
```

Polar Re-projection Unit
3.1 Granule Remapper

Unit Level Interfaces

- A unit PCF generated up front at the system level is used for setting up the local environment for the unit.
- For each polar granule on the list, a number of preselected variables are extracted from the framework output files (SDR data file, Cloud product files) into flat arrays. Each extracted variable data array is saved into one data file. Currently, 24 variables are being used by the VPW product processing.
- The swatch-to-grid tool (MS2GT) is used to remap each of the extracted variable data file onto PS projection map grids of 3305x3305.
- A log file is generated for each run.

Polar Re-projection Unit
3.2 Granule Combiner

Unit Level Interfaces

- For each selected variable, the data arrays are extracted from the remapped granule data files and merged into one data array on the fixed map grids of 3305x3305 over the polar region.
- For easy data handling in the Winds product generation, the selected variable data arrays are written into one single NetCDF file which will be used as input for the VPW generation.
- A PSF file is created which contains the remapped filename and location.
- A log file is generated for each run.

Preprocessor
Polar Re-projection Unit

Input Data Files

File	Input /Output	Data description	Format
VIIRS channel Reflectance/BT/Radiance output from framework	Input	VIIRS SDR data in framework output format for each granule, which contains channel Refl/BT/Radiance and the geolocation data, etc.	NetCDF
VIIRS Cloud Mask output from framework	Input	VIIRS Cloud Mask in framework output format for each granule, which contains framework defined cloud mask and flags, etc.	NetCDF
Cloud Height product files	Input	The CLARX Cloud Height products in framework netCDF4 for each granule	NetCDF
Cloud Phase product files	Input	The CLARX Cloud Phase products in framework netCDF4 for each granule	NetCDF
System PCF	Input	Contains the setup of status file, working directory, log file, etc.	ASCII
Unit PCF	Input	Unit configuration file containing directory setting for input and output data	ASCII
Map projection parameter files	Input	Contains map projection parameters and needed by the remapping and gridding processing.	ASCII
Cloud product generation unit PSF file	Input	Contains list of outputs (all granules) of SDR, VIIRS Cloud Mask, CLARX Cloud Phase and Height files (from Cloud product generation unit)	NetCDF
Geolocation data for the Polar Stereographic map grid points	Input	Contains latitude and longitude of the PS map of 3305x3305 for north and south polar regions	NetCDF
Granule List Process Status file	Input	Process status of each granule for the current processed image	ASCII

Preprocessor
Polar Re-projection Unit

Output Data Files

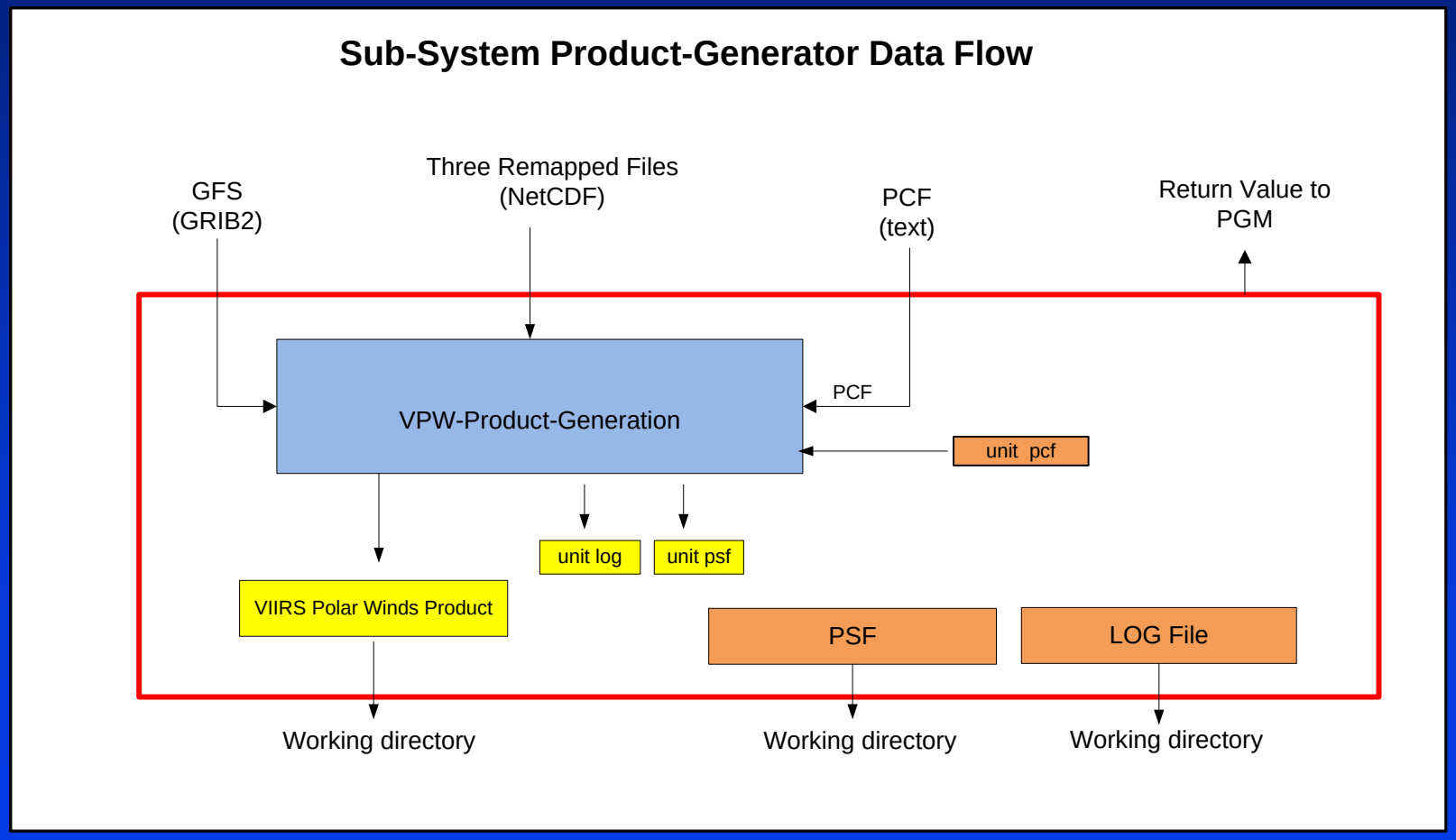
File	Input /Output	Data description	Format
Remapped file for current image	Output	Contains the selected variables(i.e. VIIRS channel Refl/BT/Rad, Cloud mask, Cloud Phase and type, etc.) from all listed granules on current image	NetCDF
Unit PSF	Output	Contains the unit output remapped filename	ASCII
Unit Log files	Output	Contain the unit processing log of each run	ASCII



VPWPS
Software Architecture

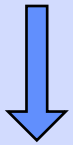
Sub-System:
Product Generator

VPWPS
Sub-System: Product Generator

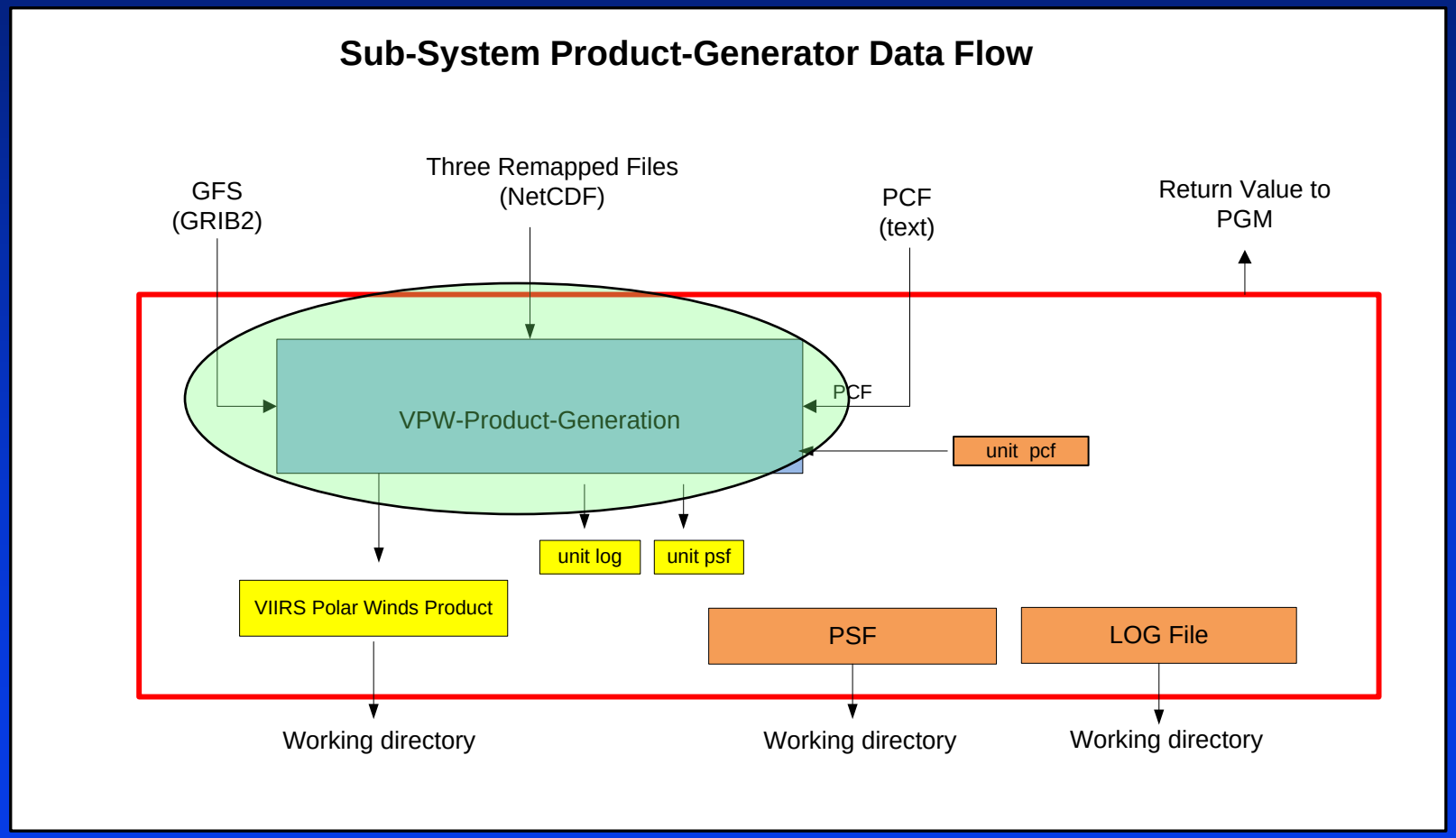


VPWPS
Sub-system: Product Generator

Script Execution Tree

	VPW_PRODUCT.pl (System driver script for Product Generator) ➢ Setup_links_for_static_ancillary.pl
	➢NVPWPS_Proc_PW.pl (sub-unit Framework CFG-PCF Generator) • pcf_framework.exe (sub-unit Framework-Winds-Generation) o run_wgrib.pl

Product Generator
VPW Product Generation Unit



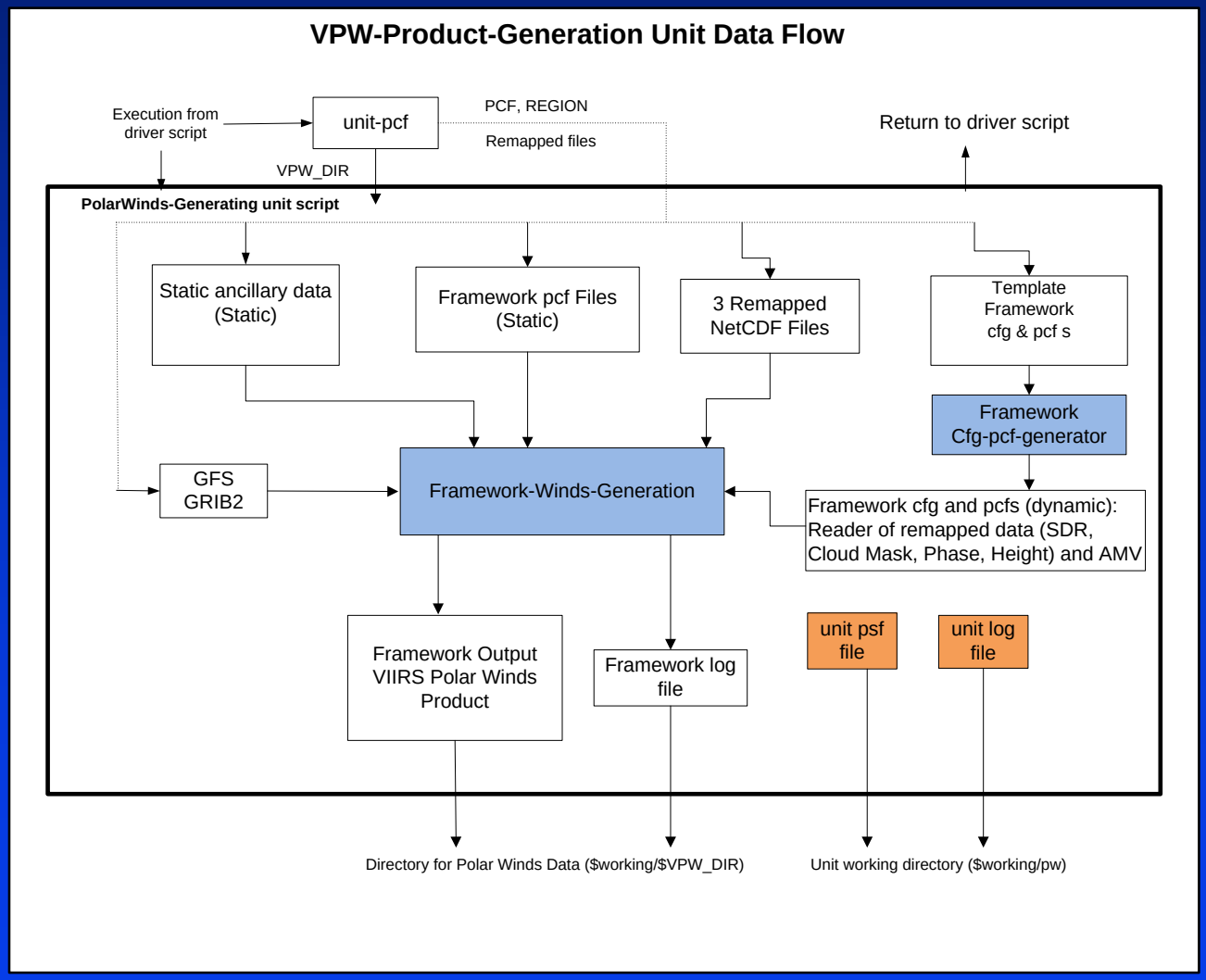
Product Generator
VPW Product Generation Unit

In this production unit, VIIRS polar winds products are retrieved from the remapped triplet images in AIT framework

Two sub units:

1) Framework PCF-CFG Generator
2) Framework Product Generation

Product Generator
VPW Product Generation Unit



Product Generator
VPW Product Generation Unit

- A unit PCF generated up front at the system level is used for setting up the local environment, including creating symbolic links for AIT framework ancillary data and output directory.
- It contains the location of the system PCF and the directory settings of unit input and output data files
- An example is given below

```
OPS_CONFIG_FILE=/data/data046/mfan/working/VPW_PRODUCT.pl.PCF
REGION=NORTH
VPW_DIR=/data/data046/mfan/working/OUT/viirs_results_pw
PSF_Remap_before=/data/data046/mfan/working/psf_remap_1
PSF_Remap_current=/data/data046/mfan/working/psf_remap_2
PSF_Remap_after=/data/data046/mfan/working/psf_remap_3
```

VPW Product Generation
Unit 1: Framework CFG-PCF Generator

Unit Level Interfaces

- A unit lever driver script is triggered by the system driver script to start the unit processing.
- The AIT framework overwritten PCFs and CFGs are generated from the template files before the Winds product processing is triggered.
- A log file is generated for each run.

VPW Product Generation
Unit 2: Framework Winds Generation

Unit Level Interfaces

- The AIT framework is invoked to produce the VPW products from the remapped triplet images.
- The framework CFGs and PCFs (from unit 1) are passed to framework to run the VPW algorithm for generating the Winds products for the three overlapped polar passes.
- A unit PSF is generated containing the framework Winds product filename and location for the triplet image run.
- A framework processing log file is generated for each unit run.

VPWPS
Sub-system: Product Generator

Input Data Files

File	Input /Output	Data description	Format
Remapped file for the first image in the triplets	Input	Contains remapped data for the selected variables for Winds product generation for the first image.	NetCDF
Remapped file for the middle image in the triplets	Input	Contains remapped data for the selected variables for Winds product generation for the middle image.	NetCDF
Remapped file for the third image in the triplets	Input	Contains remapped data for the selected variables for Winds product generation for the third image	NetCDF
Map projection parameter files	Input	Contains map projection parameters and needed by the remapping and gridding processing.	ASCII
Framework product template files of PCFs and CFGs for winds product	Input	Template files used to generate overwritten PCFs and CFGs for framework winds product generation	ASCII
NWP GFS data	Input	NCEP GFS 6-hr model forecast data in grib2 format at 0.5 degree resolution	GRIB2
Framework ancillary data	Input	Static data for framework product run, including land mask, satellite instrument ancillary data, winds coefficients, etc.	NetCDF
System PCF	Input	Contains the setup of status file, working directory, log file, etc.	ASCII
Unit PCF	Input	Unit configuration file containing directory setting for input and output data	ASCII

VPWPS
Sub-system: Product Generator

Output Data Files

File	Input /Output	Data description	Format
VIIRS Polar Winds product file	Output	Contains all winds product variables and metadata for north or south polar region.	NetCDF
Unit PSF	Output	Contains the unit output Winds product filename.	ASCII
Unit Log files	Output	Contain the unit processing log of unit run.	ASCII

Outline

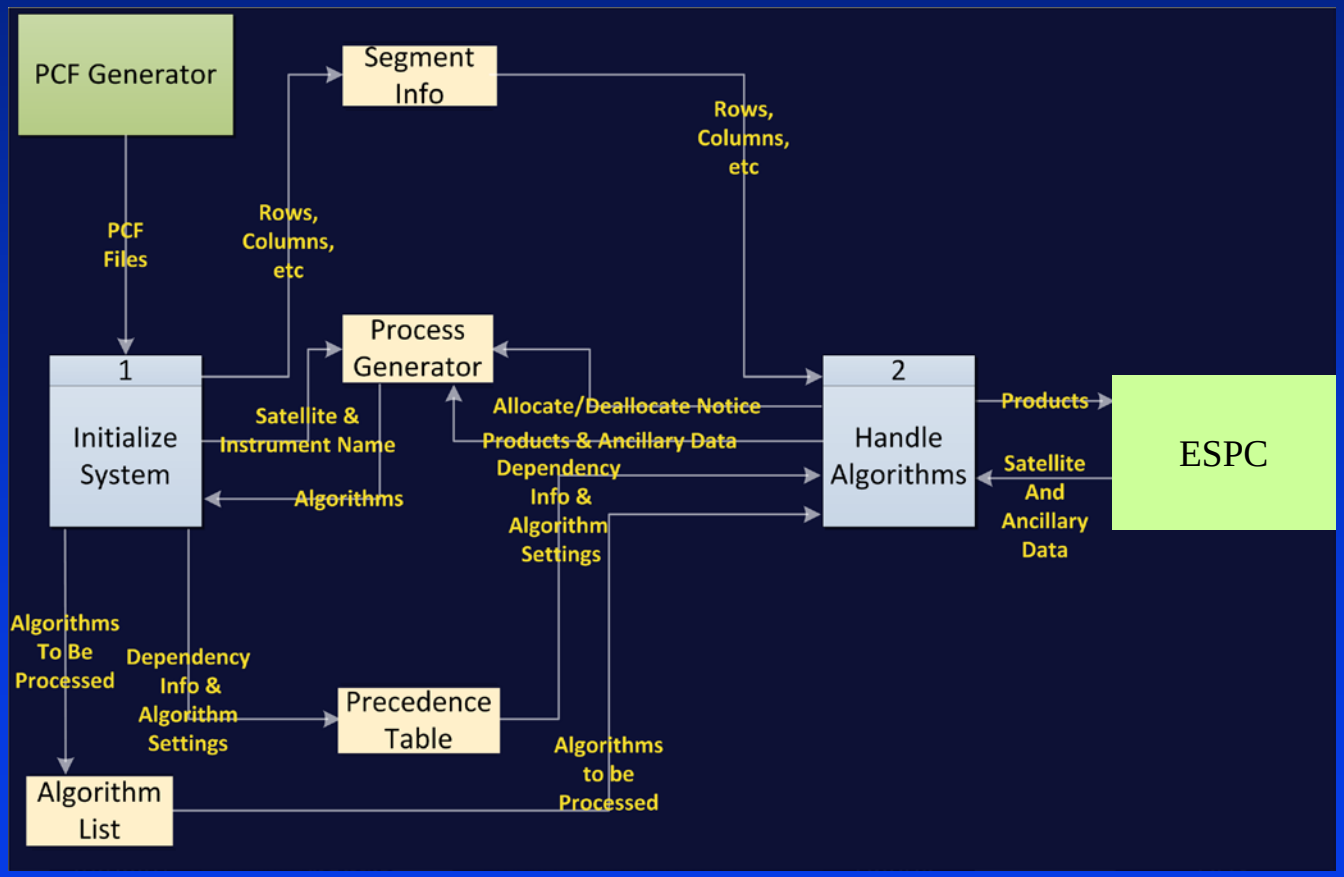
- VPWPS System Overview
- VPWPS Detailed Design
- AIT Framework
- Transition to Operations

VPWPS
AIT Framework

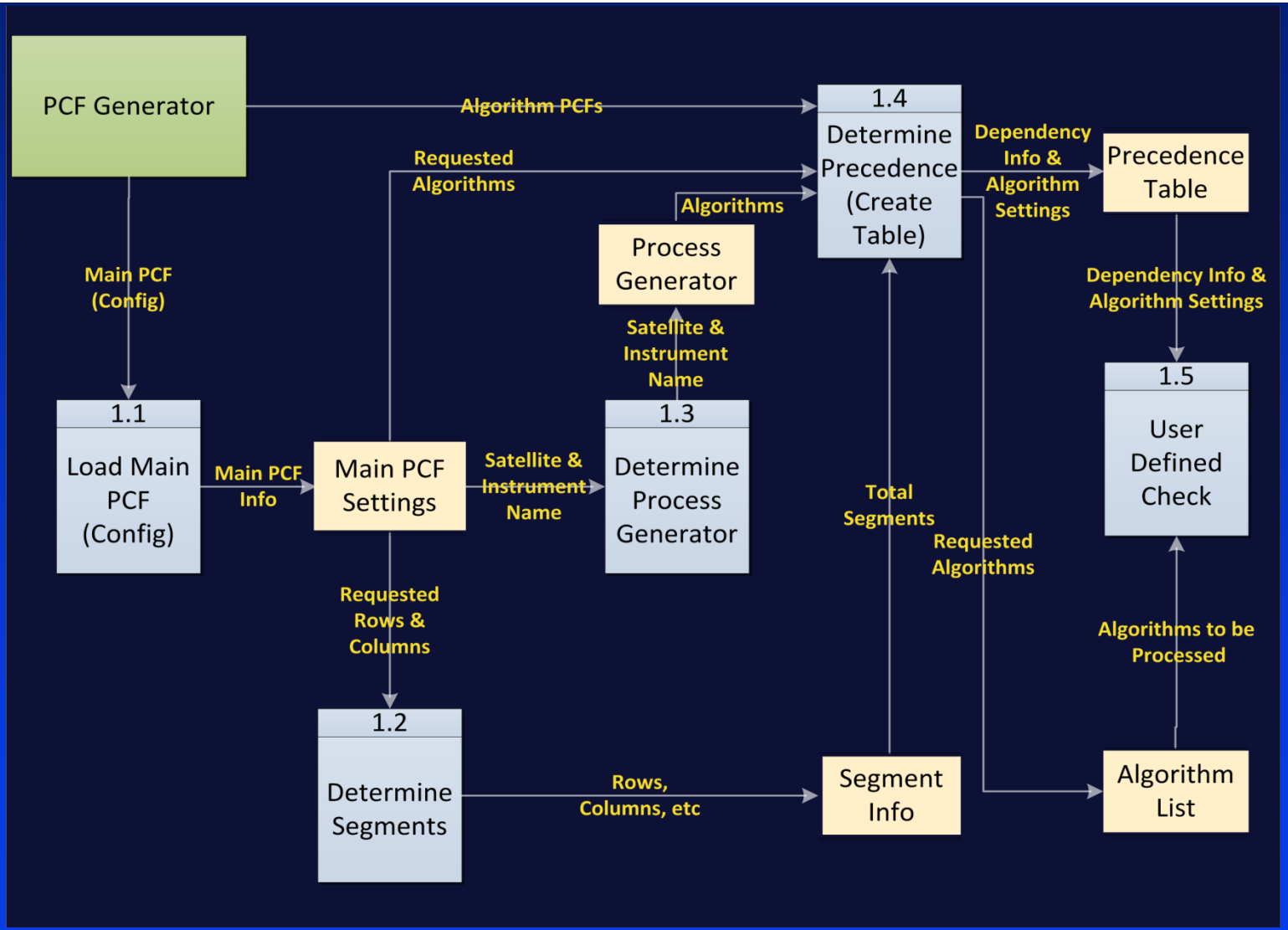
- AIT Framework provides an integrated processing environment for cloud and winds product retrieval processing.

- The following nine slides show the AIT framework unit detailed design, diagram, and data flow.

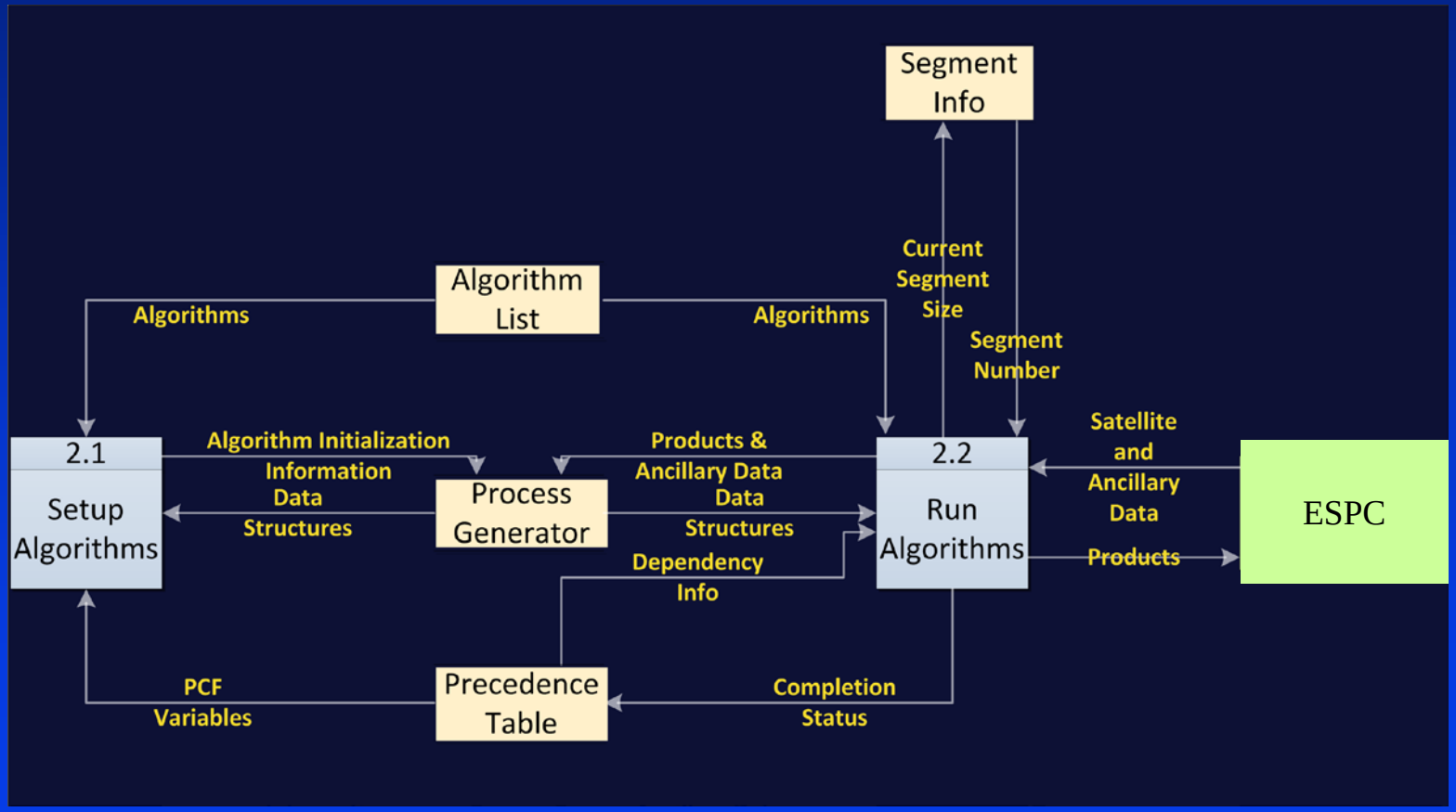
Framework
System Level



Framework Unit Level
1) Initialize System



Framework Unit Level
2) Handle Algorithms



Framework
Inputs

- **Production Control Files (PCF)**

- **Data**
  - » Radiance data
  - » Common ancillary data
  - » Specific algorithm data

Framework
PCF Files

- Production Control Files (PCF) contain the information required to run an algorithm
  - » Algorithm dependencies to determine product precedence
  - » Algorithm specific variables such as flags and thresholds
  - » Framework loads the contents of the PCF file when the algorithm has been flagged to run in the configuration file or if it is needed by something that has been flagged to run in the configuration file.

Framework
Dynamic Ancillary Data

Ancillary Data	Description	Filename	Size
CRTM	Community Radiative Transfer Model	N/A	N/A
NWP_GFS	NCEP GFS model data in grib2 format – 0.5x0.5 degree (720x361), 26 levels	gfs.tHHz.pgrbfhh	55 – 56MB
OISST_DAILY_QTRDEG	NCEP EMC Reynolds OISST daily analysis, 0.25 degree resolution	avhrr-only-v2.YYYYMMDD.nc	8305492 Bytes
SNOW_MASK_NWP	Snow/Ice mask, calculated from snow surface variable in the GFS grib file	N/A	N/A

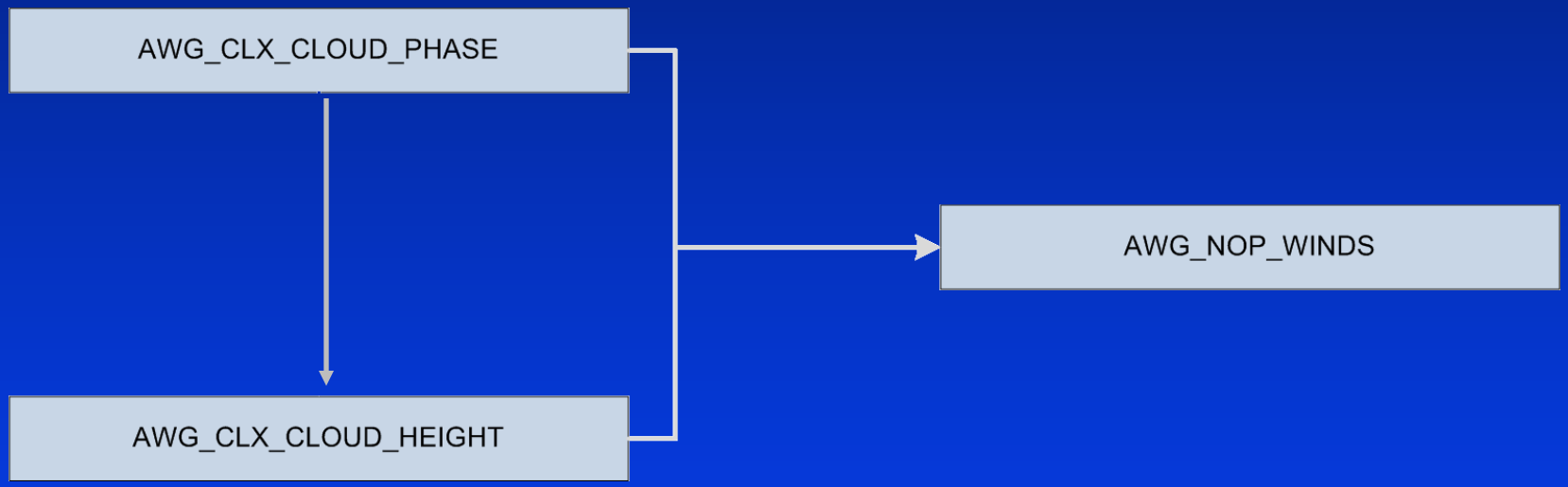
Framework
Static Ancillary Data

Ancillary Data	Description	Filename	Size
COAST_MASK_NASA_1KM	Global 1km land/water used for MODIS collection 5	coast_mask_1km.nc	890 MB
DESERT_MASK_CALCLTED	Desert mask calculated using LAND_MASK_NASA_1KM and SFC_TYPE_AVHRR_1KM	N/A	N/A
LAND_MASK_NASA_1KM	Global 1km land/water used for MODIS collection 5	lw_geo_2001001_v03m.nc	890 MB
SFC_ELEV_GLOBE_1KM	Digital surface elevation at 1km resolution	GLOBE_1km_digelev.nc	1843.2 MB
SFC_TYPE_AVHRR_1KM	Surface type mask based on AVHRR at 1km resolution	gl-latlong-1km-landcover.nc	890 MB
SFC_EMISS_SEEBOR	Surface emissivity at 5km resolution, climatology monthly	global_emiss_intABI_2005DDD.nc	693 MB x 12
GOESR_ABI_CHN7_EMISS	Pseudo channel 7 emissivity	N/A	N/A

Framework
CRTM Inputs

Filename	Description	Size
CloudCoeff.bin	Cloud coefficient data for CRTM	1654180 Bytes
AerosolCoeff.bin	Aerosol coefficient data for CRTM	5766260 Bytes
EmisCoeff.bin	Emissivity coefficient data for CRTM	1888256 Bytes
viirs_npp.SpcCoeff.bin	Space coefficient data of NPP VIIRS-M for CRTM	472 Bytes
viirs_npp.TauCoeff.bin	Tau coefficient data of NPP VIIRS-M for CRTM	105704 Bytes

Framework
Product Precedence



VPWPS
Software Architecture

Outline

- VPWPS System Overview
- VPWPS Detailed Design
- AIT Framework
- Transition to Operations

VPWPS
Error Handling

- All return values and exit status from executables, script functions, unit driver scripts, and system calls from scripts are checked;

- All errors or noteworthy conditions are trapped. Three common error codes are used in AIT framework product processing to categorize the severity levels:
  - 0 = NOTICE
  - 1 = WARNING
  - 2 = FATAL

  Messages are labeled and produced based on the error codes and are directed to log files.

VPWPS
Product Processing Monitoring

- The SDR processing status file contains information about the VIIRS SDR data preparation and cloud product generation status. It can be used for tracking the Preprocessor unit on the generating of the remapped data files.

- The unit PSFs provide information about the successfully generated product files from each unit run. The files can be made available to NDE monitoring system for tracking the processing steps and checking the availability of the product files.

- The Polar Winds product files contain algorithm specified metadata. Depending on the requirements, detailed product attributes, such as categorized QA flags and product statistic reports, are provided in the output files. The metadata can be extracted and used as input to the product quality monitoring tool at OPSO.

VPWPS
System Design Assumptions

- The VPWPS design assumes the NDE PGM will run VPWPS in a single working directory that is unique to that run.
- The system PCF will be made available to the system driver script locally in the working directory at run time.
- The NDE DHS must be able to run multiple instances of these units to process concurrently available input data.
- The VPWPS is designed to process the data from a list of VIIRS granules , or a set of remapped triplet images, that meet the criteria on data coverage and timeliness from the Winds product retrieval requirements.
  - The VPWPS relies on NDE on the data availability of the input data sets, as well as the dynamic ancillary data, such as GFS data , OISST, etc.
- The NDE DHS will perform all file management for VPWPS outside of the working directory where VPWPS runs.

VPWPS
Transition to NDE

- VPWPS has been fully tested in STAR development environment. The product processing time meets the latency requirements (<30 min). It is ready to run in NDE operational environment.
- All VPW final products will be generated by VPWPS in NDE DHS and will be distributed to NDE DDS
- Detailed information about the unit and subunit level data flow are documented in the VPWPS System Maintenance Manual (SMM) in the DAP delivery
- The VPWPS package will be:
  - » Delivered to NDE
  - » Integrated into the NDE DHS by the NDE team
  - » NDE will run VPWPS in production mode at NSOF

ARR Risks and Actions For VPW

- **ARR Risk 1:** The VPW processing system currently uses all M bands as inputs though the algorithms only use 5 of the bands
- **Risk Assessment:** Medium
- **Impact:** The Framework uses excess memory and the NDE moves more VIIRS data than it should.
- **Likelihood:** High
- **Mitigation:**
  - » Update the Framework to only ingest the channels that it needs
  - » Update the Gap Filling algorithm to only ingest the channels that it needs
- **Status:** Open

ARR Risks and Actions For VPW

- **ARR Risk 2:** The VPW processing system requires the IMS for the ancillary data snow mask
- **Risk Assessment:** Medium
- **Impact:** The VPW processing system will need to be updated when the global IMS is available
- **Likelihood:** Medium
- **Mitigation:**
  - » Use the NCEP snow mask until the global IMS is available. The requirements are still met
  - » Update VPW processing system when the global IMS is available
- **Status:** Open

Summary

- The software architecture design and interfaces of each component in VPWPS have been presented.
- The system data flow of each processing unit has been presented.
- The error handling and product generation information that can be used for VPWPS product processing monitoring have been presented.

Agenda

- Introduction
- Risks
- Requirements
- Algorithm Readiness
- Software Architecture
- Delivered Algorithm Package
- Risks and Actions
- Summary and Conclusions
- Appendix – VIIRS Polar Winds Unit Tests

Delivered Algorithm Package

Presented by

Aiwu Li

VPWPS DAP

- The VPWPS DAP is contained in a single tar file that has been compressed using gzip. It has the following name: VPW_PAMV_v1-0_20140328.tar.gz
- It is located on SADIE:
  /utilraid/data/users/mfan/VPW_20140328/VPW_PAMV_v1-0_20140328.tar.gz
- Note that the DAP file name complies with the DAP naming convention identified on page 7 of the DAP Standards.
- When ungzip'd and untar'd, there are 4 main subdirectories, CRTM tar file, and a README file:
  - *SOURCE/* - All Fortran 77/90, C/C++ code
  - *NVPWPS_OPS/* - All scripts and static system files
  - *DATA/* - All sample/test data
  - *DOC/* - All SPSRB and NDE documentation
  - REL-2.0.2.JCSDA_CRTM_gm.tar - CRTM 2.0.2 package
  - README - the README file for VPWPS

DAP Checklist

Required DAP Item	Location in Delivered DAP
Science algorithm source code, including make files and build scripts.	./SOURCE/AIT_framework/ ./SOURCE/remap/ ./SOURCE/extrcat_scan_time/
Source code of the third-party utility tools.	./SOURCE/Tools/
Test plans, test description, test procedures, and detailed performance testing results	./DOC/
Test input data, ancillary data(static and dynamic), and expected output data.	./DATA/dynamic_input/ ./DATA/static_input/ ./DATA/output/
Processing Scripts	./NVPWPS-OPS/scripts/
Template files for framework algorithm PCFs and processing configuration files.	./NVPWPS-OPS/PCF_Overwrites/ ./NVPWPS-OPS/config_templates/
Quality monitoring information (quality flags, quality flag values).	Section 6 of ./DOC/NVPWPS_SMM_1.0.docx
Production rule-set definitions.	./DOC/NVPWPS_Production_Rules.docx

DAP Checklist (cont.)

Required DAP Item	Location in Delivered DAP
Product file specifications – layout, content, and size.	Section 7 of ./DOC/NVPWPS_SMM_1.0.docx Section 1 of ./DOC/NVPWPS_EUM_1.0.docx
Data flow diagrams.	Section 7 of ./DOC/NVPWPS_SMM_1.0.docx
List of exit codes and their associated messages.	Section 6 of ./DOC/NVPWPS_SMM_1.0.docx
List of expected compiler warnings	Section 4 of ./DOC/NVPWPS_SMM_1.0.docx
Estimates of resources required for execution.	Section 2 of ./DOC/NVPWPS_SMM_1.0.docx
Algorithm Theoretical Basis Documents (ATBDs) or reference to where the ATBDs can be obtained.	./DOC/Algorithm_Theoretical_Basis_Document_VIIRS_Polar_Winds_032714.docx
Delivery Memo.	./DOC/Delivery_Memo To be sent via email at time of delivery to NDE
README text file.	./README

VPWPS Source Code

VPWPS Code Area	Language	Number of Files	Number of Lines
AIT Framework(including Cloud& Winds algorithm code)	Fortran 90	386	175903
AIT Framework(including Cloud& Winds algorithm code)	C	711	173108
VIIRS SDR gap-filling*	Python	2	326
VIIRS SDR data preparation* (granule remapper&combiner)	Fortran 90	2	1325
VIIRS SDR data preparation * (granule remapper&combiner)	C	1	455
Unit scripts and utilities	Perl	13	5503
Makefiles, compile scripts*	Make	168	7079

*Exclude third-party software source code.

VPWPS Documentation

- The VPWPS DAP will contains the following documents:
  - Delivery Memo (also included in email notification)
  - README
  - NVPWPS_SMM_1.0.docx
  - NVPWPS_EUM_1.0.docx
  - Algorithm_Theoretical_Basis_Document_VIIRS_Polar_Winds_032714.docx
  - NVPWPS_Production_Rules.docx

- The SMM, EUM, and ATBD contain unfinished sections that need to be completed by Integration Programmers (NDE) and the PAL (OSPO).
  - The document objects for sections have been highlighted in yellow

VPWPS Algorithm Readiness Summary

- The VPWPS software has been tested. The results have been presented.
- The DAP contents have been verified.
- DAP has been delivered to SADIE.
- VPWPS developers will work with NDE to address any issues that arise during integration and NDE system testing.

Agenda

- Introduction
- Risks
- Requirements
- Algorithm Readiness
- Software Architecture
- Delivered Algorithm Package
- Risks and Actions
- Summary and Conclusions
- Appendix – VIIRS Polar Winds Unit Tests

Risks and Actions Summary

Presented by

Walter Wolf

ARR Risks and Actions For VPW

- **ARR Risk 1:** The VPW processing system currently uses all M bands as inputs though the algorithms only use 5 of the bands
  - **Mitigation:**
    - Update the Framework to only ingest the channels that it needs
    - Update the Gap Filling algorithm to only ingest the channels that it needs

- **ARR Risk 2:** The VPW processing system requires the IMS for the ancillary data snow mask
  - **Mitigation:**
    - Use the NCEP snow mask until the global IMS is available. The requirements are still met
    - Update VPW processing system when the global IMS is available

Review Items Summary

- **8** CDR risks have been addressed at ARR, **0** of which is still open.

- **2** risks and actions were reported from the ARR and they are both open

Agenda

- Introduction
- Risks
- Requirements
- Algorithm Readiness
- Software Architecture
- Delivered Algorithm Package
- Risks and Actions
- Summary and Conclusions
- Appendix – VIIRS Polar Winds Unit Tests

Summary and Conclusions

Presented by

Walter Wolf

Review Objectives Have Been Addressed

- Risks and Actions have been reviewed
- Requirements have been reviewed
- Algorithm Readiness has been reviewed
- Software Architecture has been reviewed
- Open Risks and Actions have been reviewed

Current Status

- Development on the VPW system has been completed
- The VPW documents have been completed and delivered
- The final VPW system has been delivered to NDE

Next Steps

- VIIRS Polar Winds ORR
  - » According to the NDE schedule
- SPSRB Presentation

Open Discussion

- The review is now open for free discussion

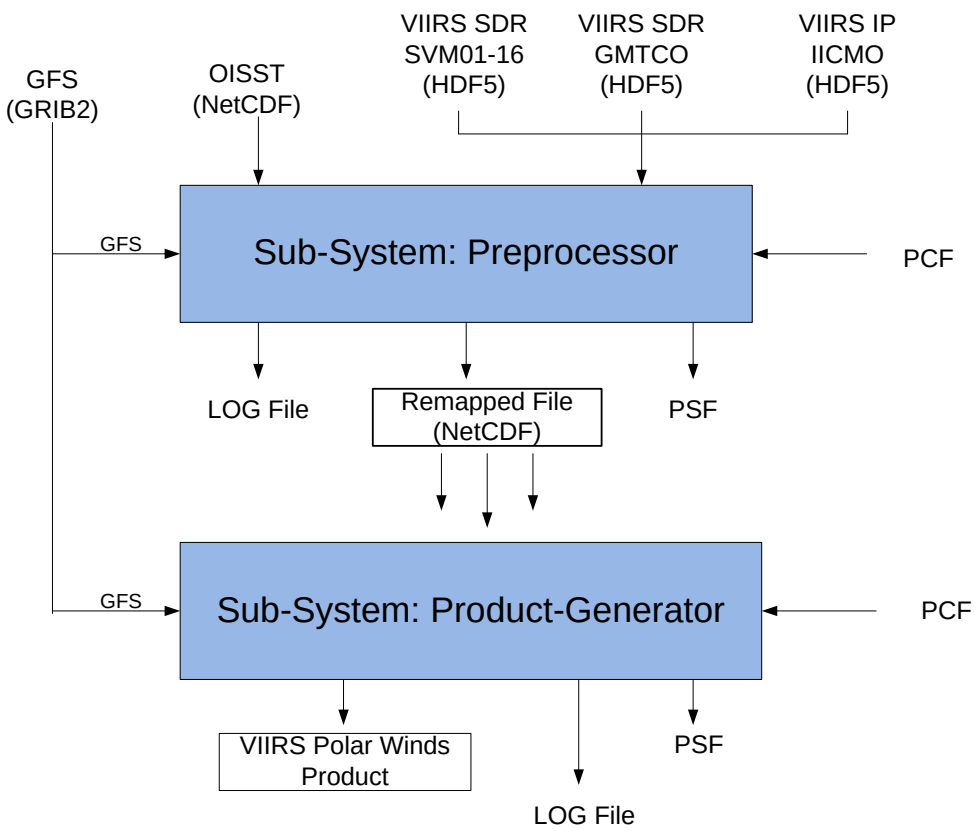
Appendix

VIIRS Polar Winds Unit Tests

VPW Unit Tests

- **VIIRS Polar Winds(VPW) Production System consists of 2 sub-systems:**
  - **Preprocessor (has 3 units)**
    - Unit1: SDR Data Preparation- Gap-filling SDR granule followed by conversion from HDF5 to NetCDF format
    - Unit 2: Cloud Product Generation - Generate granule cloud mask, cloud type and cloud height products
    - Unit 3: Polar Re-projection - Re-map satellite data and cloud products into one polar stereographic projection region (North/South)
  - **Product Generator (has 1 unit)**
    - Unit: VPW Product Generation Generate wind product from remapped files

VPW Unit Tests
System Level Diagram



VPW Unit Tests

- **VPWPS System Driver Scripts**
  - **Each sub-system has a system level driver script**
    - **System Driver script for Preprocessor:**

      VPW_REMAP.pl
    - **System Driver Script for Product Generator :**

      VPW_PRODUCT.pl

VPW Unit Tests

- **System Driver Script: VPW_REMAP.pl**
  - **Input File: System Processing Configuration File(PCF)**
    VPW_REMAP.pl.PCF
  - **Purpose:**
    - Parse system PCF file to set up running environment
    - Call NVPWPS_Preproc_SDR.pl to convert SDR data (Unit 1)
    - Call NVPWPS_Preproc_Clouds.pl to generate cloud products (Unit 2)
    - Call NVPWPS_Preproc_Remap.pl to create polar stereographic projection NetCDF file (Unit 3)
    - Generate remap log and PSF file

VPW Unit Tests

- **System Driver Script: VPW_PRODUCT.pl**
  - **Input File: System Processing Configuration File**
    VPW_PRODUCT.pl.PCF
  - **Purpose:**
    - Parse system PCF file to set running environment
    - Call NVPWPS_Preproc_PW.pl to generate polar wind product
    - Generate product log and PSF file

VPW Unit Tests

- **PCF and PSF files for scripts**
  - Each script has a PCF (Processing Configuration File) and PSF (Processing Status File) associated with it.
  - The PCF is the input for a script, it contains information needed to run the script.
  - The PSF shows the resultant status of the script running, it might contain output file location and name or error messages.

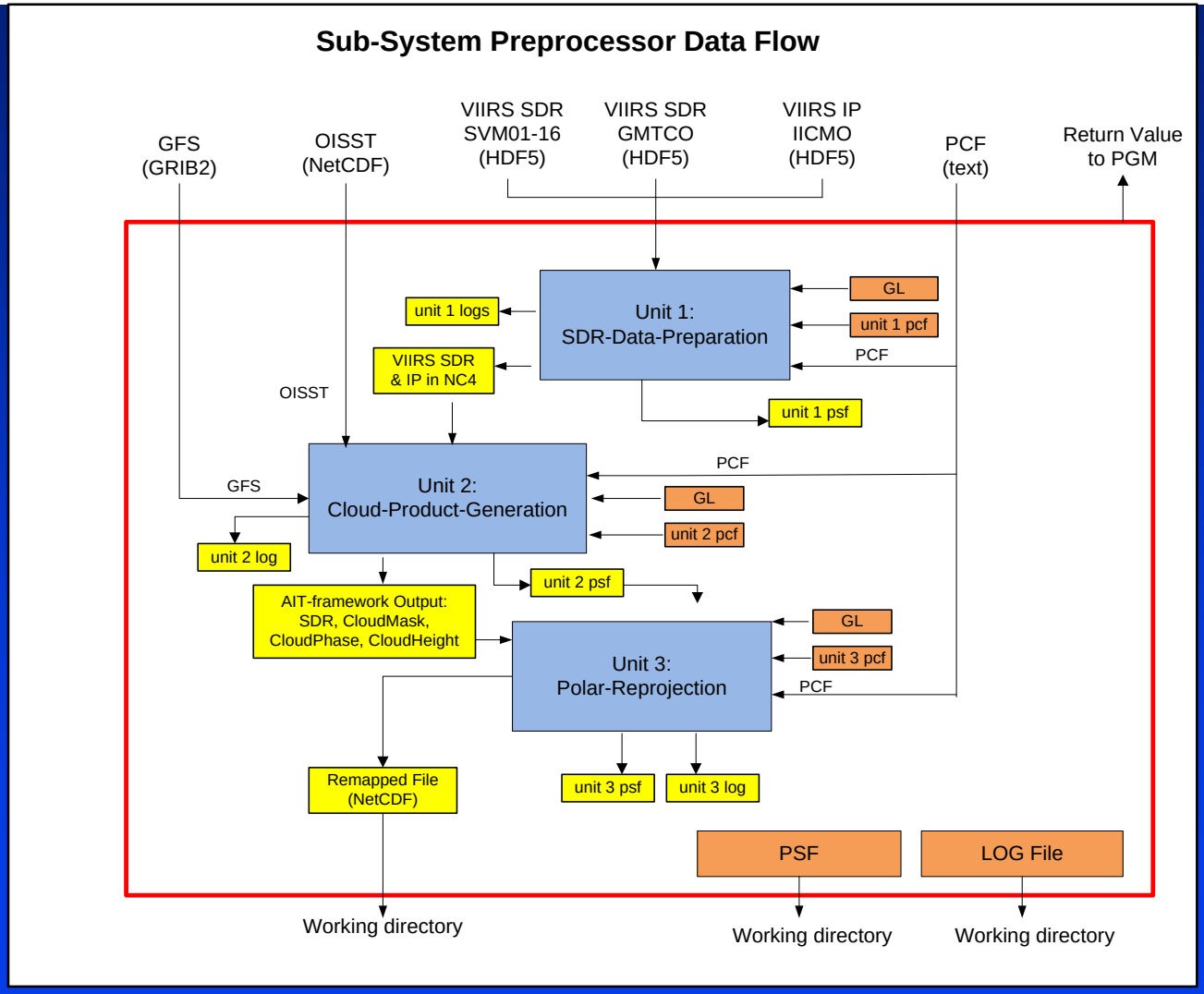
Script Name	PCF file	PSF file
VPW_REMAP.pl (driver script)	**VPW_REMAP.pl.PCF**	**VPW_REMAP.pl.PSF**
NVPWPS_Preproc_SDR.pl	NVPWPS_Preproc_SDR.pl.PCF	NVPWPS_Preproc_SDR.pl.PSF
NVPWPS_Preproc_Clouds.pl	NVPWPS_Preproc_Clouds.pl.PCF	NVPWPS_Preproc_Clouds.pl.PSF
NVPWPS_Preproc_Remap.pl	NVPWPS_Preproc_Remap.pl.PCF	NVPWPS_Preproc_Remap.pl.PSF
VPM_PRODUCT.pl (driver script)	**VPM_PRODUCT.pl.PCF**	**VPM_PRODUCT.pl.PSF**
NVPWPS_Proc_PW.pl	NVPWPS_Proc_PW.pl.PCF	NVPWPS_Proc_PW.pl.PSF

VPW Unit Tests

Sub-system:

Preprocessor

VPW Unit Tests
Sub-system: Preprocessor



VPW Unit Tests

System Driver Script:

VPW_REMAP.pl

Usage:

VPW_REMAP.pl *<System PCF>*

VPW Unit Tests
Preprocessor System PCF

- **System PCF:**

  VPW_REMAP.pl.PCF

  - **This file is created by NDE PGM at run time and it should exist in the 'working' directory**
  - **This file is parsed by the scripts to set up processing environments.**
  - **PCF Content (an example)**

```
#-------------------------------------------------
# SDR orbit start and end info
#-------------------------------------------------
job_coverage_start=201305020438171
job_coverage_end=201305020458110

#-----------------------------------------
# pole region
#-----------------------------------------
REGION=NORTH
```

VPW Unit Tests

- **Process Configuration File VPW_REMAP.pl.PCF**
  - » **PCF Content (cont.)**

```
#--------------------------------------
# system files and directories
# --------------------------------------

SCRIPT_FOR_GFS=run_wgrib.pl
PERL_LOC=/usr/bin/perl
OPS_SCRIPT=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/scripts
OPS_BIN=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/bin
MS2GT_DIR=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/bin/ms2gt0.24/bin

CFG_TEMPL_DIR=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/config_templates
PCF_TEMPL_DIR=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/PCF_Overwrites
PCF_OVERWRITES=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/PCF_Overwrites

CFG_TEMPL_FILE_FOR_CLOUDS=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/config_templates/AWG_NPP_VIIRS_Clouds.cfg
CFG_TEMPL_FILE_FOR_POLAR_WINDS=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/config_templates/AWG_NPP_VIIRS_Polar_Winds.cfg
PCF_TEMPL_FILE_FOR_VIIRS_SDR_Data=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/PCF_Overwrites/NPP_VIIRS_SDR_Data.pcf
PCF_TEMPL_FILE_FOR_VIIRS_Cloud_Mask=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/PCF_Overwrites/NPP_VIIRS_Cloud_Mask.pcf
```

VPW Unit Tests

- **Process Configuration File VPW_REMAP.pl.PCF**
  - » **PCF Content (cont.)**

```
#------------------------------------------------------------------------------
#static ancillary data: latitude, longitude, CRTM Coef land mask etc
#------------------------------------------------------------------------------

REMAP_GPD_FILE_SP=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/remap/mySpseh.gpd
REMAP_GPD_FILE_NP=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/remap/myNpseh.gpd
REMAP_MPP_FILE_SP=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/remap/mySpse.mpp
REMAP_MPP_FILE_NP=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/remap/myNpse.mpp
CRTM_VIIRS_M_SpcCoeff=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/viirs-m_npp.SpcCoeff.bin
CRTM_VIIRS_M_TauCoeff=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/viirs-m_npp.TauCoeff.bin
CRTM_CloudCoeff=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/CloudCoeff.bin
CRTM_AerosolCoeff=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/AerosolCoeff.bin
CRTM_EmisCoeff=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/EmisCoeff.bin
AWG_NOP_AMV_COEF=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/AWG_WINDS_AMV/AMV_coeff.nc
NPP_VIIRS_SDR_ANCIL=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/npp_viirs_ancil.nc
POLAR_MAP_ANCIL=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/aqua_modis_abi_pol_ancil.nc
SPS_LATLON_ANCIL=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/mySpseh_Lats.Lons.3305x3305x1.nc
NPS_LATLON_ANCIL=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/myNpseh_Lats.Lons.3305x3305x1.nc
NPP_VIIRS_CLX_CLDHGT_ANCIL=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/pol_winds/polar_winds_ancil.nc
COAST_MASK_NASA_1KM=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/masks/coast_mask_1km.nc
SFC_TYPE_AVHRR_1KM=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/masks/gl-latlong-1km-landcover.nc
LAND_MASK_NASA_1KM=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/masks/lw_geo_2001001_v03m.nc
SFC_ELEV_GLOBE_1KM=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/sfc_elevation/GLOBE_1km_digelev.nc
```

VPW Unit Tests

- **Process Configuration File VPW_REMAP.pl.PCF**
  - » **PCF Content (cont.)**

```
#----------------------------------------------------
# static ancillary data: SEEBOR emissivity
#----------------------------------------------------

F1_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005001.nc
F2_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005032.nc
F3_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005060.nc
F4_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005091.nc
F5_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005121.nc
F6_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005152.nc
F7_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005182.nc
F8_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005213.nc
F9_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005244.nc
F10_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005274.nc
F11_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005305.nc
F12_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005335.nc
```

VPW Unit Tests

- **Process Configuration File VPW_REMAP.pl.PCF**
  - » **PCF Content (cont.)**

```
#------------------------------------------------------------------
# dirs need to be created under the WORKING_DIR
# ------------------------------------------------------------------
OPS_LIST_DIR=GRANULE_LIST
HDF_DATA=OUT/viirs_hdf5
SDR_DATA=OUT/npp_viirs_sdr
VIIRS_REMAP_DATA_DIR=OUT/npp_viirs_remap
REMAP_WORKING=remap_working

ALGORITHM_ANCILLARY_DIR=algorithm_ancillary
FRAMEWORK_ANCILLARY_DIR=framework_ancillary

CLOUD_DATA=OUT/viirs_results_clouds
VPW_OUTPUT=OUT/viirs_results_pw
```

VPW Unit Tests

- **Process Configuration File VPW_REMAP.pl.PCF**
  - » **PCF Content (cont.)**

```
#-------------------------------------------------------
# dynamic ancillary data: GFS and OISST
#-------------------------------------------------------

gfs_file=gfs_data/gfs.t00z.pgrb2f03.20130502
gfs_file=gfs_data/gfs.t00z.pgrb2f06.20130502

oisst_file=oisst_data/avhrr-only-v2.20130501_preliminary.nc
```

VPW Unit Tests

- **Process Configuration File VPW_REMAP.pl.PCF**
  - » **PCF Content (cont.)**

```
#--------------------------------------------------
# SDR data
#--------------------------------------------------

gmtco=sdr_data/GMTCO_npp_d20130502_t0438171_e0439412_b07829_c20130502105911542952_noaa_ops.h5
gmtco=sdr_data/GMTCO_npp_d20130502_t0439425_e0441066_b07829_c20130502105911542952_noaa_ops.h5
……
gmtco=sdr_data/GMTCO_npp_d20130502_t0456455_e0458097_b07829_c20130502111932325389_noaa_ops.h5
gmtco=sdr_data/GMTCO_npp_d20130502_t0458110_e0459351_b07829_c20130502111932325389_noaa_ops.h5

iicmo=sdr_data/IICMO_npp_d20130502_t0438171_e0439412_b07829_c20130502110556311843_noaa_ops.h5
iicmo=sdr_data/IICMO_npp_d20130502_t0439425_e0441066_b07829_c20130502110556311843_noaa_ops.h5
……
iicmo=sdr_data/IICMO_npp_d20130502_t0456455_e0458097_b07829_c20130502112343455424_noaa_ops.h5
iicmo=sdr_data/IICMO_npp_d20130502_t0458110_e0459351_b07829_c20130502112343455424_noaa_ops.h5

svm01=sdr_data/SVM01_npp_d20130502_t0438171_e0439412_b07829_c20130502110553465700_noaa_ops.h5
svm01=sdr_data/SVM01_npp_d20130502_t0439425_e0441066_b07829_c20130502110553465700_noaa_ops.h5
……
svm01=sdr_data/SVM01_npp_d20130502_t0456455_e0458097_b07829_c20130502112335426578_noaa_ops.h5
svm01=sdr_data/SVM01_npp_d20130502_t0458110_e0459351_b07829_c20130502112335426578_noaa_ops.h5
```

VPW Unit Tests

- **Process Configuration File VPW_REMAP.pl.PCF**
  - » **PCF Content (cont.)**

```
svm02=sdr_data/SVM02_npp_d20130502_t0438171_e0439412_b07829_c20130502110545757179_noaa_ops.h5
svm02=sdr_data/SVM02_npp_d20130502_t0439425_e0441066_b07829_c20130502110545757179_noaa_ops.h5
……
svm02=sdr_data/SVM02_npp_d20130502_t0456455_e0458097_b07829_c20130502112336394790_noaa_ops.h5
svm02=sdr_data/SVM02_npp_d20130502_t0458110_e0459351_b07829_c20130502112336394790_noaa_ops.h5

svm03=sdr_data/SVM03_npp_d20130502_t0438171_e0439412_b07829_c20130502110533233128_noaa_ops.h5
svm03=sdr_data/SVM03_npp_d20130502_t0439425_e0441066_b07829_c20130502110533233128_noaa_ops.h5
……
svm03=sdr_data/SVM03_npp_d20130502_t0456455_e0458097_b07829_c20130502112333341451_noaa_ops.h5
svm03=sdr_data/SVM03_npp_d20130502_t0458110_e0459351_b07829_c20130502112333341451_noaa_ops.h5

svm04=sdr_data/SVM04_npp_d20130502_t0438171_e0439412_b07829_c20130502110534156258_noaa_ops.h5
svm04=sdr_data/SVM04_npp_d20130502_t0439425_e0441066_b07829_c20130502110534156258_noaa_ops.h5
……
svm04=sdr_data/SVM04_npp_d20130502_t0456455_e0458097_b07829_c20130502112334444932_noaa_ops.h5
svm04=sdr_data/SVM04_npp_d20130502_t0458110_e0459351_b07829_c20130502112334444932_noaa_ops.h5

svm05=sdr_data/SVM05_npp_d20130502_t0438171_e0439412_b07829_c20130502110536033128_noaa_ops.h5
svm05=sdr_data/SVM05_npp_d20130502_t0439425_e0441066_b07829_c20130502110536033128_noaa_ops.h5
……
svm05=sdr_data/SVM05_npp_d20130502_t0456455_e0458097_b07829_c20130502112340278888_noaa_ops.h5
svm05=sdr_data/SVM05_npp_d20130502_t0458110_e0459351_b07829_c20130502112340278888_noaa_ops.h5
```

VPW Unit Tests

- **Process Configuration File VPW_REMAP.pl.PCF**
  - » **PCF Content (cont.)**

```
svm06=sdr_data/SVM06_npp_d20130502_t0438171_e0439412_b07829_c20130502111654623348_noaa_ops.h5
svm06=sdr_data/SVM06_npp_d20130502_t0439425_e0441066_b07829_c20130502111654623348_noaa_ops.h5
……
svm06=sdr_data/SVM06_npp_d20130502_t0456455_e0458097_b07829_c20130502112338182835_noaa_ops.h5
svm06=sdr_data/SVM06_npp_d20130502_t0458110_e0459351_b07829_c20130502112338182835_noaa_ops.h5

svm07=sdr_data/SVM07_npp_d20130502_t0438171_e0439412_b07829_c20130502111658089165_noaa_ops.h5
svm07=sdr_data/SVM07_npp_d20130502_t0439425_e0441066_b07829_c20130502111658089165_noaa_ops.h5
……
svm07=sdr_data/SVM07_npp_d20130502_t0456455_e0458097_b07829_c20130502112341471535_noaa_ops.h5
svm07=sdr_data/SVM07_npp_d20130502_t0458110_e0459351_b07829_c20130502112341471535_noaa_ops.h5

svm08=sdr_data/SVM08_npp_d20130502_t0438171_e0439412_b07829_c20130502110538017419_noaa_ops.h5
svm08=sdr_data/SVM08_npp_d20130502_t0439425_e0441066_b07829_c20130502110538017419_noaa_ops.h5
……
svm08=sdr_data/SVM08_npp_d20130502_t0455201_e0456443_b07829_c20130502112347079956_noaa_ops.h5
svm08=sdr_data/SVM08_npp_d20130502_t0456455_e0458097_b07829_c20130502112347079956_noaa_ops.h5

svm09=sdr_data/SVM09_npp_d20130502_t0438171_e0439412_b07829_c20130502110538961429_noaa_ops.h5
svm09=sdr_data/SVM09_npp_d20130502_t0439425_e0441066_b07829_c20130502110538961429_noaa_ops.h5
……
svm09=sdr_data/SVM09_npp_d20130502_t0456455_e0458097_b07829_c20130502112346684448_noaa_ops.h5
svm09=sdr_data/SVM09_npp_d20130502_t0458110_e0459351_b07829_c20130502112346684448_noaa_ops.h5
```

VPW Unit Tests

- **Process Configuration File VPW_REMAP.pl.PCF**
  - » **PCF Content (cont.)**

```
svm10=sdr_data/SVM10_npp_d20130502_t0438171_e0439412_b07829_c20130502110545700492_noaa_ops.h5
svm10=sdr_data/SVM10_npp_d20130502_t0439425_e0441066_b07829_c20130502110545700492_noaa_ops.h5
……
svm10=sdr_data/SVM10_npp_d20130502_t0456455_e0458097_b07829_c20130502112346429948_noaa_ops.h5
svm10=sdr_data/SVM10_npp_d20130502_t0458110_e0459351_b07829_c20130502112346429948_noaa_ops.h5

svm11=sdr_data/SVM11_npp_d20130502_t0438171_e0439412_b07829_c20130502110539928599_noaa_ops.h5
svm11=sdr_data/SVM11_npp_d20130502_t0439425_e0441066_b07829_c20130502110539928599_noaa_ops.h5
……
svm11=sdr_data/SVM11_npp_d20130502_t0456455_e0458097_b07829_c20130502112353806025_noaa_ops.h5
svm11=sdr_data/SVM11_npp_d20130502_t0458110_e0459351_b07829_c20130502112353806025_noaa_ops.h5

svm12=sdr_data/SVM12_npp_d20130502_t0438171_e0439412_b07829_c20130502110546811697_noaa_ops.h5
svm12=sdr_data/SVM12_npp_d20130502_t0439425_e0441066_b07829_c20130502110546811697_noaa_ops.h5
……
svm12=sdr_data/SVM12_npp_d20130502_t0456455_e0458097_b07829_c20130502112350647322_noaa_ops.h5
svm12=sdr_data/SVM12_npp_d20130502_t0458110_e0459351_b07829_c20130502112350647322_noaa_ops.h5

svm13=sdr_data/SVM13_npp_d20130502_t0438171_e0439412_b07829_c20130502110614330280_noaa_ops.h5
svm13=sdr_data/SVM13_npp_d20130502_t0439425_e0441066_b07829_c20130502110614330280_noaa_ops.h5
……
svm13=sdr_data/SVM13_npp_d20130502_t0456455_e0458097_b07829_c20130502112350654352_noaa_ops.h5
svm13=sdr_data/SVM13_npp_d20130502_t0458110_e0459351_b07829_c20130502112350654352_noaa_ops.h5
```

VPW Unit Tests

- **System Process Configuration File VPW_REMAP.pl.PCF**
  - » **PCF Content (cont.)**

```
svm14=sdr_data/SVM14_npp_d20130502_t0438171_e0439412_b07829_c20130502110547569639_noaa_ops.h5
svm14=sdr_data/SVM14_npp_d20130502_t0439425_e0441066_b07829_c20130502110547569639_noaa_ops.h5
……
svm14=sdr_data/SVM14_npp_d20130502_t0456455_e0458097_b07829_c20130502112345152529_noaa_ops.h5
svm14=sdr_data/SVM14_npp_d20130502_t0458110_e0459351_b07829_c20130502112345152529_noaa_ops.h5

svm15=sdr_data/SVM15_npp_d20130502_t0438171_e0439412_b07829_c20130502110607615353_noaa_ops.h5
svm15=sdr_data/SVM15_npp_d20130502_t0439425_e0441066_b07829_c20130502110607615353_noaa_ops.h5
……
svm15=sdr_data/SVM15_npp_d20130502_t0456455_e0458097_b07829_c20130502112351862087_noaa_ops.h5
svm15=sdr_data/SVM15_npp_d20130502_t0458110_e0459351_b07829_c20130502112351862087_noaa_ops.h5

svm16=sdr_data/SVM16_npp_d20130502_t0438171_e0439412_b07829_c20130502110545216163_noaa_ops.h5
svm16=sdr_data/SVM16_npp_d20130502_t0439425_e0441066_b07829_c20130502110545216163_noaa_ops.h5
……
svm16=sdr_data/SVM16_npp_d20130502_t0456455_e0458097_b07829_c20130502112349038848_noaa_ops.h5
svm16=sdr_data/SVM16_npp_d20130502_t0458110_e0459351_b07829_c20130502112349038848_noaa_ops.h5
```

VPW Unit Tests
Running Environment

- Setup Running Environment
  - » Purpose: Extract variables from system PCF file VPW_REMAP.pl.PCF and setup environment for polar winds
  - » Major Steps
    - Parse system PCF file
    - Create necessary directories if they do not exist
    - Make links to framework
    - Make links to static ancillary data
      - Call Setup_links_for_static_ancillary.pl
    - Make links to dynamic and ancillary data
    - Create granule list file and make links
    - Create log file VPW_REMAP.pl.log

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Create necessary directories if they do not exist**
    - **working directory**

```
drwxr-sr-x 12 nobody nobody 4096 Mar 12 16:58 working
```

    - **GRANULE_LIST**

```
drwxr-sr-x 2 nobody nobody       25 Mar 11 11:48 GRANULE_LIST
```

    - **Directory working/OUT**

```
drwxr-sr-x 7 nobody nobody     4096 Mar 11 10:24 OUT
```

    - **Directories under OUT**

```
drwxr-sr-x 2 nobody nobody    55 Mar 11 10:42 npp_viirs_remap
drwxr-sr-x 2 nobody nobody 20480 Mar 11 10:27 npp_viirs_sdr
drwxr-sr-x 2 nobody nobody 53248 Mar 11 10:27 viirs_hdf5
drwxr-sr-x 5 nobody nobody    45 Mar 11 10:27 viirs_results_clouds
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - **Make links to framework**
    - **Location**

```
/data/data046/working
```

    - **Links**

```
lrwxrwxrwx 1 nobody nobody 69 Jan 17 12:33 CustomProductLists -> /data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/scripts/CustomProductLists
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 Default_PCF -> /data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/scripts/Default_PCF
lrwxrwxrwx 1 nobody nobody 57 Jan 17 12:33 PCF_Overwrites -> /data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/PCF_Overwrites
lrwxrwxrwx 1 nobody nobody 50 Jan  8 10:55 run_wgrib.pl -> /data/data046/mfan/NVPWPS/OPS/scripts/run_wgrib.pl
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Make links to static ancillary data**
    - **CRTM Coefficient**
      - **Location**

/data/data046/working/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 126 Jan 17 12:33 AerosolCoeff.bin ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/AerosolCoeff.bin
lrwxrwxrwx 1 nobody nobody 124 Jan 17 12:33 CloudCoeff.bin ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/CloudCoeff.bin
lrwxrwxrwx 1 nobody nobody 123 Jan 17 12:33 EmisCoeff.bin ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/EmisCoeff.bin
lrwxrwxrwx 1 nobody nobody 134 Jan 17 12:33 viirs-m_npp.SpcCoeff.bin ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/viirs-m_npp.SpcCoeff.bin
lrwxrwxrwx 1 nobody nobody 134 Jan 17 12:33 viirs-m_npp.TauCoeff.bin ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/viirs-m_npp.TauCoeff.bin
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Make links to static ancillary data**
    - **Wind Coefficient**
      - **Location**

```
/data/data046/working/algorithm_ancillary/AWG_WINS_AMV
```

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 96 Jan 17 12:33 AMV_coeff.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/AWG_WINDS_AMV/AMV_coeff.nc
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Make links to static ancillary data**
    - **Framework ancillary files**
      - **Location**

```
/data/data046/working/framework_ancillary/ancil_file
```

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 myNpseh_Lats.Lons.3305x3305x1.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/myNpseh_Lats.Lons.3305x3305x1.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 mySpseh_Lats.Lons.3305x3305x1.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/mySpseh_Lats.Lons.3305x3305x1.nc
lrwxrwxrwx 1 nobody nobody  99 Jan 17 12:33 npp_viirs_ancil.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/npp_viirs_ancil.nc
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - **Make links to static ancillary data**
    - **Polar winds**
      - **Location**

```
/data/data046/working/framework_ancillary/pol_winds
```

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 100 Jan 17 12:33 polar_winds_ancil.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/pol_winds/polar_winds_ancil.nc
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Make links to static ancillary data**
    - **Mask files**
      - **Location**

```
/data/data046/working/framework_ancillary/masks
```

      - **Links**

```
lrwxrwxrwx 1 nobody nobody  93 Jan 17 12:33 coast_mask_1km.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/masks/coast_mask_1km.nc
lrwxrwxrwx 1 nobody nobody 103 Jan 17 12:33 gl-latlong-1km-landcover.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/masks/gl-latlong-1km-landcover.nc
lrwxrwxrwx 1 nobody nobody  98 Jan 17 12:33 lw_geo_2001001_v03m.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/masks/lw_geo_2001001_v03m.nc
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - **Make links to static ancillary data**
    - **Surface elevation**
      - **Location**

```
/data/data046/working/framework_ancillary/sfc_elevation
```

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 104 Jan 17 12:33 GLOBE_1km_digelev.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/sfc_elevation/GLOBE_1km_digelev.nc
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - **Make links to static ancillary data**
    - **Seebor emissivity**
      - **Location**

/data/data046/working/framework_ancillary/seebor_emiss

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005001.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005001.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005032.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005032.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005060.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005060.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005091.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005091.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005121.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005121.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005152.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005152.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005182.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005182.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005213.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005213.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005244.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005244.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005274.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005274.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005305.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005305.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005335.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005335.nc
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Make links to static ancillary data**
    - **Log file location and name**

/data/data046/working/Setup_links_for_static_ancillary.pl.log

- **Log file content**

```
ln: creating symbolic link `algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/./viirs-m_npp.SpcCoeff.bin': File exists
ln: creating symbolic link `algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/./viirs-m_npp.TauCoeff.bin': File exists
ln: creating symbolic link `algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/./CloudCoeff.bin': File exists
ln: creating symbolic link `algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/./AerosolCoeff.bin': File exists
ln: creating symbolic link `algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/./EmisCoeff.bin': File exists
ln: creating symbolic link `algorithm_ancillary/AWG_WINDS_AMV/./AMV_coeff.nc': File exists
ln: creating symbolic link `framework_ancillary/ancil_file/./npp_viirs_ancil.nc': File exists
ln: creating symbolic link `framework_ancillary/ancil_file/./aqua_modis_abi_pol_ancil.nc': File exists
ln: creating symbolic link `framework_ancillary/ancil_file/./mySpseh_Lats.Lons.3305x3305x1.nc': File exists
ln: creating symbolic link `framework_ancillary/ancil_file/./myNpseh_Lats.Lons.3305x3305x1.nc': File exists
ln: creating symbolic link `framework_ancillary/pol_winds/./polar_winds_ancil.nc': File exists
ln: creating symbolic link `framework_ancillary/masks/./coast_mask_1km.nc': File exists
ln: creating symbolic link `framework_ancillary/masks/./gl-latlong-1km-landcover.nc': File exists
ln: creating symbolic link `framework_ancillary/masks/./lw_geo_2001001_v03m.nc': File exists
ln: creating symbolic link `framework_ancillary/sfc_elevation/./GLOBE_1km_digelev.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005001.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005032.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005060.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005091.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005121.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005152.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005182.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005213.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005244.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005274.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005305.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005335.nc': File exists
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Make links to dynamic ancillary data**
    - **GFS GRIB2**
      - **Location**

/data/data046/working/framework_ancillary/gfs_grib2_0.5deg/2013/05/02

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t00z.pgrbf03 -> /data/data046/mfan/working/gfs_data/gfs.t00z.pgrb2f03.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t00z.pgrbf06 -> /data/data046/mfan/working/gfs_data/gfs.t00z.pgrb2f06.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t00z.pgrbf09 -> /data/data046/mfan/working/gfs_data/gfs.t00z.pgrb2f09.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t06z.pgrbf03 -> /data/data046/mfan/working/gfs_data/gfs.t06z.pgrb2f03.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t06z.pgrbf06 -> /data/data046/mfan/working/gfs_data/gfs.t06z.pgrb2f06.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t06z.pgrbf09 -> /data/data046/mfan/working/gfs_data/gfs.t06z.pgrb2f09.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t12z.pgrbf03 -> /data/data046/mfan/working/gfs_data/gfs.t12z.pgrb2f03.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t12z.pgrbf06 -> /data/data046/mfan/working/gfs_data/gfs.t12z.pgrb2f06.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t12z.pgrbf09 -> /data/data046/mfan/working/gfs_data/gfs.t12z.pgrb2f09.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t18z.pgrbf03 -> /data/data046/mfan/working/gfs_data/gfs.t18z.pgrb2f03.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t18z.pgrbf06 -> /data/data046/mfan/working/gfs_data/gfs.t18z.pgrb2f06.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t18z.pgrbf09 -> /data/data046/mfan/working/gfs_data/gfs.t18z.pgrb2f09.20130502
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - **Make links to dynamic ancillary data**
    - **OISST**
      - **Location**

```
/data/data046/working/framework_ancillary/oisst_daily
```

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 75 Jan 17 12:33 avhrr-only-v2.20130501.nc -> /data/data046/mfan/working/oisst_data/avhrr-only-v2.20130501_preliminary.nc
lrwxrwxrwx 1 nobody nobody 75 Jan 17 12:33 avhrr-only-v2.20130502.nc -> /data/data046/mfan/working/oisst_data/avhrr-only-v2.20130502_preliminary.nc
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Create granule list file and make links**
    - **SDR granule list file**
      - **SDR list file name**

```
/data/data046/working/GRANULE_LIST/granule_list
```

      - **SDR granule list file name content**

```
2013 05 02 d20130502_t0438171_e0439412_b07829 U U
2013 05 02 d20130502_t0439425_e0441066_b07829 U U
2013 05 02 d20130502_t0441079_e0442320_b07829 U U
2013 05 02 d20130502_t0442333_e0443575_b07829 U U
2013 05 02 d20130502_t0443587_e0445229_b07829 U U
2013 05 02 d20130502_t0445241_e0446483_b07829 U U
2013 05 02 d20130502_t0446495_e0448137_b07829 U U
2013 05 02 d20130502_t0448149_e0449391_b07829 U U
2013 05 02 d20130502_t0449403_e0451027_b07829 U U
2013 05 02 d20130502_t0451039_e0452281_b07829 U U
2013 05 02 d20130502_t0452293_e0453535_b07829 U U
2013 05 02 d20130502_t0453547_e0455189_b07829 U U
2013 05 02 d20130502_t0455201_e0456443_b07829 U U
2013 05 02 d20130502_t0456455_e0458097_b07829 U U
2013 05 02 d20130502_t0458110_e0459351_b07829 U U
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Create granule list file and make links**
    - **Links location**

/data/data046/working/OUT/viirs_hdf5

    - **Links**

lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 GMTCO_npp_d20130502_t0438171_e0439412_b07829_c20130502105911542952_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/GMTCO_npp_d20130502_t0438171_e0439412_b07829_c20130502105911542952_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 IICMO_npp_d20130502_t0438171_e0439412_b07829_c20130502110556311843_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/IICMO_npp_d20130502_t0438171_e0439412_b07829_c20130502110556311843_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM01_npp_d20130502_t0438171_e0439412_b07829_c20130502110553465700_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM01_npp_d20130502_t0438171_e0439412_b07829_c20130502110553465700_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM02_npp_d20130502_t0438171_e0439412_b07829_c20130502110545757179_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM02_npp_d20130502_t0438171_e0439412_b07829_c20130502110545757179_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM03_npp_d20130502_t0438171_e0439412_b07829_c20130502110533233128_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM03_npp_d20130502_t0438171_e0439412_b07829_c20130502110533233128_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM04_npp_d20130502_t0438171_e0439412_b07829_c20130502110534156258_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM04_npp_d20130502_t0438171_e0439412_b07829_c20130502110534156258_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM05_npp_d20130502_t0438171_e0439412_b07829_c20130502110536033128_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM05_npp_d20130502_t0438171_e0439412_b07829_c20130502110536033128_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM06_npp_d20130502_t0438171_e0439412_b07829_c20130502111654623348_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM06_npp_d20130502_t0438171_e0439412_b07829_c20130502111654623348_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM07_npp_d20130502_t0438171_e0439412_b07829_c20130502111658089165_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM07_npp_d20130502_t0438171_e0439412_b07829_c20130502111658089165_noaa_ops.h5

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Create granule list file and make links**
    - **Links (cont.)**

```
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM08_npp_d20130502_t0438171_e0439412_b07829_c20130502110538017419_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM08_npp_d20130502_t0438171_e0439412_b07829_c20130502110538017419_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM09_npp_d20130502_t0438171_e0439412_b07829_c20130502110538961429_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM09_npp_d20130502_t0438171_e0439412_b07829_c20130502110538961429_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM10_npp_d20130502_t0438171_e0439412_b07829_c20130502110545700492_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM10_npp_d20130502_t0438171_e0439412_b07829_c20130502110545700492_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM11_npp_d20130502_t0438171_e0439412_b07829_c20130502110539928599_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM11_npp_d20130502_t0438171_e0439412_b07829_c20130502110539928599_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM12_npp_d20130502_t0438171_e0439412_b07829_c20130502110546811697_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM12_npp_d20130502_t0438171_e0439412_b07829_c20130502110546811697_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM13_npp_d20130502_t0438171_e0439412_b07829_c20130502110614330280_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM13_npp_d20130502_t0438171_e0439412_b07829_c20130502110614330280_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM14_npp_d20130502_t0438171_e0439412_b07829_c20130502110547569639_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM14_npp_d20130502_t0438171_e0439412_b07829_c20130502110547569639_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM15_npp_d20130502_t0438171_e0439412_b07829_c20130502110607615353_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM15_npp_d20130502_t0438171_e0439412_b07829_c20130502110607615353_noaa_ops.h5
lrwxrwxrwx 1 nobody nobody 114 Mar 11 10:24 SVM16_npp_d20130502_t0438171_e0439412_b07829_c20130502110545216163_noaa_ops.h5 ->
/data/data046/mfan/working/sdr_data/SVM16_npp_d20130502_t0438171_e0439412_b07829_c20130502110545216163_noaa_ops.h5

.....
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Create log file**
    - **Log file location and name**

/data/data046/mfan/working /VPW_REMAP.pl.log

    - **Log file content**

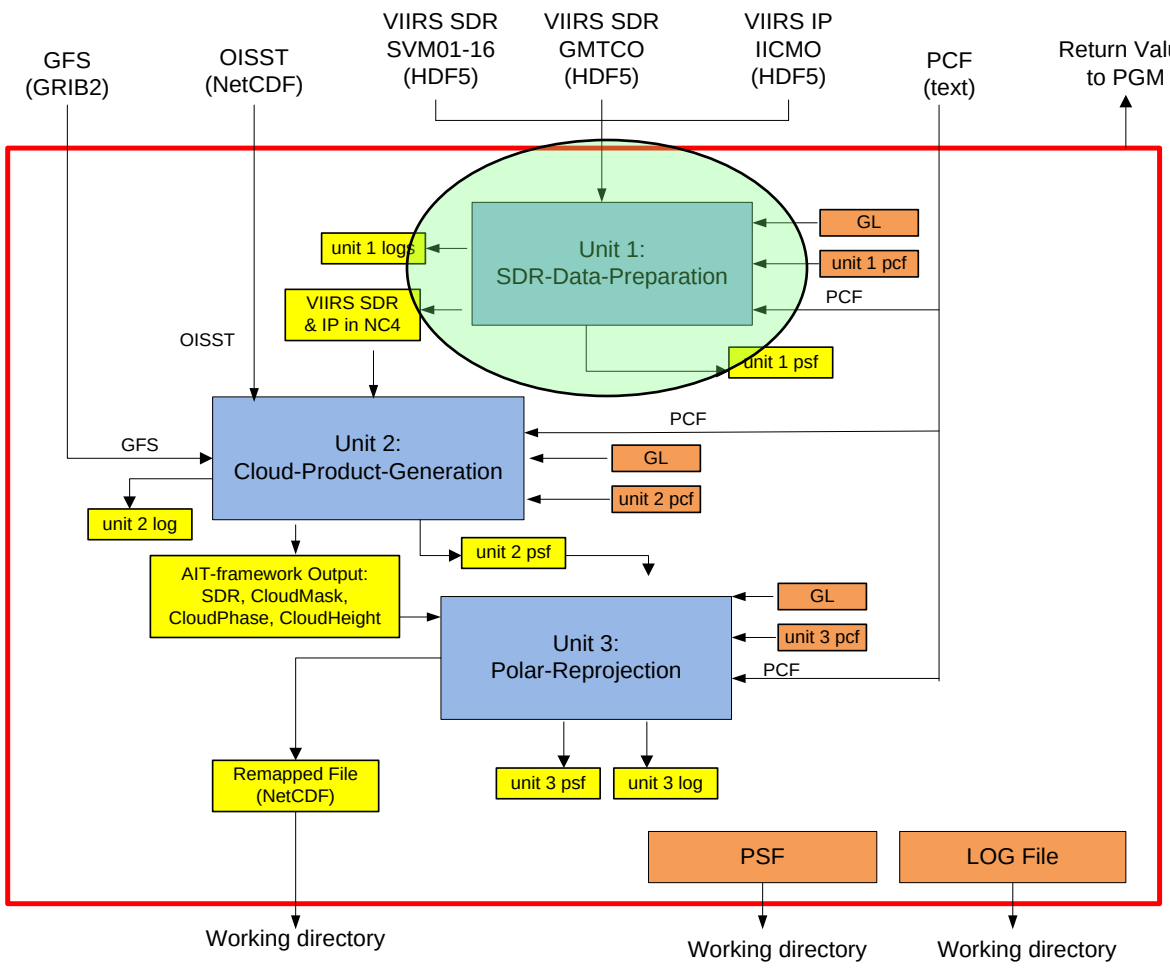
VPW_REMAP.pl is now starting at 20140311-14:24:19

granule_time = d20130502_t0438171_e0439412_b07829
granule_time = d20130502_t0439425_e0441066_b07829
granule_time = d20130502_t0441079_e0442320_b07829
granule_time = d20130502_t0442333_e0443575_b07829
granule_time = d20130502_t0443587_e0445229_b07829
granule_time = d20130502_t0445241_e0446483_b07829
granule_time = d20130502_t0446495_e0448137_b07829
granule_time = d20130502_t0448149_e0449391_b07829
granule_time = d20130502_t0449403_e0451027_b07829
granule_time = d20130502_t0451039_e0452281_b07829
granule_time = d20130502_t0452293_e0453535_b07829
granule_time = d20130502_t0453547_e0455189_b07829
granule_time = d20130502_t0455201_e0456443_b07829
granule_time = d20130502_t0456455_e0458097_b07829
granule_time = d20130502_t0458110_e0459351_b07829

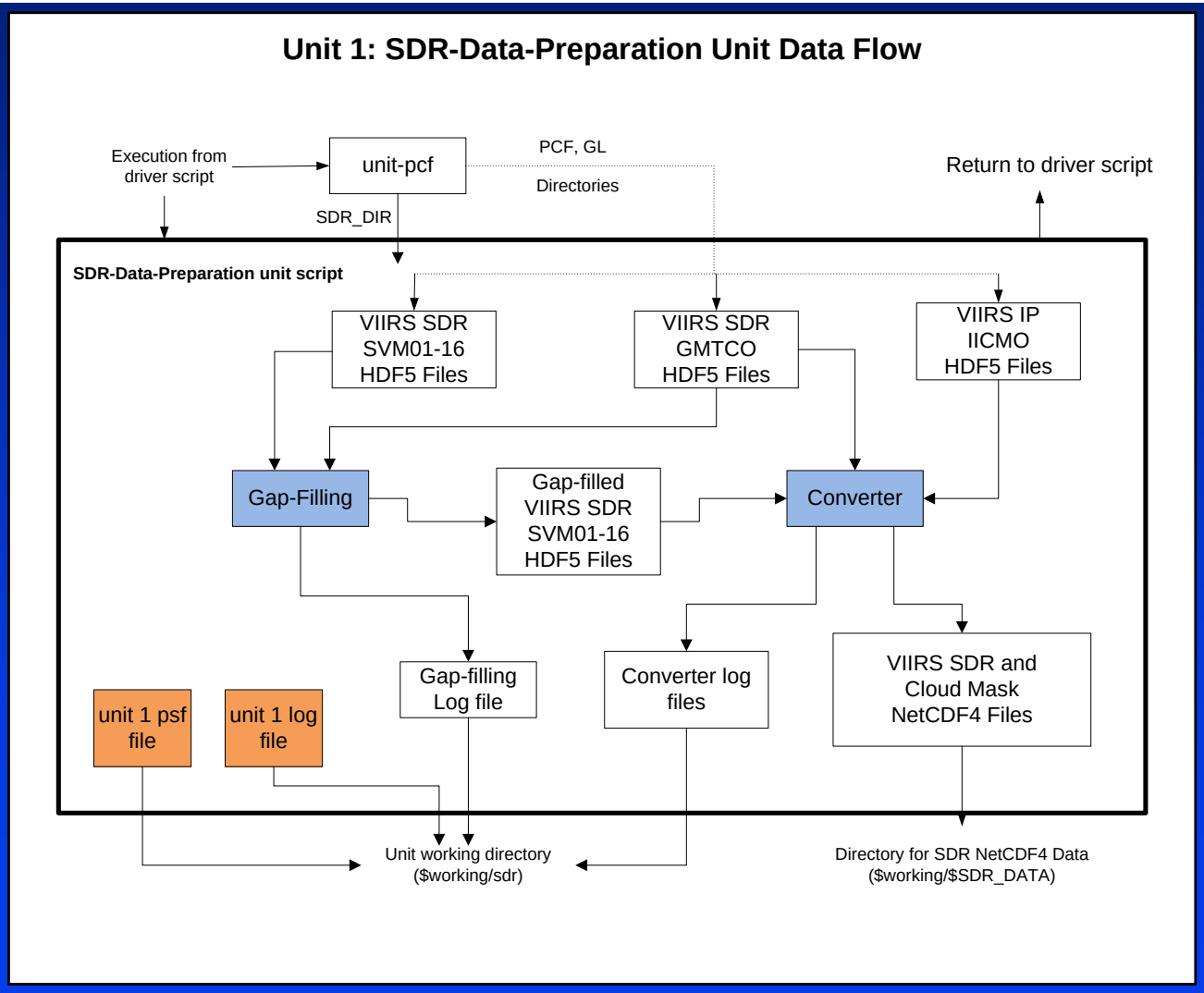
first_granule = d20130502_t0438171_e0439412_b07829
last_granule = d20130502_t0458110_e0459351_b07829

LIST_FILE = /data/data046/mfan/working/GRANULE_LIST/granule_list
sdr_hdf_dir = /data/data046/mfan/working/OUT/viirs_hdf5

Preprocessor
Unit 1: SDR Data Preparation



Preprocessor
Unit 1: SDR Data Preparation



VPW Unit Tests
SDR Data Preparation (unit 1)

- **SDR Data Preparation**
  - **Purpose: Perform gap filling on HDF5 SDR data and Convert the gap filled SDR data into NetCDF format**
  - **Major steps**
    - **Create directory $working/sdr and enter into it**
    - **Generate SDR PCF file NVPWPS_Preproc_SDR.pl.PCF**

VPW Unit Tests
SDR Data Preparation (unit 1)

- SDR Data Preparation
  - » Major steps (cont.)
    - Perform gap filling and Convert SDR data
      - ➢ Call NVPWPS_Preproc_SDR.pl
        - o Perform gap filling on SDR HDF5 data
        - ✓ Call mender.py
        - ❖ Call viirsmend.py
        - o Convert SDR data to NetCDF
        - ✓ Call convert_viirs_hdf5_2_nc4.pl
        - ❖ Execute get_sdr_scan_time.exe
        - ❖ Call H5augjpss
        - o Update status file
        - ✓ Call update_pre_process_status.pl
        - o Create a log file for NVPWPS_Preproc_SDR.pl
        - o Create conversion PSF file
    - Update log file VPW_REMAP.pl.log

VPW Unit Tests
SDR Data Preparation (unit 1)

- **Validation: SDR Data Preparation**
  - **Create directory $working/sdr**

```
/data/data046/working/sdr
```

  - **Generate SDR PCF file**
    - **SDR PCF location and name**

```
/data/data046/working/sdr/NVPWPS_Preproc_SDR.pl.PCF
```

    - **SDR PCF Content**

```
OPS_CONFIG_FILE=/data/data046/working/VPW_REMAP.pl.PCF
LIST_FILE=/data/data046/working/GRANULE_LIST/granule_list
SDR_HDF_DIR=/data/data046/working/OUT/viirs_hdf5
SDR_NC_DIR=/data/data046/working/OUT/npp_viirs_sdr
```

VPW Unit Tests
SDR Data Preparation (unit 1)

- **Validation: SDR Data Preparation**
  - » **Perform gap filling and convert SDR data**
    - **Perform gap filling**
      - **Gap filled data location**

/data/data046/working/OUT/viirs_hdf5

      - **Gap filed data name**

```
-rw-r--r-- 1 nobody nobody 12362792 Mar 11 10:24 SVM01_npp_d20130502_t0438171_e0439412_b07829_c20130502110553465700_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362792 Mar 11 10:24 SVM02_npp_d20130502_t0438171_e0439412_b07829_c20130502110545757179_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 17275596 Mar 11 10:24 SVM03_npp_d20130502_t0438171_e0439412_b07829_c20130502110533233128_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 17275596 Mar 11 10:24 SVM04_npp_d20130502_t0438171_e0439412_b07829_c20130502110534156258_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 17275596 Mar 11 10:24 SVM05_npp_d20130502_t0438171_e0439412_b07829_c20130502110536033128_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362792 Mar 11 10:24 SVM06_npp_d20130502_t0438171_e0439412_b07829_c20130502111654623348_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 17275596 Mar 11 10:24 SVM07_npp_d20130502_t0438171_e0439412_b07829_c20130502111658089165_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362792 Mar 11 10:24 SVM08_npp_d20130502_t0438171_e0439412_b07829_c20130502110538017419_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362792 Mar 11 10:24 SVM09_npp_d20130502_t0438171_e0439412_b07829_c20130502110538961429_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362800 Mar 11 10:24 SVM10_npp_d20130502_t0438171_e0439412_b07829_c20130502110545700492_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362800 Mar 11 10:24 SVM11_npp_d20130502_t0438171_e0439412_b07829_c20130502110539928599_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362800 Mar 11 10:24 SVM12_npp_d20130502_t0438171_e0439412_b07829_c20130502110546811697_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 22188408 Mar 11 10:24 SVM13_npp_d20130502_t0438171_e0439412_b07829_c20130502110614330280_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362800 Mar 11 10:24 SVM14_npp_d20130502_t0438171_e0439412_b07829_c20130502110547569639_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362800 Mar 11 10:24 SVM15_npp_d20130502_t0438171_e0439412_b07829_c20130502110607615353_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362800 Mar 11 10:24 SVM16_npp_d20130502_t0438171_e0439412_b07829_c20130502110545216163_noaa_ops.mended.h5


.....
```

VPW Unit Tests
SDR Data Preparation (unit 1)

- **Validation: SDR Data Preparation**
  - » **Perform gap filling and convert SDR data**
    - **Perform gap filling**
      - **Gap filled file name (cont.)**

```
.....
-rw-r--r-- 1 nobody nobody  9873584 Mar 11 10:27 SVM01_npp_d20130502_t0458110_e0459351_b07829_c20130502112335426578_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody  9873584 Mar 11 10:27 SVM02_npp_d20130502_t0458110_e0459351_b07829_c20130502112336394790_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 14788492 Mar 11 10:27 SVM03_npp_d20130502_t0458110_e0459351_b07829_c20130502112333341451_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 14788492 Mar 11 10:27 SVM04_npp_d20130502_t0458110_e0459351_b07829_c20130502112334444932_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 14788492 Mar 11 10:27 SVM05_npp_d20130502_t0458110_e0459351_b07829_c20130502112340278888_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody  9873584 Mar 11 10:27 SVM06_npp_d20130502_t0458110_e0459351_b07829_c20130502112338182835_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 17275356 Mar 11 10:27 SVM07_npp_d20130502_t0458110_e0459351_b07829_c20130502112341471535_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362552 Mar 11 10:27 SVM08_npp_d20130502_t0458110_e0459351_b07829_c20130502112347079956_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody  9873584 Mar 11 10:27 SVM09_npp_d20130502_t0458110_e0459351_b07829_c20130502112346684448_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362560 Mar 11 10:27 SVM10_npp_d20130502_t0458110_e0459351_b07829_c20130502112346429948_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody  9873592 Mar 11 10:27 SVM11_npp_d20130502_t0458110_e0459351_b07829_c20130502112353806025_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362560 Mar 11 10:27 SVM12_npp_d20130502_t0458110_e0459351_b07829_c20130502112350647322_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 22188168 Mar 11 10:27 SVM13_npp_d20130502_t0458110_e0459351_b07829_c20130502112350654352_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362560 Mar 11 10:27 SVM14_npp_d20130502_t0458110_e0459351_b07829_c20130502112345152529_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362560 Mar 11 10:27 SVM15_npp_d20130502_t0458110_e0459351_b07829_c20130502112351862087_noaa_ops.mended.h5
-rw-r--r-- 1 nobody nobody 12362560 Mar 11 10:27 SVM16_npp_d20130502_t0458110_e0459351_b07829_c20130502112349038848_noaa_ops.mended.h5
```

VPW Unit Tests
SDR Data Preparation (unit 1)

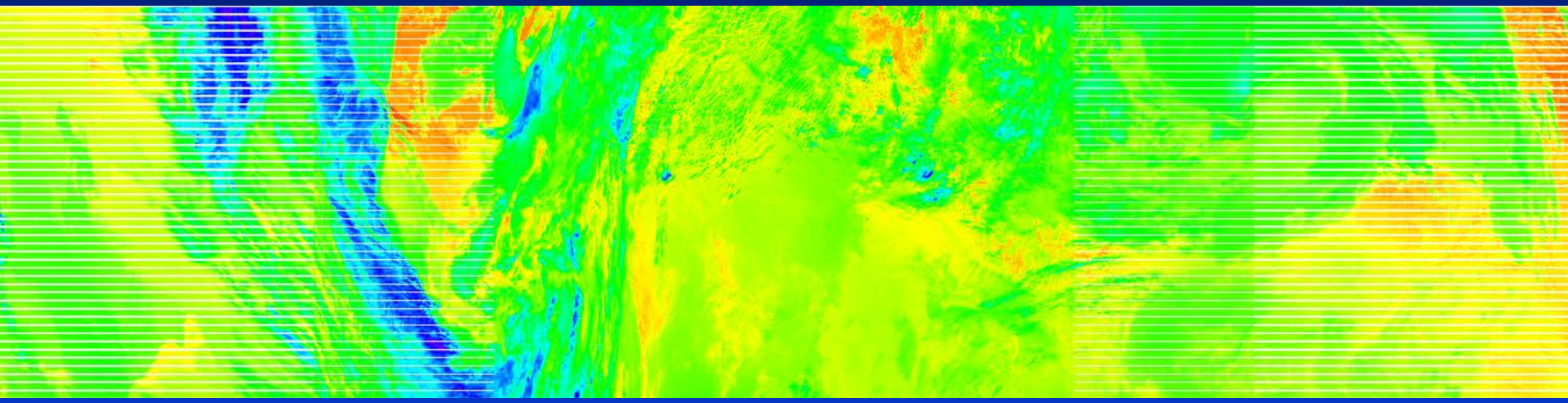
- **Validation: SDR Data Preparation**
  - » **Perform gap filling and convert SDR data**
    - **Perform gap filling**
      - **mender.py log file location and name**

/data/data046/working/sdr/mender.log

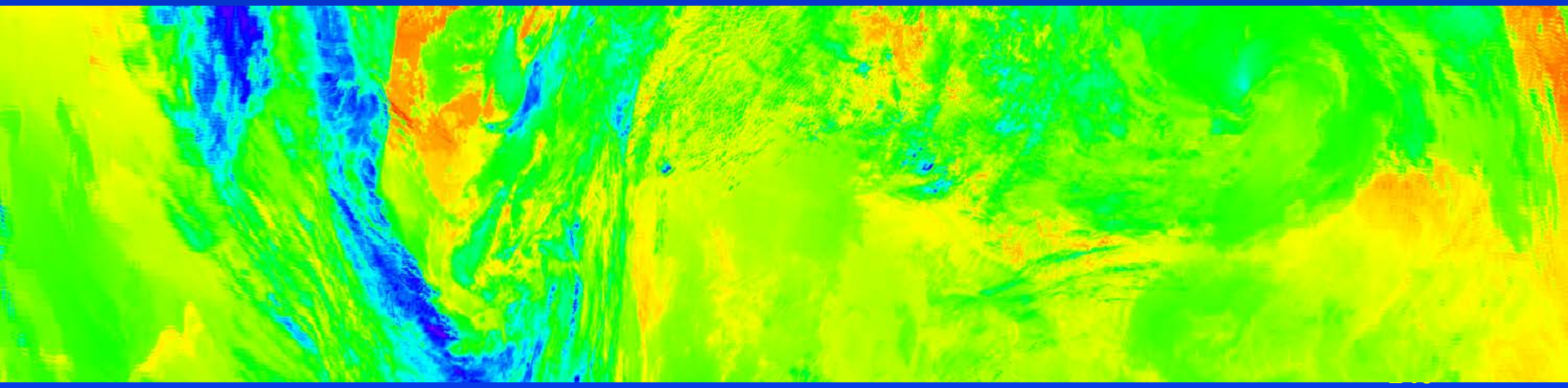
      - **mender.py log file content**

```
Creating:  /data/data046/working/OUT/viirs_hdf5/SVM01_npp_d20130502_t0438171_e0439412_b07829_c20130502110553465700_noaa_ops.mended.h5
Creating:  /data/data046/working/OUT/viirs_hdf5/SVM02_npp_d20130502_t0438171_e0439412_b07829_c20130502110545757179_noaa_ops.mended.h5
Creating:  /data/data046/working/OUT/viirs_hdf5/SVM03_npp_d20130502_t0438171_e0439412_b07829_c20130502110533233128_noaa_ops.mended.h5
…
Creating:  /data/data046/working/OUT/viirs_hdf5/SVM13_npp_d20130502_t0438171_e0439412_b07829_c20130502110614330280_noaa_ops.mended.h5
Creating:  /data/data046/working/OUT/viirs_hdf5/SVM14_npp_d20130502_t0438171_e0439412_b07829_c20130502110547569639_noaa_ops.mended.h5
Creating:  /data/data046/working/OUT/viirs_hdf5/SVM15_npp_d20130502_t0438171_e0439412_b07829_c20130502110607615353_noaa_ops.mended.h5..
…
Creating:  /data/data046/working/OUT/viirs_hdf5/SVM01_npp_d20130502_t0458110_e0459351_b07829_c20130502112335426578_noaa_ops.mended.h5
Creating:  /data/data046/working/OUT/viirs_hdf5/SVM02_npp_d20130502_t0458110_e0459351_b07829_c20130502112336394790_noaa_ops.mended.h5
Creating:  /data/data046/working/OUT/viirs_hdf5/SVM03_npp_d20130502_t0458110_e0459351_b07829_c20130502112333341451_noaa_ops.mended.h5
….
Creating:  /data/data046/working/OUT/viirs_hdf5/SVM14_npp_d20130502_t0458110_e0459351_b07829_c20130502112345152529_noaa_ops.mended.h5
Creating:  /data/data046/working/OUT/viirs_hdf5/SVM15_npp_d20130502_t0458110_e0459351_b07829_c20130502112351862087_noaa_ops.mended.h5
Creating:  /data/data046/working/OUT/viirs_hdf5/SVM16_npp_d20130502_t0458110_e0459351_b07829_c20130502112349038848_noaa_ops.mended.h5
```

VPW Unit Tests
SDR Data Preparation (unit 1)



VIIRS Channel 15 BT: before gap filling



VIIRS Channel 15 BT: after gap filling

VPW Unit Tests
SDR Data Preparation (unit 1)

- **Validation: SDR Data Preparation**
  - » **Perform gap filling and convert SDR data**
    - **Convert SDR data to NetCDF**
      - **NetCDF location**

/data/data046/working/OUT/npp_viirs_sdr

      - **NetCDF file name**

```
-rw-r--r-- 1 nobody nobody 81191224 Mar 11 10:24 GMTCO_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 14803122 Mar 11 10:24 IICMO_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 12364552 Mar 11 10:24 SVM01_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 12364552 Mar 11 10:24 SVM02_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 17277028 Mar 11 10:24 SVM03_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 17277028 Mar 11 10:24 SVM04_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 17277028 Mar 11 10:24 SVM05_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 12364552 Mar 11 10:24 SVM06_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 17277028 Mar 11 10:24 SVM07_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 12364552 Mar 11 10:24 SVM08_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 12364552 Mar 11 10:24 SVM09_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 12364560 Mar 11 10:24 SVM10_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 12364560 Mar 11 10:24 SVM11_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 12364936 Mar 11 10:24 SVM12_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 22189840 Mar 11 10:24 SVM13_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 12364936 Mar 11 10:24 SVM14_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 12364936 Mar 11 10:24 SVM15_npp_20130502_043852_d20130502_t0438171.nc
-rw-r--r-- 1 nobody nobody 12364936 Mar 11 10:24 SVM16_npp_20130502_043852_d20130502_t0438171.nc


....
```

VPW Unit Tests
SDR Data Preparation (unit 1)

- **Validation: SDR Data Preparation**
  - » **Perform gap filling and convert SDR data**
    - **Convert SDR data to NetCDF**
      - **Conversion log file location and name**

/data/data046/working/sdr/convert_viirs_hd5_2_nc4.pl.log

      - **Conversion log file content**

```
Starting at Tue Mar 11 10:24:39 EDT 2014
==============================
 Done on converting /data/data046/mfan/working/OUT/viirs_hdf5/working/GMTCO_npp_d20130502_t0438171_e0439412_b07829_c20130502105911542952_noaa_ops.h5
 Done on converting /data/data046/mfan/working/OUT/viirs_hdf5/working/IICMO_npp_d20130502_t0438171_e0439412_b07829_c20130502110556311843_noaa_ops.h5
 Done on converting /data/data046/mfan/working/OUT/viirs_hdf5/working/SVM01_npp_d20130502_t0438171_e0439412_b07829_c20130502110553465700_noaa_ops.mended.h5
 Done on converting /data/data046/mfan/working/OUT/viirs_hdf5/working/SVM02_npp_d20130502_t0438171_e0439412_b07829_c20130502110545757179_noaa_ops.mended.h5
....
 Done on converting /data/data046/mfan/working/OUT/viirs_hdf5/working/SVM15_npp_d20130502_t0438171_e0439412_b07829_c20130502110607615353_noaa_ops.mended.h5
 Done on converting /data/data046/mfan/working/OUT/viirs_hdf5/working/SVM16_npp_d20130502_t0438171_e0439412_b07829_c20130502110545216163_noaa_ops.mended.h5
Scan Time = 20130502_043852

Starting at Tue Mar 11 10:27:39 EDT 2014
==============================
 Done on converting /data/data046/mfan/working/OUT/viirs_hdf5/working/GMTCO_npp_d20130502_t0458110_e0459351_b07829_c20130502111932325389_noaa_ops.h5
 Done on converting /data/data046/mfan/working/OUT/viirs_hdf5/working/IICMO_npp_d20130502_t0458110_e0459351_b07829_c20130502112343455424_noaa_ops.h5
 Done on converting /data/data046/mfan/working/OUT/viirs_hdf5/working/SVM01_npp_d20130502_t0458110_e0459351_b07829_c20130502112335426578_noaa_ops.mended.h5
 Done on converting /data/data046/mfan/working/OUT/viirs_hdf5/working/SVM02_npp_d20130502_t0458110_e0459351_b07829_c20130502112336394790_noaa_ops.mended.h5

....

 Done on converting /data/data046/mfan/working/OUT/viirs_hdf5/working/SVM15_npp_d20130502_t0458110_e0459351_b07829_c20130502112351862087_noaa_ops.mended.h5
 Done on converting /data/data046/mfan/working/OUT/viirs_hdf5/working/SVM16_npp_d20130502_t0458110_e0459351_b07829_c20130502112349038848_noaa_ops.mended.h5
```

VPW Unit Tests
SDR Data Preparation (unit 1)

- **Validation: SDR Data Preparation**
  - » **Perform gap filling and convert SDR data**
    - **Update status file**
      - **Status file location and name**

```
/data/data046/working/GRANULE_LIST/granule_list
```

      - **Status file content**

```
2013 05 02 d20130502_t0438171_e0439412_b07829 U U
2013 05 02 d20130502_t0439425_e0441066_b07829 U U
2013 05 02 d20130502_t0441079_e0442320_b07829 U U
2013 05 02 d20130502_t0442333_e0443575_b07829 U U
....

2013 05 02 d20130502_t0453547_e0455189_b07829 U U
2013 05 02 d20130502_t0455201_e0456443_b07829 U U
2013 05 02 d20130502_t0456455_e0458097_b07829 U U
2013 05 02 d20130502_t0458110_e0459351_b07829 U U
```

(before)

```
2013 05 02 d20130502_t0438171_e0439412_b07829 D U
2013 05 02 d20130502_t0439425_e0441066_b07829 D U
2013 05 02 d20130502_t0441079_e0442320_b07829 D U
2013 05 02 d20130502_t0442333_e0443575_b07829 D U
....

2013 05 02 d20130502_t0453547_e0455189_b07829 D U
2013 05 02 d20130502_t0455201_e0456443_b07829 D U
2013 05 02 d20130502_t0456455_e0458097_b07829 D U
2013 05 02 d20130502_t0458110_e0459351_b07829 D U
```

(after)

VPW Unit Tests
SDR Data Preparation (unit 1)

- **Validation: SDR Data Preparation**
  - » **Perform gap filling and convert SDR data**
    - **Create conversion PSF file**
      - **Conversion PSF location and name**

```
/data/data046/working/sdr/NVPWPS_Preproc_SDR.pl.PSF
```

      - **Conversion PSF file content**

```
/data/data046/working/OUT/npp_viirs_sdr/GMTCO_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/IICMO_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM01_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM02_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM03_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM04_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM05_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM06_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM07_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM08_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM09_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM10_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM11_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM12_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM13_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM14_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM15_npp_20130502_043852_d20130502_t0438171.nc
/data/data046/working/OUT/npp_viirs_sdr/SVM16_npp_20130502_043852_d20130502_t0438171.nc

.....
```

VPW Unit Tests
SDR Data Preparation (unit 1)

- **Validation: SDR Data Preparation**
  - » **Update VPW_REMAP.pl.log**
    - **Log file location**

```
/data/data046/working/
```

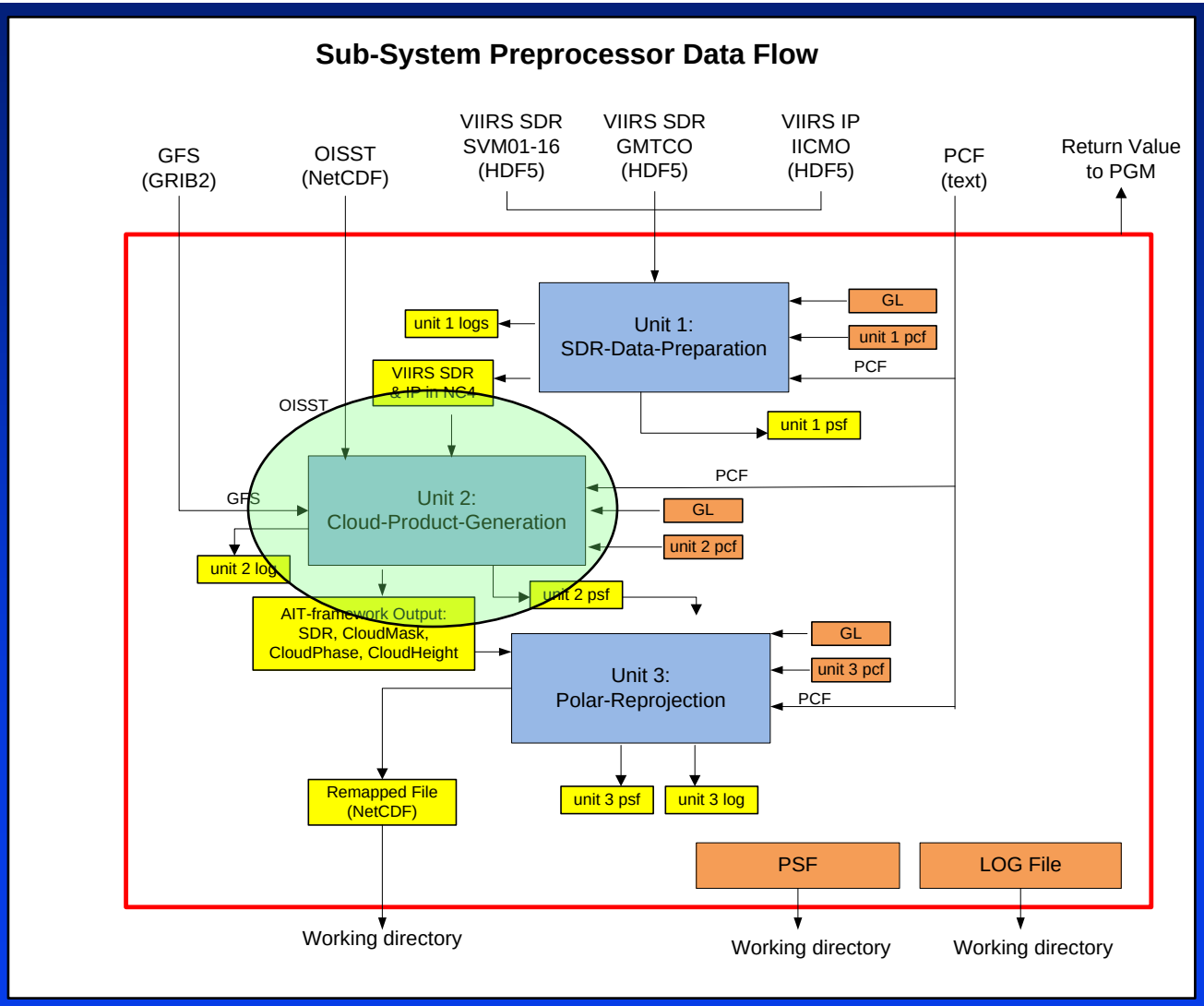
    - **Log file content**

```
Processing ....
  - NORTH
  - /data/data046/working/GRANULE_LIST/granule_list

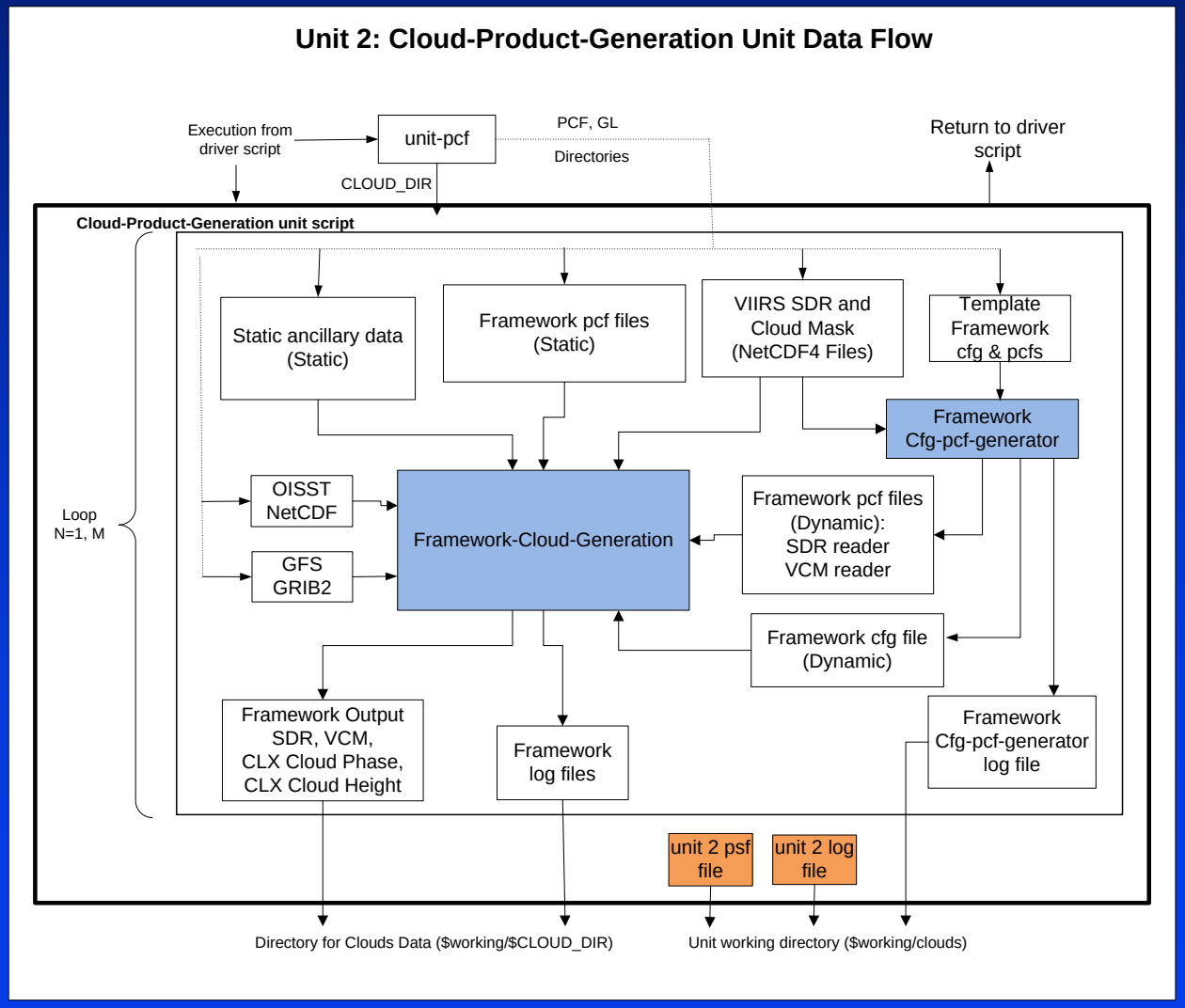
   SDR
   ------------
   Creating SDR working directory: /data/data046/working/sdr
   Creating PCF file: /data/data046/working/sdr/NVPWPS_Preproc_SDR.pl.PCF
   Running NVPWPS_Preproc_SDR.pl
     - log:  /data/data046/working/sdr/NVPWPS_Preproc_SDR.pl.log
       - sublog:  /data/data046/working/sdr/mender.log
       -          /data/data046/working/sdr/convert_viirs_hd5_2_nc4.pl.log
       -              /data/data046/working/sdr/h5augjpss.log

real    7m19.301s
user    2m19.966s
sys     0m36.113s
    running NVPWPS_Preproc_SDR.pl: OK
```

Preprocessor
Cloud Product Generation (unit 2)



Preprocessor
Cloud Product Generation (unit 2)



VPW Unit Tests
Cloud Product Generation (unit 2)

- Cloud Product Generation
  - » Purpose: Generate cloud product (cloud mask, cloud phase, cloud height) for each selected granule
  - » Major Steps
    - Create directory $working/clouds/and enter into it
    - Make necessary links for cloud product generation
    - Generate cloud PCF file NVPWPS_Preproc_Clouds.pl.PCF
    - Generate cloud products
      - ➢ Call NVPWPS_Preproc_Clouds.pl
        - o Create cloud configuration and overwritten PCF files
        - ✓ call get_cfg_pcf_for_clouds.pl
        - o Execute pcf_framework.exe to generate cloud products
        - o Update status file
        - ✓ call update_pre_process_status.pl
    - Create cloud PSF for NVPWPS_Preproc_Clouds.pl
    - Update log file VPW_REMAP.pl.log

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - **Create $working/clouds**

```
/data/data046/working/clouds
```

  - **Links for generation of cloud products**

```
lrwxrwxrwx 1 nobody nobody   46 Mar 11 10:27 algorithm_ancillary -> /data/data046/mfan/working/algorithm_ancillary
lrwxrwxrwx 1 nobody nobody   45 Mar 11 10:27 CustomProductLists -> /data/data046/mfan/working/CustomProductLists
lrwxrwxrwx 1 nobody nobody   38 Mar 11 10:27 Default_PCF -> /data/data046/mfan/working/Default_PCF
lrwxrwxrwx 1 nobody nobody   46 Mar 11 10:27 framework_ancillary -> /data/data046/mfan/working/framework_ancillary
lrwxrwxrwx 1 nobody nobody   41 Mar 11 10:27 PCF_Overwrites -> /data/data046/mfan/working/PCF_Overwrites
lrwxrwxrwx 1 nobody nobody   39 Mar 11 10:27 run_wgrib.pl -> /data/data046/mfan/working/run_wgrib.pl
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - **Generate Cloud PCF file**
    - **Cloud PCF file location and name**

```
/data/data046/working/clouds/NVPWPS_Preproc_Clouds.pl.PCF
```

    - **Cloud PCF file content**

```
OPS_CONFIG_FILE=/data/data046/working/VPW_REMAP.pl.PCF
LIST_FILE=/data/data046/working/GRANULE_LIST/granule_list
SDR_NC_DIR=/data/data046/working/OUT/npp_viirs_sdr
CLOUD_DIR=/data/data046/working/OUT/viirs_results_clouds
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - **Generate cloud products**
    - **Create cloud configuration files**
      - **Cloud configuration file location**

/data/data046/working/OUT/viirs_results_clouds/configs

      - **Cloud configuration file name**

```
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:27 AWG_NPP_VIIRS_Clouds_2013122_0438_52_d201305020438.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:27 AWG_NPP_VIIRS_Clouds_2013122_0440_17_d201305020439.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:29 AWG_NPP_VIIRS_Clouds_2013122_0441_42_d201305020441.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:29 AWG_NPP_VIIRS_Clouds_2013122_0443_08_d201305020442.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:30 AWG_NPP_VIIRS_Clouds_2013122_0444_33_d201305020443.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:30 AWG_NPP_VIIRS_Clouds_2013122_0445_59_d201305020445.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:32 AWG_NPP_VIIRS_Clouds_2013122_0447_24_d201305020446.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:32 AWG_NPP_VIIRS_Clouds_2013122_0448_49_d201305020448.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:34 AWG_NPP_VIIRS_Clouds_2013122_0450_15_d201305020449.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:34 AWG_NPP_VIIRS_Clouds_2013122_0451_38_d201305020451.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:36 AWG_NPP_VIIRS_Clouds_2013122_0453_04_d201305020452.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:36 AWG_NPP_VIIRS_Clouds_2013122_0454_29_d201305020453.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:37 AWG_NPP_VIIRS_Clouds_2013122_0455_55_d201305020455.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:37 AWG_NPP_VIIRS_Clouds_2013122_0457_20_d201305020456.cfg
-rw-r--r-- 1 nobody nobody 2856 Mar 11 10:39 AWG_NPP_VIIRS_Clouds_2013122_0458_46_d201305020458.cfg
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - » **Generate cloud products**
    - **Create cloud configuration files**
      - **File content (an example)**

```
# Satellite info
SATELLITE: NPP
INSTRUMENT: VIIRS

SEGMENT_MODE: SCANLINE

#
# Data dimensions
#
ROWS: 768
COLUMNS: 3200
CHANNELS: 16

#
# PCF Framework Settings
#

#
# Segment details
#
ROW_START: 1
ROW_END: 768
COLUMN_START: 1
COLUMN_END: 3200
#SEGMENT_SIZE: 200
SEGMENT_SIZE: 300
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - **Generate cloud products**
    - **Create cloud configuration files**
      - **File content (cont.)**

```
# RUN_MODE - USER_DEFINED_PCF or ALLOW_DEFAULT_PCF
# USER_DEFINED forces user to specify each product in this file
# instead of relying on dependencies to determine if that product should run
# and using the default pcf for that product
#
RUN_MODE: ALLOW_DEFAULT_PCF

#
# DEPENDENCY_INFO_LOCATION - ALLOW_IN_MAIN or PCF_ONLY
# ALLOW_IN_MAIN allows specific dependencies in pcf files to be
# blank, but a type must be declared in the default.
# If they specific dependencies are missing the framework will search
# the main list to see if any of that type has been declared and
# add them to the list as a dependency
# PCF_ONLY forces user to specify all PCF dependency information
# in that PCFs
#
#
DEPENDENCY_INFO_LOCATION: ALLOW_IN_MAIN

#
# Output Directory
#
OUTPUT_DIRECTORY: /data/data046/working/OUT/viirs_results_clouds/results/
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - » **Generate cloud products**
    - **Create cloud configuration files**
      - **File content (cont.)**

```
# Output Option
# NETCDF_SEGMENT - creates a NetCDF file for each segment
# NETCDF_ALL - creates a single file for the full run
# Important to note, each product (team) has its own output file
# Specifying the ALL part will not create a single file with everything
# in it.
#
OUTPUT_FILE_OPTION: NETCDF_ALL
FILE_VERSION: NETCDF_V4


#
# Specify here what you want to run
#
RUN:

#
# For satellite data, specify actual data file instead of pcf
#
SATELLITE_DATA:
NPP_VIIRS_SDR_DATA = /data/data046/working/OUT/npp_viirs_sdr/SVM01_npp_20130502_043852_d20130502_t0438171.nc +
/data/data046/working/OUT/viirs_results_clouds/configs/NPP_VIIRS_SDR_Data_2013122_0438_52_d201305020438.pcf

CLOUD_MASK:
NPP_VIIRS_CLOUD_MASK =
/data/data046/working/OUT/viirs_results_clouds/configs/NPP_VIIRS_Cloud_Mask_2013122_0438_52_d201305020438.pcf
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - **Generate cloud products**
    - **Create cloud configuration files**
      - **File content (cont.)**

```
LAND_MASK:
LAND_MASK_NASA_1KM = PCF_Overwrites/Land_Mask.pcf

SST:
OISST_DAILY_QTRDEG = PCF_Overwrites/OISST_Daily_QtrDeg.pcf

NWP_DATA:
NWP_GFS = PCF_Overwrites/NWP_Data_grib2_0.5deg.pcf


SURFACE_ELEVATION:
SFC_ELEV_GLOBE_1KM = PCF_Overwrites/Surface_Elevation.pcf

SURFACE_EMISSIVITY:
SFC_EMISS_SEEBOR = PCF_Overwrites/Surface_Emissivity.pcf

RTM:
CRTM = PCF_Overwrites/CRTM.pcf

COAST_MASK:
COAST_MASK_NASA_1KM = PCF_Overwrites/Coast_Mask.pcf

DESERT_MASK:
DESERT_MASK_CALCLTED = PCF_Overwrites/Desert_Mask.pcf

SNOW_MASK:
SNOW_MASK_NWP =  PCF_Overwrites/Snow_Mask.pcf
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - » **Generate cloud products**
    - **Create cloud configuration files**
      - **File content (cont.)**

```
PSEUDO_EMISSIVITY:
GOESR_ABI_CHN7_EMISS = PCF_Overwrites/Pseudo_Emissivity.pcf

SURFACE_TYPE:
SFC_TYPE_AVHRR_1KM = PCF_Overwrites/Surface_Type.pcf

CLOUD_PHASE:
AWG_CLX_CLOUD_PHASE = PCF_Overwrites/AWG_Clx_Cloud_Phase.pcf

CLOUD_HEIGHT:
AWG_CLX_CLOUD_HEIGHT = PCF_Overwrites/AWG_Clx_Cloud_Height.pcf
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - » **Generate cloud products**
    - **Create cloud Overwritten PCF file**
      - **Cloud overwritten PCF file location**

```
/data/data046/working/OUT/viirs_results_clouds/configs
```

      - **Cloud overwritten PCF file name**

```
-rw-r--r-- 1 nobody nobody 1463 Mar 11 10:27 NPP_VIIRS_Cloud_Mask_2013122_0438_52_d201305020438.pcf
-rw-r--r-- 1 nobody nobody 1463 Mar 11 10:27 NPP_VIIRS_Cloud_Mask_2013122_0440_17_d201305020439.pcf
-rw-r--r-- 1 nobody nobody 1463 Mar 11 10:29 NPP_VIIRS_Cloud_Mask_2013122_0441_42_d201305020441.pcf
-rw-r--r-- 1 nobody nobody 1463 Mar 11 10:29 NPP_VIIRS_Cloud_Mask_2013122_0443_08_d201305020442.pcf
-rw-r--r-- 1 nobody nobody 1463 Mar 11 10:30 NPP_VIIRS_Cloud_Mask_2013122_0444_33_d201305020443.pcf
-rw-r--r-- 1 nobody nobody 1463 Mar 11 10:30 NPP_VIIRS_Cloud_Mask_2013122_0445_59_d201305020445.pcf
-rw-r--r-- 1 nobody nobody 1463 Mar 11 10:32 NPP_VIIRS_Cloud_Mask_2013122_0447_24_d201305020446.pcf
-rw-r--r-- 1 nobody nobody 1463 Mar 11 10:32 NPP_VIIRS_Cloud_Mask_2013122_0448_49_d201305020448.pcf

....

-rw-r--r-- 1 nobody nobody 1865 Mar 11 10:27 NPP_VIIRS_SDR_Data_2013122_0438_52_d201305020438.pcf
-rw-r--r-- 1 nobody nobody 1865 Mar 11 10:27 NPP_VIIRS_SDR_Data_2013122_0440_17_d201305020439.pcf
-rw-r--r-- 1 nobody nobody 1865 Mar 11 10:29 NPP_VIIRS_SDR_Data_2013122_0441_42_d201305020441.pcf
-rw-r--r-- 1 nobody nobody 1865 Mar 11 10:29 NPP_VIIRS_SDR_Data_2013122_0443_08_d201305020442.pcf
-rw-r--r-- 1 nobody nobody 1865 Mar 11 10:30 NPP_VIIRS_SDR_Data_2013122_0444_33_d201305020443.pcf
-rw-r--r-- 1 nobody nobody 1865 Mar 11 10:30 NPP_VIIRS_SDR_Data_2013122_0445_59_d201305020445.pcf
-rw-r--r-- 1 nobody nobody 1865 Mar 11 10:32 NPP_VIIRS_SDR_Data_2013122_0447_24_d201305020446.pcf
-rw-r--r-- 1 nobody nobody 1865 Mar 11 10:32 NPP_VIIRS_SDR_Data_2013122_0448_49_d201305020448.pcf
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - » **Generate cloud products**
    - **Create cloud Overwritten PCF file**
      - **Cloud overwritten PCF file content: NPP_VIIRS_Cloud_Mask_2013122_0438_52_d201305020438.pcf**

```
NPP_VIIRS_CLOUD_MASK
ANCILLARY

#
# Set flag for output
#
OUTPUT_TO_FILE: Y

##############################################
# This section needed for Default PCF only #
##############################################

#
# Structure information used for code generation
#

# Default nav resolution - leave blank if only 1 nav structure in data structure
# If multiple resolutions for Nav structure, choose a default most products
# will be processed on.
#
# Ex. MSG SEVIRI data would be left blank
#   DEFAULT_NAV_RESOLUTION:
#
# Ex. GOESR ABI data would be 2km
#   DEFAULT_NAV_RESOLUTION: 2km
#
DEFAULT_NAV_RESOLUTION:

# For GOESR either Framework_Input_Structure or Framework_Output_Structure
MAIN_STRUCTURE_TYPE: Framework_Output_Structure
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - **Generate cloud products**
    - **Create cloud Overwritten PCF file**
      - **Cloud overwritten PCF file content: NPP_VIIRS_Cloud_Mask_2013122_0438_52_d201305020438.pcf (cont.)**

```
# Algorithm structure type
ALGORITHM_STRUCTURE_TYPE: Cloud_Mask_Structure
#
# Algorithm structure name (long)
#
# Output example: AWG_CLOUD_MASK
# ALGORITHM_STRUCTURE_NAME: AWG_Cloud_Mask
#
# Input example: SFC_EMISS_SEEBOR
# ALGORITHM_STRUCTURE_NAME: GOESR_ABI_Data.Ancil_2km.SfcEmiss_Seebor
#
#
ALGORITHM_STRUCTURE_NAME: NPP_VIIRS_CLOUD_MASK

################################
# End Default PCF only Section #
################################

DEPENDENCIES:
SATELLITE_DATA:

OTHER:
Filename: /data/data046/working/OUT/npp_viirs_sdr/IICMO_npp_20130502_043852_d20130502_t0438171.nc
CldMask_Packed_Constant: 1
Flag_Constant: 1
Meta10: 10
Meta20: 20

END NPP_VIIRS_CLOUD_MASK
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - » **Generate cloud products**
    - **Create cloud Overwritten PCF file**
      - **Cloud overwritten PCF file content: NPP_VIIRS_SDR_Data_2013122_0438_52_d201305020438.pcf**

```
NPP_VIIRS_SDR_DATA
ANCILLARY

#
# Set flag for output
#
OUTPUT_TO_FILE: Y

##########################################
# This section needed for Default PCF only #
##########################################

#
# Default nav resolution - leave blank if only 1 nav structure in data structure
# If multiple resolutions for Nav structure, choose a default most products
# will be processed on.
#
# Ex. MSG SEVIRI data would be left blank
#   DEFAULT_NAV_RESOLUTION:
#
# Ex. GOESR ABI data would be 2km
#   DEFAULT_NAV_RESOLUTION: 2km
#
DEFAULT_NAV_RESOLUTION:

# For GOESR either Framework_Input_Structure or Framework_Output_Structure
MAIN_STRUCTURE_TYPE: Framework_Input_Structure

# Algorithm structure type
ALGORITHM_STRUCTURE_TYPE: Generic_Sat_Data_Struct
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - » **Generate cloud products**
    - **Create cloud Overwritten PCF file**
      - **Cloud overwritten PCF file content: NPP_VIIRS_SDR_Data_2013122_0438_52_d201305020438.pcf (cont.)**

```
# Algorithm structure name (long)
#
# Output example: AWG_CLOUD_MASK
# ALGORITHM_STRUCTURE_NAME: AWG_Cloud_Mask
#
# Input example: SFC_EMISS_SEEBOR
# ALGORITHM_STRUCTURE_NAME: GOESR_ABI_Data.Ancil_2km.SfcEmiss_Seebor
#
#
ALGORITHM_STRUCTURE_NAME: NPP_VIIRS_SDR_Data

################################
# End Default PCF only Section #
################################

OTHER:
Sat_Name: NPP
Instrument_Name: VIIRS
#Sat_ID:00
Sat_ID: 9

# 16 M-bands ( DNB is not included)
Total_Channels: 16
#
Total_Rows: 768
Total_Columns: 3200

#Vis bands( 11 ===> 12 w. M12 added fro reflectance)
#ReflChannels: 11
ReflChannels: 12
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - » **Generate cloud products**
    - **Create cloud Overwritten PCF file**
      - **Cloud overwritten PCF file content: NPP_VIIRS_SDR_Data_2013122_0438_52_d201305020438.pcf (cont.)**

```
#IR bands
IR_Channels: 5

ChnMapTotalChns: 16

#Not used for now
AncilFilename: framework_ancillary/ancil_file/npp_viirs_ancil.nc

MaxSatZenAngle: 120

#
# For Planck
#
PlanckMaxTemp: 340.0
PlanckMinTemp: 179.0
PlanckDeltaTemp: 1.0

#
# Need to figure out what version we have, if applicable.
#
# Version: 1.0


#
# File containing lats/lons/etc
#
StaticFile: /data/data046/working/OUT/npp_viirs_sdr/GMTCO_npp_20130502_043852_d20130502_t0438171.nc

END NPP_VIIRS_SDR_DATA
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - » **Generate cloud products**
    - **Execute pcf_framework.exe**
      - **Cloud product file location**

/data/data046/working/OUT/viirs_results_clouds/results

      - **Cloud product file name**

```
-rw-r--r-- 1 nobody nobody  98331741 Mar 11 10:29 NPP_VIIRS_2013122_0438_52_AWG_CLX_CLOUD_HEIGHT.nc
-rw-r--r-- 1 nobody nobody 127813247 Mar 11 10:29 NPP_VIIRS_2013122_0438_52_AWG_CLX_CLOUD_PHASE.nc
-rw-r--r-- 1 nobody nobody  41801841 Mar 11 10:29 NPP_VIIRS_2013122_0438_52_NPP_VIIRS_CLOUD_MASK.nc
-rw-r--r-- 1 nobody nobody 385900799 Mar 11 10:29 NPP_VIIRS_2013122_0438_52_NPP_VIIRS_SDR_DATA.nc

.....

-rw-r--r-- 1 nobody nobody  98331741 Mar 11 10:40 NPP_VIIRS_2013122_0458_46_AWG_CLX_CLOUD_HEIGHT.nc
-rw-r--r-- 1 nobody nobody 127813247 Mar 11 10:40 NPP_VIIRS_2013122_0458_46_AWG_CLX_CLOUD_PHASE.nc
-rw-r--r-- 1 nobody nobody  41801841 Mar 11 10:40 NPP_VIIRS_2013122_0458_46_NPP_VIIRS_CLOUD_MASK.nc
-rw-r--r-- 1 nobody nobody 385900799 Mar 11 10:40 NPP_VIIRS_2013122_0458_46_NPP_VIIRS_SDR_DATA.nc
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - » **Generate cloud products**
    - **Update status file**
      - **Status file location and name**

/data/data046/working/GRANULE_LIST/granule_list

      - **Status file content**

```
2013 05 02 d20130502_t0438171_e0439412_b07829 D U
2013 05 02 d20130502_t0439425_e0441066_b07829 D U
2013 05 02 d20130502_t0441079_e0442320_b07829 D U
2013 05 02 d20130502_t0442333_e0443575_b07829 D U
....
2013 05 02 d20130502_t0453547_e0455189_b07829 D U
2013 05 02 d20130502_t0455201_e0456443_b07829 D U
2013 05 02 d20130502_t0456455_e0458097_b07829 D U
2013 05 02 d20130502_t0458110_e0459351_b07829 D U
```

(before)

```
2013 05 02 d20130502_t0438171_e0439412_b07829 D D
2013 05 02 d20130502_t0439425_e0441066_b07829 D D
2013 05 02 d20130502_t0441079_e0442320_b07829 D D
2013 05 02 d20130502_t0442333_e0443575_b07829 D D
....
2013 05 02 d20130502_t0453547_e0455189_b07829 D D
2013 05 02 d20130502_t0455201_e0456443_b07829 D D
2013 05 02 d20130502_t0456455_e0458097_b07829 D D
2013 05 02 d20130502_t0458110_e0459351_b07829 D D
```

(after)

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - **Generate cloud products**
    - **Cloud log file location and name**

/data/data046/working/clouds/NVPWPS_Preproc_Clouds.pl.log

- **Cloud log file content**

```
NVPWPS_Preproc_Clouds.pl is now starting at 20131015-10:36:19

Current Granule List File is: /data/data046/working/GRANULE_LIST/granule_list_SDR_NL3
  Orbit Number = 07829
  First Granule Start Time = 20130502_0438

Processing 20130502_043817
-------------------------
scan_time_string in jday = 2013122_0438_52

….

Processing 20130502_045811
-------------------------
scan_time_string in jday = 2013122_0458_46
DYLAN: number of threads = 15
Maximum threads assiged at one time = 2
DYLAN: icount is 1
DYLAN: Looping through threads 0+1
DYLAN: Looping through threads 1+1
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - » **Generate cloud products**
    - **Cloud log file content (cont.)**

```
……
cfg_clouds =
/data/data046/mfan/working/OUT/viirs_results_clouds/configs/AWG_NPP_VIIRS_Clouds_2013122_0440_17_d201305020439.cfg
running get_cfg_pcf_for_clouds.pl: OK

cfg_clouds =
/data/data046/mfan/working/OUT/viirs_results_clouds/configs/AWG_NPP_VIIRS_Clouds_2013122_0438_52_d201305020438.cfg
running get_cfg_pcf_for_clouds.pl: OK

…

cfg_clouds =
/data/data046/mfan/working/OUT/viirs_results_clouds/configs/AWG_NPP_VIIRS_Clouds_2013122_0458_46_d201305020458.cfg
running get_cfg_pcf_for_clouds.pl: OK

/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/bin/pcf_framework.exe
/data/data046/mfan/working/OUT/viirs_results_clouds/configs/AWG_NPP_VIIRS_Clouds_2013122_0458_46_d201305020458.cfg
>/data/data046/mfan/working/OUT/viirs_results_clouds/logs/AWG_cldHgt_2013122_0458_46_d201305020458.log OK
Numer fo granule = 15
 NVPWPS_Preproc_Clouds.pl is now completing successfully at: 20140311-10:40:21
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - **Create cloud PSF file for NVPWPS_Preproc_Clouds.pl**
    - **Cloud PSF file**

```
/data/data046/working/clouds/NVPWPS_Preproc_Clouds.pl.PSF
```

    - **Cloud PSF file content**

```
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0438_52_NPP_VIIRS_SDR_DATA.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0438_52_NPP_VIIRS_CLOUD_MASK.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0438_52_AWG_CLX_CLOUD_PHASE.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0438_52_AWG_CLX_CLOUD_HEIGHT.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0440_17_NPP_VIIRS_SDR_DATA.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0440_17_NPP_VIIRS_CLOUD_MASK.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0440_17_AWG_CLX_CLOUD_PHASE.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0440_17_AWG_CLX_CLOUD_HEIGHT.nc

...

/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0457_20_NPP_VIIRS_SDR_DATA.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0457_20_NPP_VIIRS_CLOUD_MASK.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0457_20_AWG_CLX_CLOUD_PHASE.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0457_20_AWG_CLX_CLOUD_HEIGHT.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0458_46_NPP_VIIRS_SDR_DATA.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0458_46_NPP_VIIRS_CLOUD_MASK.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0458_46_AWG_CLX_CLOUD_PHASE.nc
/data/data046/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0458_46_AWG_CLX_CLOUD_HEIGHT.nc
```

VPW Unit Tests
Cloud Product Generation (unit 2)

- **Validation: Cloud Product Generation**
  - **Update VPW_REMAP.pl.log**
    - **Log file location and name**

```
/data/data046/working/VPW_REMAP.pl.log
```

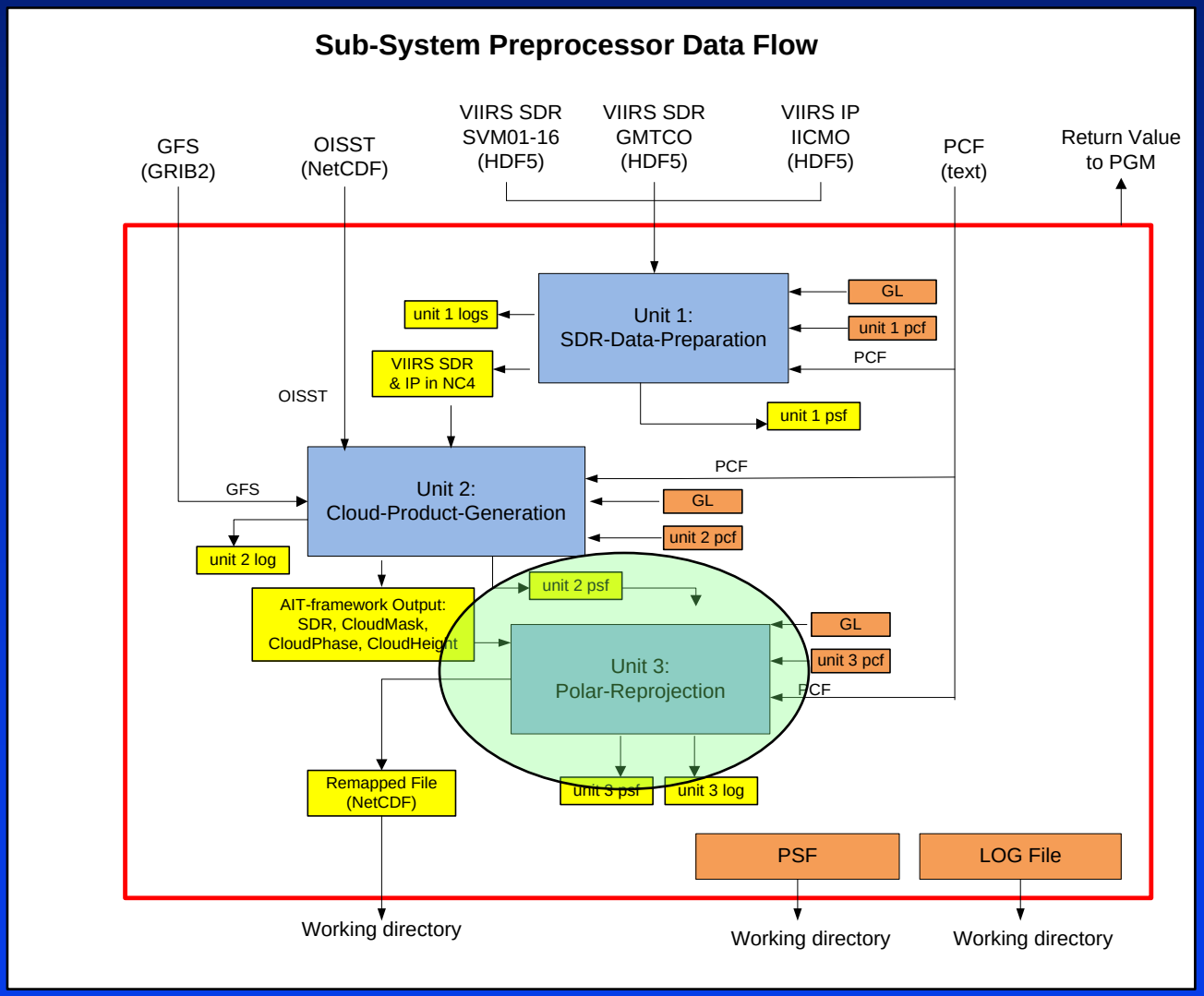
    - **Log file updated/appended**

```
…..

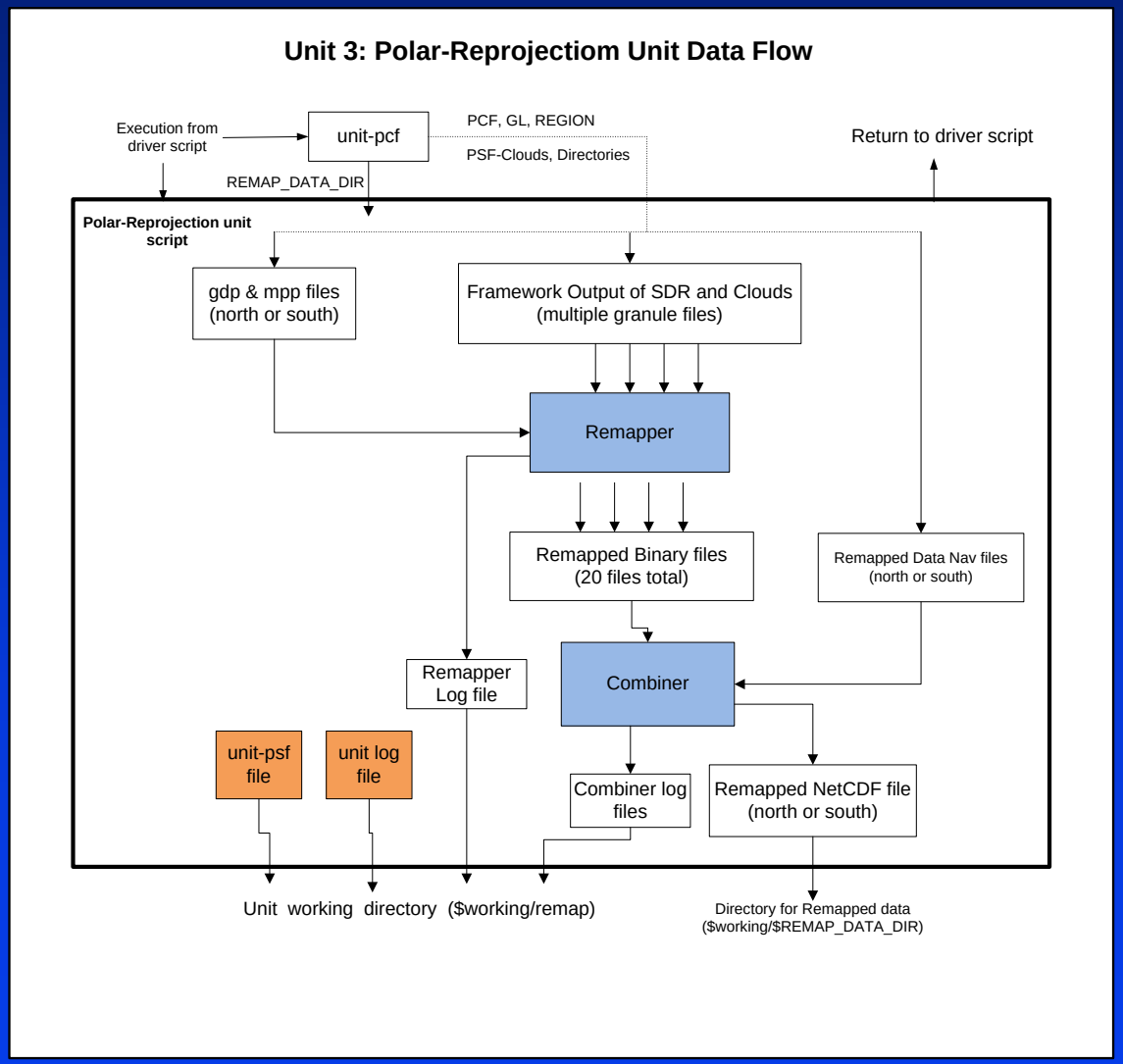
Clouds
  -------------
  Creating Clouds working directory: /data/data046/working/clouds
  Creating PCF file: /data/data046/working/clouds/NVPWPS_Preproc_Clouds.pl.PCF
  Running NVPWPS_Preproc_Clouds.pl
   - /data/data046/working/clouds/NVPWPS_Preproc_Clouds.pl.log
     - sublog:  /data/data046/working/clouds/get_cfg_pcf_for_clouds.pl.log

real    43m15.330s
user    18m4.769s
sys     3m42.761s
  running NVPWPS_Preproc_Clouds.pl: OK
```

Preprocessor
Unit3: Polar Re-projection



Preprocessor
Polar Re-projection Unit



VPW Unit Tests
Polar Re-Projection (unit 3)

- **Polar Re-Projection**
  - » **Purpose: remap granule data to one polar stereographic project file**
  - » **Major Steps:**
    - **Create directory $working/remap and enter into it**
    - **Generate remap PCF file NVPWPS_Preproc_Remap.pl.PCF**
    - **Create the PS projection NetCDF file**
      - ➢ **Call NVPWPS_Preproc_Remap.pl**
        - o **Convert granule NetCDF files (SDR, cloud mask, cloud phase, cloud height) into binary files**
        - ✓ **Call get_remap_binary.pl**
        - ❖ **Execute get_data_nc.exe and ms2gt**
        - o **Combine the binary files into one PS NetCDF file**
        - ✓ **Call convert_remap_bin2nc.pl**
        - ❖ **Execute combine_remap_binary.exe**
        - o **Create log file for NVPWPS_Preproc_Remap.pl**
        - o **Generate PSF file for NVPWPS_Preproc_Remap.pl**
    - **Create remap PSF file**
    - **Update log file VPW_REMAP.pl.log**

VPW Unit Tests
Polar Re-Projection (unit 3)

- **Validation: Polar Re-Projection**
  - **Create directory $working/remap**

```
/data/data046/working/remap
```

  - **Generate remap PCF file**
    - **Remap PCF file location and name**

```
/data/data046/working/remap/NVPWPS_Preproc_Remap.pl.PCF
```

    - **Remap PCF File Content**

```
OPS_CONFIG_FILE=/data/data046/working/VPW_REMAP.pl.PCF
LIST_FILE=/data/data046/working/GRANULE_LIST/granule_list
REGION=NORTH
REMAP_DATA_DIR=/data/data046/working/OUT/npp_viirs_remap
PSF_Clouds=/data/data046/working/clouds/NVPWPS_Preproc_Clouds.pl.PSF
Time_Over_Pole=/data/data046/working/remap/time_over_pole.txt:
```

VPW Unit Tests
Polar Re-Projection (unit 3)

- **Validation: Polar Re-Projection**
  - » **Create PS projection NetCDF file**
    - **Generate binary file**
      - **Binary file location**

/data/data046/working/remap/remap_working

      - **Binary file name**

```
-rw-r--r-- 1 nobody nobody 147456000 Mar 11 10:41 all_data_1
-rw-r--r-- 1 nobody nobody 147456000 Mar 11 10:41 all_data_2
-rw-r--r-- 1 nobody nobody 147456000 Mar 11 10:41 all_data_3
-rw-r--r-- 1 nobody nobody 147456000 Mar 11 10:41 all_data_4
-rw-r--r-- 1 nobody nobody 147456000 Mar 11 10:41 all_data_5
-rw-r--r-- 1 nobody nobody 147456000 Mar 11 10:41 all_data_6
-rw-r--r-- 1 nobody nobody 147456000 Mar 11 10:41 all_data_7
-rw-r--r-- 1 nobody nobody 147456000 Mar 11 10:41 all_lats
-rw-r--r-- 1 nobody nobody 147456000 Mar 11 10:41 all_lons
-rw-r--r-- 1 nobody nobody      1385 Mar 11 10:42 fcsize.txt
-rw-r--r-- 1 nobody nobody       302 Jun 25  2012 myNpse.mpp
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:42 nps_VIIRS_2013122_0448_49_BT__0_3305_3305_f4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:42 nps_VIIRS_2013122_0448_49_BT__3_3305_3305_f4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:41 nps_VIIRS_2013122_0448_49_CldHgtFlag##0_3305_3305_s4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:41 nps_VIIRS_2013122_0448_49_CldHgtFlag##1_3305_3305_s4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:41 nps_VIIRS_2013122_0448_49_CldHgtFlag##2_3305_3305_s4.bin
```

VPW Unit Tests
Polar Re-Projection (unit 3)

- **Validation: Polar Re-Projection**
  - » **Create PS projection NetCDF file**
    - **Generate binary file**
      - **Binary file name (cont.)**

```
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:41 nps_VIIRS_2013122_0448_49_CldTopHght_3305_3305_f4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:41 nps_VIIRS_2013122_0448_49_CldTopPres_3305_3305_f4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:41 nps_VIIRS_2013122_0448_49_CldTopTemp_3305_3305_f4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:41 nps_VIIRS_2013122_0448_49_CloudHgtQF_3305_3305_s4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:40 nps_VIIRS_2013122_0448_49_CloudMask_3305_3305_s4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:40 nps_VIIRS_2013122_0448_49_CloudPhase_3305_3305_s4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:40 nps_VIIRS_2013122_0448_49_CloudType_3305_3305_s4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:41 nps_VIIRS_2013122_0448_49_InverFlag_3305_3305_s4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:41 nps_VIIRS_2013122_0448_49_PcError_3305_3305_f4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:42 nps_VIIRS_2013122_0448_49_Radiance__0_3305_3305_f4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:42 nps_VIIRS_2013122_0448_49_Radiance__3_3305_3305_f4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:42 nps_VIIRS_2013122_0448_49_ReflPrct__4_3305_3305_f4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:42 nps_VIIRS_2013122_0448_49_SolarZenith_3305_3305_f4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:42 nps_VIIRS_2013122_0448_49_SpaceMask_3305_3305_s4.bin
-rw-r--r-- 1 nobody nobody  43692100 Mar 11 10:41 nps_VIIRS_2013122_0448_49_TcError_3305_3305_f4.bin
```

      - **Log file location and name**

```
/data/data046/working/remap/get_remap_binary.pl.log
```

VPW Unit Tests
Polar Re-Projection (unit 3)

- **Validation: Polar Re-Projection**
  - » **Create PS projection NetCDF file**
    - **Generate binary file**
      - **Log file content**

```
Starting at Tue Mar 11 10:40:21 EDT 2014
==============================
OPS_CONFIG_FILE = /data/data046/mfan/working/VPW_REMAP.pl.PCF
REGION = NORTH
m_option = -m
var_name = CloudMask
gpd_file = /data/data046/mfan/dlv_2013/VPW/DATA/static_input/remap/myNpseh.gpd
REGION = NORTH
Variable = CloudMask
 data file: /data/data046/mfan/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0438_52_NPP_VIIRS_CLOUD_MASK.nc
HDF5-DIAG: Error detected in HDF5 (1.8.10) thread 0:
  #000: H5D.c line 334 in H5Dopen2(): not found
    major: Dataset
    minor: Object not found
  #001: H5Gloc.c line 430 in H5G_loc_find(): can't find object
    major: Symbol table
    minor: Object not found
  #002: H5Gtraverse.c line 861 in H5G_traverse(): internal path traversal failed
    major: Symbol table
    minor: Object not found
  #003: H5Gtraverse.c line 641 in H5G_traverse_real(): traversal operator failed
    major: Symbol table
    minor: Callback failed
  #004: H5Gloc.c line 385 in H5G_loc_find_cb(): object 'CloudMaskFactors' doesn't exist
    major: Symbol table
    minor: Object not found
 Filename: /data/data046/mfan/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0438_52_NPP_VIIRS_CLOUD_MASK.nc
 var. list: CloudMask


……
```

VPW Unit Tests
Polar Re-Projection (unit 3)

- **Validation: Polar Re-Projection**
  - » **Create PS projection NetCDF file**
    - **Generate binary file**
      - **Log file content (cont.)**

```
var. list: CloudPhase,CloudType
Dim. 0: 768
Dim. 1: 3200
total size = 2457600
 Dataname: CloudPhase
fill value = -128
class = 0
order = 0
sign = 1
size = 1
Dim. 0: 768
Dim. 1: 3200
Integer data type - looking for scale and offset...
no scale/offset found.
Data: 0 0 0
     0
     0
     0
 Dataname: CloudType
fill value = -128
class = 0
order = 0
sign = 1
size = 1
Dim. 0: 768
Dim. 1: 3200
Integer data type - looking for scale and offset...
no scale/offset found.
Data: 0 0 0
     0
     0
     0
OPS_BIN/get_data_nc.exe OK
data file: /data/data046/mfan/working/OUT/viirs_results_clouds/results/NPP_VIIRS_2013122_0441_42_AWG_CLX_CLOUD_PHASE.nc
```

VPW Unit Tests
Polar Re-Projection (unit 3)

- **Validation: Polar Re-Projection**
  - » **Create PS projection NetCDF file**
    - **Generate binary file**
      - **Log file content (cont.)**

```
/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/bin/ms2gt0.24/bin/fornav 1 -v -t f4 -m -f -999 -D 1.0 3200 720 16 ll2cr_cols_03200_00720_00000_16.img ll2cr_rows_03200_00720_00000_16.img
all_data_7 3305 3305 nps_VIIRS_2013122_0448_49_ReflPrct__4_3305_3305_f4.bin
fornav: $Header: /disks/megadune/data/tharan/ms2gth/src/fornav/fornav.c,v 1.32 2012/07/24 21:42:17 tharan Exp $
 chan_count         = 1
 swath_cols         = 3200
 swath_scans        = 720
 swath_rows_per_scan = 16
 swath_col_file     = ll2cr_cols_03200_00720_00000_16.img
 swath_row_file     = ll2cr_rows_03200_00720_00000_16.img
 swath_chan_file[0]  = all_data_7
 grid_cols          = 3305
 grid_rows          = 3305
 grid_chan_file[0]   = nps_VIIRS_2013122_0448_49_ReflPrct__4_3305_3305_f4.bin

 maximum_weight_mode = 1
 chan_scan_first    = 0
 colrow_scan_first   = 0
 grid_col_start     = 0
 grid_row_start     = 0
 swath_data_type[0]  = f4
 grid_data_type[0]   = f4
 swath_fill[0]      = -999.000000
 grid_fill[0]       = -999.000000

 col_row_min        = 0.000000e+00
 weight_count       = 10000
 weight_min         = 0.010000
 weight_distance_max = 1.000000
 weight_delta_max    = 1.000000
 weight_sum_min     = 0.010000
 swath_col_offset    = 0.000000
 swath_row_offset    = 0.000000

fill count[0]: 6791692
wrote file: nps_VIIRS_2013122_0448_49_ReflPrct__4_3305_3305_f4.bin
```

VPW Unit Tests
Polar Re-Projection (unit 3)

- **Validation: Polar Re-Projection**
  - **Create PS projection NetCDF file**
    - **Combine binary files into one PS NetCDF file**
      - **PS NetCDF file location**

```
/data/data046/working/OUT/npp_viirs_remap
```

      - **PS NetCDF file name**

```
-rw-r--r-- 1 nobody nobody 841077844 Mar 11 10:42 nps_VIIRS_2013122_0448_49_ALL_3305_3305.nc
```

VPW Unit Tests
Polar Re-Projection (unit 3)

- **Validation: Polar Re-Projection**
  - **Create PS projection NetCDF file**
    - **Combine binary files into one PS NetCDF file**
      - **Log file location and name**

```
/data/data046/working/remap/convert_remap_bin2nc.pl.log
```

      - **Log file content**

```
Starting at Tue Oct 15 11:23:50 EDT 2013
==============================
running /data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/bin/combine_remap_binary.exe < fcsize.txt  OK
```

VPW Unit Tests
Polar Re-Projection (unit 3)

- **Validation: Polar Re-Projection**
  - » **Create PS projection NetCDF file**
    - **Log file location and name**

```
/data/data046/working/remap/NVPWPS_Preproc_Remap.pl.log
```

    - **Log file content**

```
NVPWPS_Preproc_Remap.pl is now starting at 20131015-11:19:36
Current Granule List File is: /data/data046/working/GRANULE_LIST/granule_list
  Orbit Number = 07829
  First Granule Start Time = 20130502_0438
remaping CLOUD_MASK ...
-- variables: CloudMask
running get_remap_binary.pl for CLOUD_MASK:  OK
remaping CLOUD_PHASE ...
-- variables: CloudPhase,CloudType
running get_remap_binary.pl for CLOUD_PHASE:  OK
remaping CLOUD_HEIGHT ...
-- variables: CldTopPres,CldTopTemp,CldTopHght,InverFlag,PcError,TcError,CldHgtFlag##0,CldHgtFlag##1,CldHgtFlag##2,CloudHgtQF
running get_remap_binary.pl for CLOUD_HEIGHT:  OK
remaping SAT DATA ...
-- variables: SolarZenith,SpaceMask,BT__0,BT__3,Radiance__0,Radiance__3,ReflPrct__4
running get_remap_binary.pl for SAT DATA:  OK
Pole passover time for this orbit is: 2013122_0448_49

running convert_remap_bin2nc.pl: OK
NVPWPS_Preproc_Remap.pl is now completing successfully at: 20131015-11:23:58
```

VPW Unit Tests
Polar Re-Projection (unit 3)

- **Validation: Polar Re-Projection**
  - » **Create PS projection NetCDF file**
    - **PSF file for NVPWPS_Preproc_Remap.pl**
      - **PS NetCDF file location**

```
/data/data046/working/remap/NVPWPS_Preproc_Remap.pl.PSF
```

      - **PS NetCDF file name**

```
-rw-r--r-- 1 nobody nobody 841077844 Mar 11 10:42 nps_VIIRS_2013122_0448_49_ALL_3305_3305.nc
```

VPW Unit Tests
Polar Re-Projection (unit 3)

- **Validation: Polar Re-Projection**
  - **Create remap PSF file**
    - **Remap PSF file location and name**

```
/data/data046/working/VPW_REMAP.pl.PSF
```

    - **Remap PSF file content**

```
/data/data046/working/OUT/npp_viirs_remap/nps_VIIRS_2013122_0448_49_ALL_3305_3305.nc
```

VPW Unit Tests
Polar Re-Projection (unit 3)

- **Validation: Polar Re-Projection**
  - **Update log file VPW_REMAP.pl.log**
    - **Log file location and name**

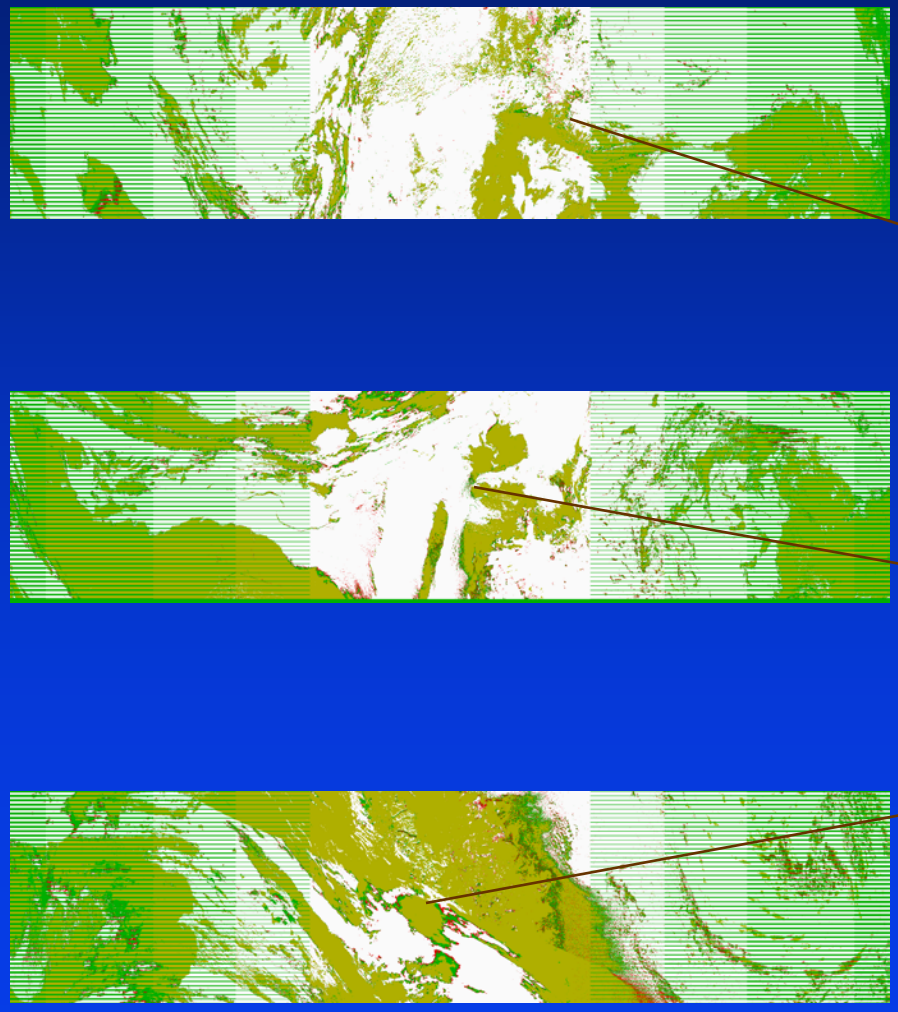
```
/data/data046/working/VPW_REMAP.pl.log
```

    - **Log file content**

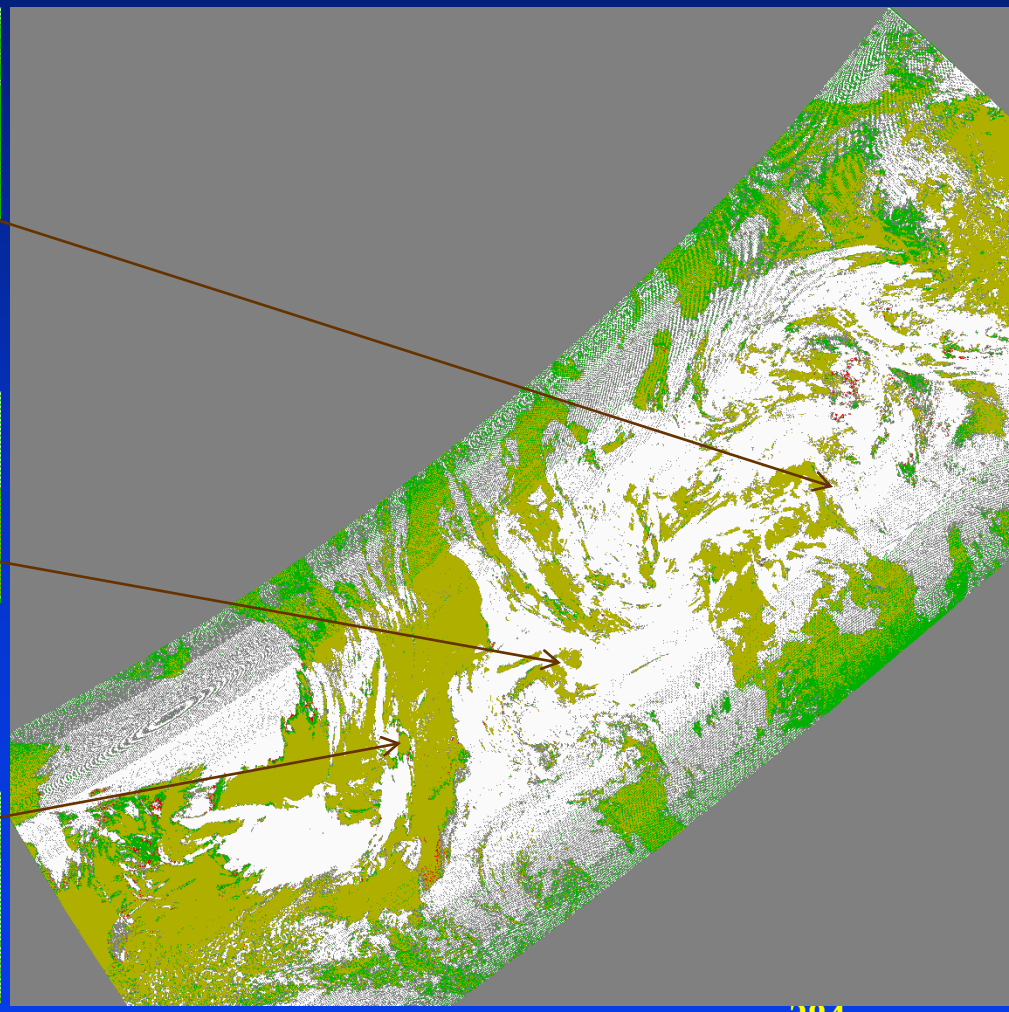
```
 Remapping
   --------------
   Creating Remapping working directory: /data/data046/working/remap
   Creating PCF file: /data/data046/working/remap/NVPWPS_Preproc_Remap.pl.PCF
   Running NVPWPS_Preproc_Remap.pl
    - /data/data046/working/remap/NVPWPS_Preproc_Remap.pl.log
       - sublog:  /data/data046/working/remap/get_remap_binary.pl.log
       -          /data/data046/working/remap/convert_remap_bin2nc.pl.log

real    4m22.202s
user    1m45.091s
sys     0m29.261s
   running NVPWPS_Preproc_Remap.pl: OK
```

VPW Unit Tests
Polar Re-Projection (unit 3)



Granule Cloud Masks



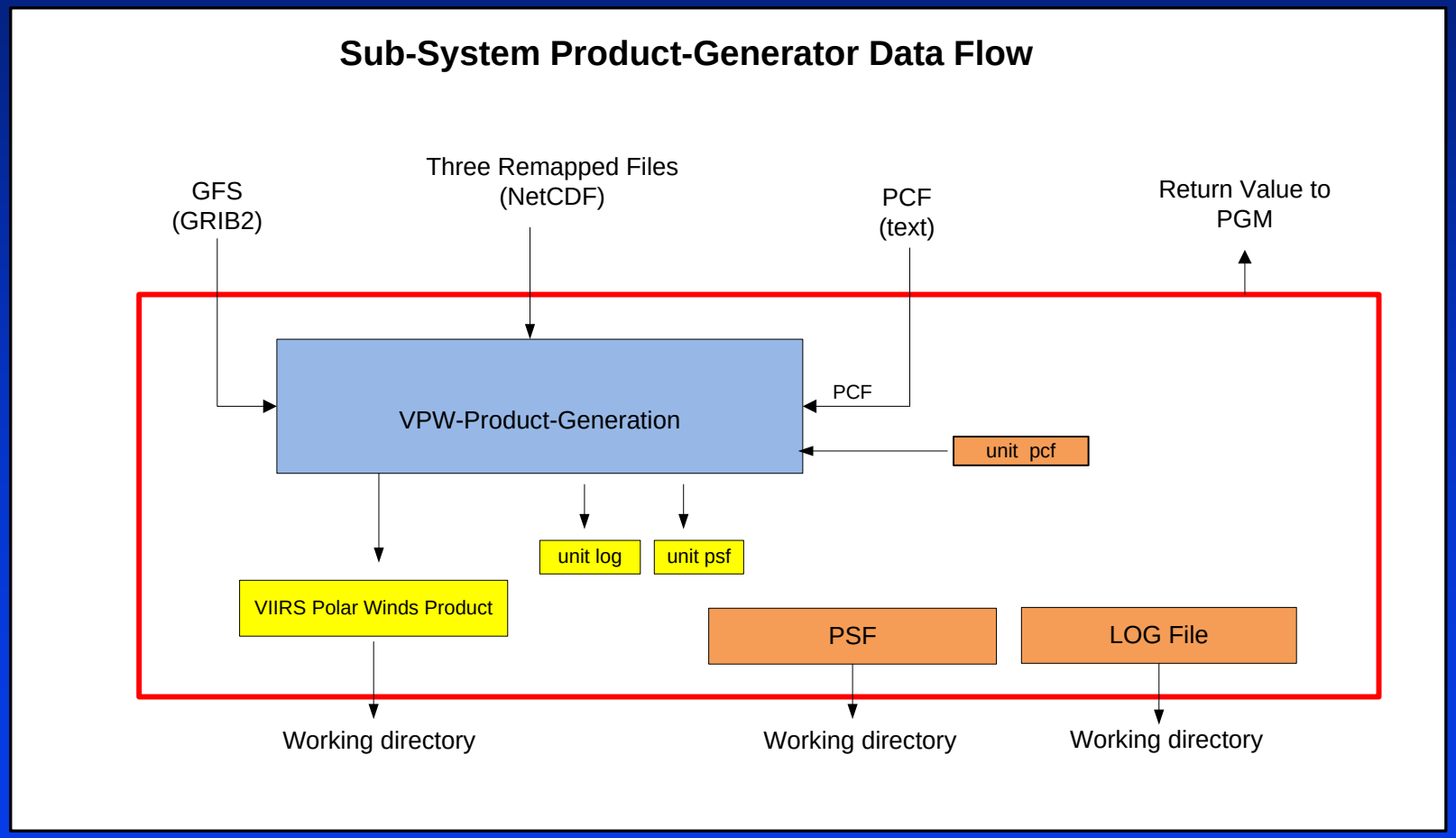
Remapped Cloud Mask

VPW Unit Tests

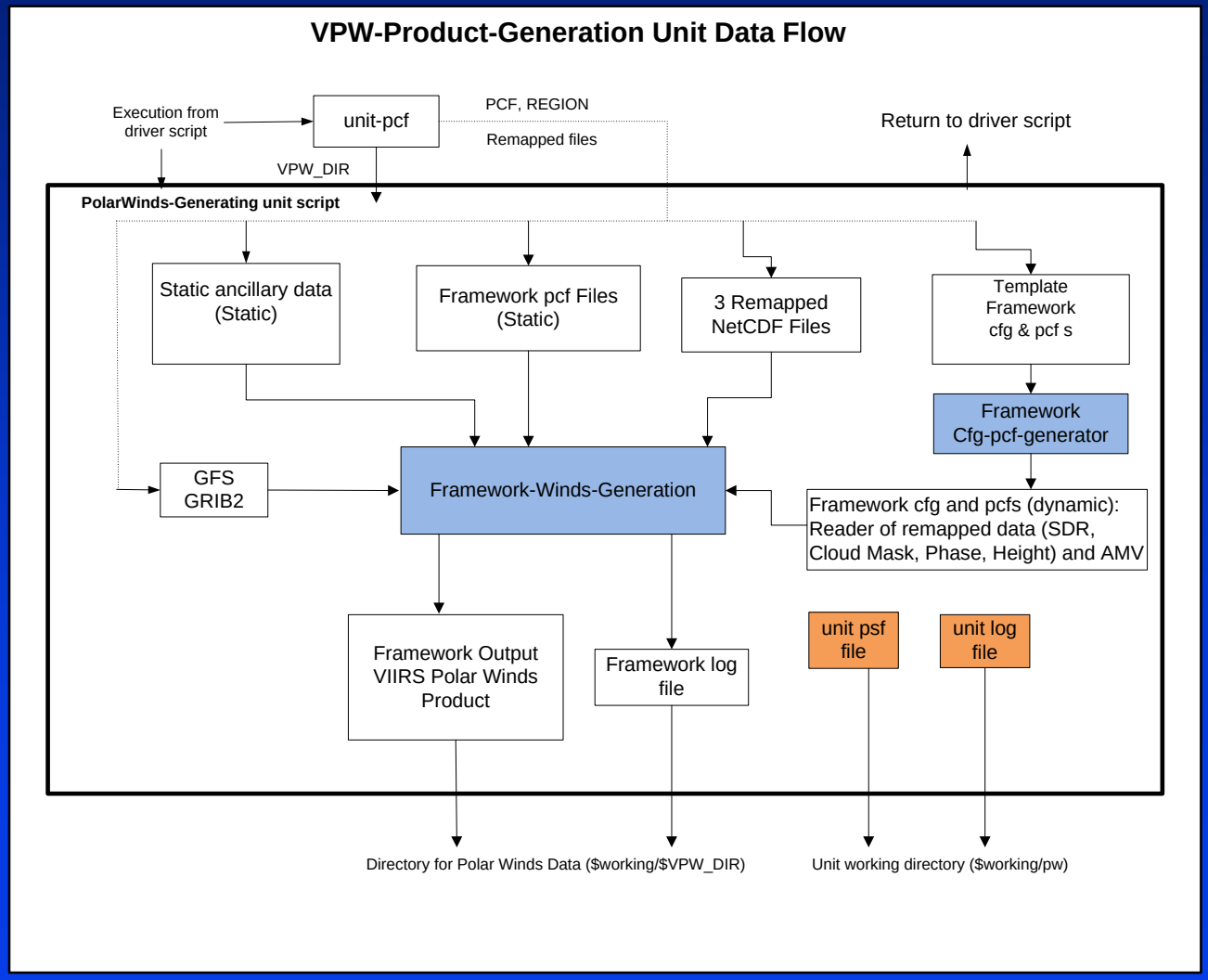
Sub-system:

Product Generator

VPWPS
Sub-System: Product Generator



Product Generator
VPW Product Generation Unit



VPW Unit Tests

System Driver Script:

VPW_PRODUCT.pl

Usage:

VPW_PRODUCT.pl *<System PCF>*

VPW Unit Tests

- **Process Configuration File VPW_PRODUCT.pl.PCF**
  - **This file need to be created for each run and it should exist in the 'working' directory**
  - **This file is parsed by the scripts to set up processing environments.**
  - **PCF Content (an example)**

```
job_coverage_start=201305020126490
job_coverage_end=201305020448490

REGION=NORTH

SCRIPT_FOR_GFS=run_wgrib.pl
PERL_LOC=/usr/bin/perl
OPS_SCRIPT=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/scripts
OPS_BIN=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/bin
MS2GT_DIR=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/bin/ms2gt0.24/bin

CFG_TEMPL_DIR=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/config_templates
PCF_TEMPL_DIR=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/PCF_Overwrites
PCF_OVERWRITES=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/PCF_Overwrites
```

VPW Unit Tests

- **Process Configuration File VPW_PRODUCT.pl.PCF**
  - » **PCF Content (cont.)**

```
CFG_TEMPL_FILE_FOR_CLOUDS=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/config_templates/AWG_NPP_VIIRS_Clouds.cfg
CFG_TEMPL_FILE_FOR_POLAR_WINDS=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/config_templates/polarWind_nps_VIIRS_2012149_1231_48.cfg
PCF_TEMPL_FILE_FOR_VIIRS_SDR_Data=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/PCF_Overwrites/NPP_VIIRS_SDR_Data.pcf
PCF_TEMPL_FILE_FOR_VIIRS_Cloud_Mask=/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/PCF_Overwrites/NPP_VIIRS_Cloud_Mask.pcf

REMAP_GPD_FILE_SP=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/remap/mySpseh.gpd
REMAP_GPD_FILE_NP=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/remap/myNpseh.gpd
REMAP_MPP_FILE_SP=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/remap/mySpse.mpp
REMAP_MPP_FILE_NP=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/remap/myNpse.mpp
CRTM_VIIRS_M_SpcCoeff=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/viirs-
m_npp.SpcCoeff.bin
CRTM_VIIRS_M_TauCoeff=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/viirs-
m_npp.TauCoeff.bin
CRTM_CloudCoeff=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/CloudCoeff.bin
CRTM_AerosolCoeff=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/AerosolCoeff.bin
CRTM_EmisCoeff=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/EmisCoeff.bin
```

VPW Unit Tests

- **Process Configuration File VPW_PRODUCT.pl.PCF**
  - » **PCF Content (cont.)**

```
AWG_NOP_AMV_COEF=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/AWG_WINDS_AMV/AMV_coeff.nc
NPP_VIIRS_SDR_ANCIL=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/npp_viirs_ancil.nc
POLAR_MAP_ANCIL=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/npp_viirs_ancil.nc
SPS_LATLON_ANCIL=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/mySpseh_Lats.Lons.3305x3305x1.nc
NPS_LATLON_ANCIL=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/myNpseh_Lats.Lons.3305x3305x1.nc
NPP_VIIRS_CLX_CLDHGT_ANCIL=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/pol_winds/polar_winds_ancil.nc
COAST_MASK_NASA_1KM=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/masks/coast_mask_1km.nc
SFC_TYPE_AVHRR_1KM=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/masks/gl-latlong-1km-landcover.nc
LAND_MASK_NASA_1KM=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/masks/lw_geo_2001001_v03m.nc
SFC_ELEV_GLOBE_1KM=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/sfc_elevation/GLOBE_1km_digelev.nc
F1_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005001.nc
F2_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005032.nc
F3_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005060.nc
F4_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005091.nc
F5_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005121.nc
F6_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005152.nc
F7_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005182.nc
F8_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005213.nc
F9_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005244.nc
F10_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005274.nc
F11_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005305.nc
F12_SEEBOR_EMISS=/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005335.nc
```

VPW Unit Tests

- **Process Configuration File VPW_PRODUCT.pl.PCF**
  - » **PCF Content (cont.)**

```
OPS_LIST_DIR=GRANULE_LIST
HDF_DATA=OUT/viirs_hdf5
SDR_DATA=OUT/npp_viirs_sdr
VIIRS_REMAP_DATA_DIR=OUT/npp_viirs_remap
REMAP_WORKING=remap_working

ALGORITHM_ANCILLARY_DIR=algorithm_ancillary
FRAMEWORK_ANCILLARY_DIR=framework_ancillary

CLOUD_DATA=OUT/viirs_results_clouds
VPW_OUTPUT=OUT/viirs_results_pw

REMAP_FILE_1=nps_VIIRS_2013122_0126_49_ALL_3305_3305.nc
REMAP_FILE_2=nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc
REMAP_FILE_3=nps_VIIRS_2013122_0448_49_ALL_3305_3305.nc

gfs_file=gfs_data/gfs.t00z.pgrb2f03.20130502
gfs_file=gfs_data/gfs.t00z.pgrb2f06.20130502
```

VPW Unit Tests
Running Environment

- Setup Running Environment
  - » Purpose: Extract variables from system PCF file VPW_PRODUCT.pl.PCF and setup environment for polar winds
  - » Major Steps
    - Parse system PCF file
    - Create necessary directories if they do not exist
    - Make links to framework
    - Make links to static ancillary data
      - Call Setup_links_for_static_ancillary.pl
    - Make links to dynamic and ancillary data

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - **Create $working/pw**

```
/data/data046/working/pw
```

  - **Make links to framework**

```
lrwxrwxrwx 1 nobody nobody    46 Mar  4 13:23 algorithm_ancillary -> /data/data046/mfan/working/algorithm_ancillary
lrwxrwxrwx 1 nobody nobody    45 Mar  4 13:23 CustomProductLists -> /data/data046/mfan/working/CustomProductLists
lrwxrwxrwx 1 nobody nobody    38 Mar  4 13:23 Default_PCF -> /data/data046/mfan/working/Default_PCF
lrwxrwxrwx 1 nobody nobody    46 Mar  4 13:23 framework_ancillary -> /data/data046/mfan/working/framework_ancillary
lrwxrwxrwx 1 nobody nobody    41 Mar  4 13:23 PCF_Overwrites -> /data/data046/mfan/working/PCF_Overwrites
lrwxrwxrwx 1 nobody nobody    39 Mar  4 13:23 run_wgrib.pl -> /data/data046/mfan/working/run_wgrib.pl
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Make links to static ancillary data**
    - **CRTM Coefficient**
      - **Location**

/data/data046/working/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 126 Jan 17 12:33 AerosolCoeff.bin ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/AerosolCoeff.bin
lrwxrwxrwx 1 nobody nobody 124 Jan 17 12:33 CloudCoeff.bin ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/CloudCoeff.bin
lrwxrwxrwx 1 nobody nobody 123 Jan 17 12:33 EmisCoeff.bin ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/EmisCoeff.bin
lrwxrwxrwx 1 nobody nobody 134 Jan 17 12:33 viirs-m_npp.SpcCoeff.bin ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/viirs-m_npp.SpcCoeff.bin
lrwxrwxrwx 1 nobody nobody 134 Jan 17 12:33 viirs-m_npp.TauCoeff.bin ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/viirs-m_npp.TauCoeff.bin
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Make links to static ancillary data**
    - **Wind Coefficient**
      - **Location**

```
/data/data046/working/algorithm_ancillary/AWG_WINS_AMV
```

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 96 Jan 17 12:33 AMV_coeff.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/algorithm_ancillary/AWG_WINDS_AMV/AMV_coeff.nc
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Make links to static ancillary data**
    - **Framework ancillary files**
      - **Location**

```
/data/data046/working/framework_ancillary/ancil_file
```

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 myNpseh_Lats.Lons.3305x3305x1.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/myNpseh_Lats.Lons.3305x3305x1.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 mySpseh_Lats.Lons.3305x3305x1.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/mySpseh_Lats.Lons.3305x3305x1.nc
lrwxrwxrwx 1 nobody nobody  99 Jan 17 12:33 npp_viirs_ancil.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/ancil_file/npp_viirs_ancil.nc
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - **Make links to static ancillary data**
    - **Polar winds**
      - **Location**

```
/data/data046/working/framework_ancillary/pol_winds
```

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 100 Jan 17 12:33 polar_winds_ancil.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/pol_winds/polar_winds_ancil.nc
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - **Make links to static ancillary data**
    - **Mask files**
      - **Location**

```
/data/data046/working/framework_ancillary/masks
```

      - **Links**

```
lrwxrwxrwx 1 nobody nobody  93 Jan 17 12:33 coast_mask_1km.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/masks/coast_mask_1km.nc
lrwxrwxrwx 1 nobody nobody 103 Jan 17 12:33 gl-latlong-1km-landcover.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/masks/gl-latlong-1km-landcover.nc
lrwxrwxrwx 1 nobody nobody  98 Jan 17 12:33 lw_geo_2001001_v03m.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/masks/lw_geo_2001001_v03m.nc
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - **Make links to static ancillary data**
    - **Surface elevation**
      - **Location**

```
/data/data046/working/framework_ancillary/sfc_elevation
```

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 104 Jan 17 12:33 GLOBE_1km_digelev.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/sfc_elevation/GLOBE_1km_digelev.nc
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - **Make links to static ancillary data**
    - **Seebor emissivity**
      - **Location**

/data/data046/working/framework_ancillary/seebor_emiss

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005001.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005001.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005032.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005032.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005060.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005060.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005091.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005091.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005121.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005121.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005152.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005152.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005182.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005182.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005213.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005213.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005244.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005244.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005274.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005274.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005305.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005305.nc
lrwxrwxrwx 1 nobody nobody 113 Jan 17 12:33 global_emiss_intABI_2005335.nc ->
/data/data046/mfan/dlv_2013/VPW/DATA/static_input/framework_ancillary/seebor_emiss/global_emiss_intABI_2005335.nc
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Make links to static ancillary data**
    - **Log file location and name**

/data/data046/working/Setup_links_for_static_ancillary.pl.log

- **Log file content**

```
ln: creating symbolic link `algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/./viirs-m_npp.SpcCoeff.bin': File exists
ln: creating symbolic link `algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/./viirs-m_npp.TauCoeff.bin': File exists
ln: creating symbolic link `algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/./CloudCoeff.bin': File exists
ln: creating symbolic link `algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/./AerosolCoeff.bin': File exists
ln: creating symbolic link `algorithm_ancillary/crtm_2.0.2/CRTM_Coefficients/Big_Endian/./EmisCoeff.bin': File exists
ln: creating symbolic link `algorithm_ancillary/AWG_WINDS_AMV/./AMV_coeff.nc': File exists
ln: creating symbolic link `framework_ancillary/ancil_file/./npp_viirs_ancil.nc': File exists
ln: creating symbolic link `framework_ancillary/ancil_file/./aqua_modis_abi_pol_ancil.nc': File exists
ln: creating symbolic link `framework_ancillary/ancil_file/./mySpseh_Lats.Lons.3305x3305x1.nc': File exists
ln: creating symbolic link `framework_ancillary/ancil_file/./myNpseh_Lats.Lons.3305x3305x1.nc': File exists
ln: creating symbolic link `framework_ancillary/pol_winds/./polar_winds_ancil.nc': File exists
ln: creating symbolic link `framework_ancillary/masks/./coast_mask_1km.nc': File exists
ln: creating symbolic link `framework_ancillary/masks/./gl-latlong-1km-landcover.nc': File exists
ln: creating symbolic link `framework_ancillary/masks/./lw_geo_2001001_v03m.nc': File exists
ln: creating symbolic link `framework_ancillary/sfc_elevation/./GLOBE_1km_digelev.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005001.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005032.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005060.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005091.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005121.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005152.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005182.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005213.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005244.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005274.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005305.nc': File exists
ln: creating symbolic link `framework_ancillary/seebor_emiss/./global_emiss_intABI_2005335.nc': File exists
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - » **Make links to dynamic ancillary data**
    - **GFS GRIB2**
      - **Location**

/data/data046/working/framework_ancillary/gfs_grib2_0.5deg/2013/05/02

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t00z.pgrbf03 -> /data/data046/mfan/working/gfs_data/gfs.t00z.pgrb2f03.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t00z.pgrbf06 -> /data/data046/mfan/working/gfs_data/gfs.t00z.pgrb2f06.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t00z.pgrbf09 -> /data/data046/mfan/working/gfs_data/gfs.t00z.pgrb2f09.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t06z.pgrbf03 -> /data/data046/mfan/working/gfs_data/gfs.t06z.pgrb2f03.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t06z.pgrbf06 -> /data/data046/mfan/working/gfs_data/gfs.t06z.pgrb2f06.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t06z.pgrbf09 -> /data/data046/mfan/working/gfs_data/gfs.t06z.pgrb2f09.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t12z.pgrbf03 -> /data/data046/mfan/working/gfs_data/gfs.t12z.pgrb2f03.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t12z.pgrbf06 -> /data/data046/mfan/working/gfs_data/gfs.t12z.pgrb2f06.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t12z.pgrbf09 -> /data/data046/mfan/working/gfs_data/gfs.t12z.pgrb2f09.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t18z.pgrbf03 -> /data/data046/mfan/working/gfs_data/gfs.t18z.pgrb2f03.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t18z.pgrbf06 -> /data/data046/mfan/working/gfs_data/gfs.t18z.pgrb2f06.20130502
lrwxrwxrwx 1 nobody nobody 62 Jan 17 12:33 gfs.t18z.pgrbf09 -> /data/data046/mfan/working/gfs_data/gfs.t18z.pgrb2f09.20130502
```

VPW Unit Tests
Running Environment

- **Validation: Setup Running Environment**
  - **Make links to dynamic ancillary data**
    - **OISST**
      - **Location**

```
/data/data046/working/framework_ancillary/oisst_daily
```

      - **Links**

```
lrwxrwxrwx 1 nobody nobody 75 Jan 17 12:33 avhrr-only-v2.20130501.nc -> /data/data046/mfan/working/oisst_data/avhrr-only-v2.20130501_preliminary.nc
lrwxrwxrwx 1 nobody nobody 75 Jan 17 12:33 avhrr-only-v2.20130502.nc -> /data/data046/mfan/working/oisst_data/avhrr-only-v2.20130502_preliminary.nc
```

VPW Unit Tests
VPW Product Generation

- VPW Product Generation
  - » Purpose: Generate polar wind product from the remapped files
  - » Major Steps
    - Create polar wind PCF file NVPWPS_Proc_PW.pl.PCF
    - Generate polar wind product
      - Call NVPWPS_Proc_PW.pl
        - o Create polar wind CFG file
        - o Create polar wind overwritten PCF files
        - o Execute pcf_framework.exe to create polar wind product
        - o Generate log file for NVPWPS_Proc_PW.pl
        - o Create PSF for NVPWPS_Proc_PW.pl
    - Create polar winds PSF file
    - Generate log file for VPM_PRODUCT.pl

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - » **Create polar winds PCF file**
    - **Polar winds PCF location and name**

```
/data/data046/working/pw/NVPWPS_Proc_PW.pl.PCF
```

    - **Polar winds PCF content**

```
OPS_CONFIG_FILE=/data/data046/mfan/working/VPW_PRODUCT.pl.PCF
REGION=NORTH
VPW_DIR=/data/data046/mfan/working/OUT/viirs_results_pw
PSF_Remap_before=/data/data046/mfan/working/psf_remap_1
PSF_Remap_current=/data/data046/mfan/working/psf_remap_2
PSF_Remap_after=/data/data046/mfan/working/psf_remap_3
```

VPW Unit Tests

VPW Product Generation

- **Validation: VPW Product Generation**
  - **Generate polar wind product**
    - **Create polar winds CFG file**
      - **CFG file location and name**

/data/data046/working/OUT/viirs_results_pw/configs/NH_poleG_2013122_0307_49.cfg

      - **CFG file content**

```
# Satellite info
SATELLITE: AQUA
INSTRUMENT: MODIS

#
# Data dimensions
#
ROWS: 3305
COLUMNS: 3305
CHANNELS: 39

#
# PCF Framework Settings
#

# Segment details
#
ROW_START: 1
ROW_END: 3305
COLUMN_START: 1
COLUMN_END: 3305
SEGMENT_SIZE: 300
```

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - » **Generate polar wind product**
    - **Create polar winds CFG file**
      - **CFG file content (cont.)**

```
#
# RUN_MODE - USER_DEFINED_PCF or ALLOW_DEFAULT_PCF
# USER_DEFINED forces user to specify each product in this file
# instead of relying on dependencies to determine if that product should run
# and using the default pcf for that product
#
RUN_MODE: ALLOW_DEFAULT_PCF

DEPENDENCY_INFO_LOCATION: ALLOW_IN_MAIN

#
# Output Directory
#
OUTPUT_DIRECTORY: /data/data046/working/OUT/viirs_results_pw/results/

# Output Option
# NETCDF_SEGMENT - creates a NetCDF file for each segment
# NETCDF_ALL - creates a single file for the full run
# Important to note, each product (team) has its own output file
# Specifying the ALL part will not create a single file with everything
# in it.
#
OUTPUT_FILE_OPTION: NETCDF_ALL
FILE_VERSION: NETCDF_V4
```

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - » **Generate polar wind product**
    - **Create polar winds CFG file**
      - **CFG file content (cont.)**

```
#
# For satellite data, specify actual data file instead of pcf
#
SATELLITE_DATA:
AQUA_MODIS_POLAR = /data/data046/working/nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc + PCF_Overwrites/AQUA_MODIS_POLAR.pcf

LAND_MASK:
LAND_MASK_NASA_1KM = PCF_Overwrites/Land_Mask.pcf

NWP_DATA:
NWP_GFS = PCF_Overwrites/NWP_Data_grib2_0.5deg.pcf

CLOUD_MASK:
MDS_POLAR_CLD_MASK = /data/data046/working/OUT/viirs_results_pw/configs/MDS_POLAR_Cloud_Mask_2013122_0307_49.pcf

CLOUD_PHASE:
MDS_POLAR_CLD_PHASE = /data/data046/working/OUT/viirs_results_pw/configs/MDS_POLAR_Cloud_Phase_2013122_0307_49.pcf
CLOUD_HEIGHT:
MDS_POLAR_CLD_HEIGHT = /data/data046/working/OUT/viirs_results_pw/configs/MDS_POLAR_Cloud_Height_2013122_0307_49.pcf

DERIVED_MOTION_WINDS:
AWG_NOP_AMV = /data/data046/working/OUT/viirs_results_pw/configs/AWG_NOP_AMV_2013122_0307_49.pcf
```

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - » **Generate polar wind product**
    - **Create polar wind overwritten pcf files**
      - **Overwritten pcf file location**

```
/data/data046/working/OUT/viirs_results_pw/configs
```

      - **Overwritten pcf file name**

```
-rw-r--r-- 1 nobody nobody 1982 Mar  4 13:23 AWG_NOP_AMV_2013122_0307_49.pcf
-rw-r--r-- 1 nobody nobody  741 Mar  4 13:23 MDS_POLAR_Cloud_Height_2013122_0307_49.pcf
-rw-r--r-- 1 nobody nobody 1170 Mar  4 13:23 MDS_POLAR_Cloud_Mask_2013122_0307_49.pcf
-rw-r--r-- 1 nobody nobody  804 Mar  4 13:23 MDS_POLAR_Cloud_Phase_2013122_0307_49.pcf
```

VPW Unit Tests

VPW Product Generation

- **Validation: VPW Product Generation**
  - **Generate polar wind product**
    - **Create polar wind overwritten PCF files**
      - **Overwritten PCF file: AWG_NOP_AMV_2013122_0307_49.pcf**

```
# AWG_NOP_AMV
#
AWG_NOP_AMV
PRODUCT

WRITER: DO_NOT_OVERWRITE

#
# Set flag for output
#
OUTPUT_TO_FILE: Y

##########################################
# This section needed for Default PCF only #
##########################################

#
# Structure information used for code generation
#

#
# For GOESR either Framework_Input_Structure or Framework_Output_Structure
#
MAIN_STRUCTURE_TYPE: Framework_Output_Structure
```

VPW Unit Tests

VPW Product Generation

- **Validation: VPW Product Generation**
  - » **Generate polar wind product**
    - **Create polar wind overwritten PCF files**
      - **Overwritten PCF file: AWG_NOP_AMV_2013122_0307_49.pcf (cont.)**

```
# Algorithm structure type
#
ALGORITHM_STRUCTURE_TYPE: PolWnd_AMV_Structure

#
# Algorithm structure name (long)
#
# Output example: AWG_CLOUD_MASK
# ALGORITHM_STRUCTURE_NAME: AWG_Cloud_Mask
#
# Input example: SFC_EMISS_SEEBOR
# ALGORITHM_STRUCTURE_NAME: GOESR_ABI_Data.Ancil_2km.SfcEmiss_Seebor
#
#
ALGORITHM_STRUCTURE_NAME: AWG_PolWnd_AMV

################################
# End Default PCF only Section #
################################


#
# Dependencies
#
DEPENDENCIES:
```

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - » **Generate polar wind product**
    - **Create polar wind overwritten PCF files**
      - **Overwritten PCF file: AWG_NOP_AMV_2013122_0307_49.pcf (cont.)**

```
SATELLITE_DATA:

LAND_MASK:
LAND_MASK_NASA_1KM

NWP_DATA:
NWP_GFS

TEMPORAL_DATA:
#NO_TEMPORAL
TEMPORAL_DATA_FIXED

CLOUD_MASK:
MDS_POLAR_CLD_MASK

CLOUD_PHASE:
MDS_POLAR_CLD_PHASE

CLOUD_HEIGHT:
MDS_POLAR_CLD_HEIGHT
```

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - » **Generate polar wind product**
    - **Create polar wind overwritten PCF files**
      - **Overwritten PCF file: AWG_NOP_AMV_2013122_0307_49.pcf (cont.)**

```
#
# Create this section only if Temporal Data part of dependencies
#
TEMPORAL:

Timestep_Type: AQUA_MODIS
Timestep: +1
Actual_Time: 6060
Load:
AQUA_MODIS_POLAR = /data/data046/working/OUT/npp_viirs_remap/nps_VIIRS_2013122_0448_49_ALL_3305_3305.nc


Timestep_Type: AQUA_MODIS
Timestep: -1
Actual_Time: -6060
Load:
AQUA_MODIS_POLAR =  /data/data046/working/nps_VIIRS_2013122_0126_49_ALL_3305_3305.nc
```

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - » **Generate polar wind product**
    - **Create polar wind overwritten PCF files**
      - **Overwritten PCF file: AWG_NOP_AMV_2013122_0307_49.pcf (cont.)**

```
# Settings
#
OTHER:
AMV_Channel: 14
Time_Interval_Minus: 99
Time_Interval_Plus: 99
Box_Size: 19
Nested_Tracking_Flag: 1
EE_Filter_Flag: 0
CSWV_Flag: 0
Temporal_Cloud_Flag: 1
NavType: PS
NavNPLine: 1653
NavNPElem: 1653
NavIpole: 0
NavStdLat: 600000
NavSpcStdLat: 2000
NavNormLon: 0
NavRadiusEQ: 6378137
NavECCPlanet: 81819
NavLonConvet: 0
ExpectedErr_Coeff_Filename: algorithm_ancillary/AWG_WINDS_AMV/AMV_coeff.nc
PSSubLat: 90.0
PSSubLon: 0.0

END AWG_NOP_AMV
```

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - **Generate polar wind product**
    - **Create polar wind overwritten PCF files**
      - **Overwritten PCF file: MDS_POLAR_Cloud_Height_2013122_0307_49.pcf**

```
MDS_POLAR_CLD_HEIGHT
ANCILLARY

#
# Set flag for output
#
OUTPUT_TO_FILE: Y

DEPENDENCIES:
SATELLITE_DATA:

OTHER:
#Filename: /data/data046/working/nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc
Flag_Constant: 3
Packed_Flag_Constant: 1
#Filename: /data/data046/working/nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc
Filename: /data/data046/working/nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc
#
# Size factor compared to data being used.
# Ex.  5km satellite data size factor would be 1
#      1km satellite data size factor would be 1/5 or .2
#
SizeFactor: 1

END MDS_POLAR_CLD_HEIGHT
```

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - **Generate polar wind product**
    - **Create polar wind overwritten PCF files**
      - **Overwritten PCF file: MDS_POLAR_Cloud_Mask_2013122_0307_49.pcf**

```
#
# MODIS POLAR Cloud Mask
# MXD35 (1KM)
#
MDS_POLAR_CLD_MASK
ANCILLARY

#
# Set flag for output
#
OUTPUT_TO_FILE: Y

##############################################
# This section needed for Default PCF only #
##############################################

#
# Structure information used for code generation
#

# For GOESR either Framework_Input_Structure or Framework_Output_Structure
MAIN_STRUCTURE_TYPE: Framework_Output_Structure

# Algorithm structure type
ALGORITHM_STRUCTURE_TYPE: Cloud_Mask_Structure
```

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - **Generate polar wind product**
    - **Create polar wind overwritten PCF files**
      - **Overwritten PCF file: MDS_POLAR_Cloud_Mask_2013122_0307_49.pcf (cont.)**

```
# Algorithm structure name (long)
#
# Output example: AWG_CLOUD_MASK
# ALGORITHM_STRUCTURE_NAME: AWG_Cloud_Mask
#
# Input example: SFC_EMISS_SEEBOR
# ALGORITHM_STRUCTURE_NAME: GOESR_ABI_Data.Ancil_2km.SfcEmiss_Seebor
#
#
ALGORITHM_STRUCTURE_NAME: MODIS_POLAR_Cloud_Mask

################################
# End Default PCF only Section #
################################
DEPENDENCIES:
SATELLITE_DATA:


OTHER:
#Filename: /data/data046/working/nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc
Filename: /data/data046/working/nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc
CldMask_Packed_Constant: 7
Flag_Constant: 1
Meta10: 10
Meta20: 20
END MDS_POLAR_CLD_MASK
```

VPW Unit Tests

VPW Product Generation

- **Validation: VPW Product Generation**
  - **Generate polar wind product**
    - **Create polar wind overwritten PCF files**
      - **Overwritten PCF file: MDS_POLAR_Cloud_Phase_2013122_0307_49.pcf**

```
MDS_POLAR_CLD_PHASE
ANCILLARY

#
# Set flag for output
#
OUTPUT_TO_FILE: Y

DEPENDENCIES:
SATELLITE_DATA:


OTHER:
#Filename: /data/data046/working/nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc
#Filename: /data/data046/working/nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc
Filename: /data/data046/working/nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc
#
# Size factor compared to data being used.
# Ex.  5km satellite data size factor would be 1
#      1km satellite data size factor would be 1/5 or .2
#
#NOTE: This is now 1KM data!!!! Ignore the misleading name.
SizeFactor: 1
Flag_Constant: 3
Packed_Flag_Constant: 1

END MDS_POLAR_CLD_PHASE
```

VPW Unit Tests

VPW Product Generation

- **Validation: VPW Product Generation**
  - **Generate polar wind product**
    - **Execute pcf_framework.exe**
      - **Polar wind product file location**

/data/data046/working/OUT/viirs_results_pw/results

      - **Polar wind product file name**

PAMV-VIIRS-CD-NH_v1r1_npp_s201305020126490_e201305020448490_c201310151533470.nc

      - **Log file location and name**

/data/data046/working/OUT/viirs_results_pw/logs/NH_poleG_2013122_0307_49.log

      - **Log file content**

```
Opened file: /data/data046/working/OUT/viirs_results_pw/configs/NH_poleG_2013122_0307_49.cfg
Opened file: Default_PCF/Master_List_AQUA_MODIS.pcf
Table size: 68
Final Process List:
AQUA_MODIS_POLAR
LAND_MASK_NASA_1KM
NWP_GFS
MDS_POLAR_CLD_MASK
MDS_POLAR_CLD_PHASE
MDS_POLAR_CLD_HEIGHT
TEMPORAL_DATA_FIXED
AWG_NOP_AMV
Setup list: 8
Run list: 8
Segment Number: 1 of 12
Running: AQUA_MODIS_POLAR
Running: LAND_MASK_NASA_1KM
Running: NWP_GFS
….
```

VPW Unit Tests

VPW Product Generation

- **Validation: VPW Product Generation**
  - **Generate polar wind product**
    - **Execute pcf_framework.exe**
      - **Log file content (cont.)**

```
InputFile1: framework_ancillary/gfs_grib2_0.5deg/2013/05/02/gfs.t00z.pgrbf03
InputFile2: framework_ancillary/gfs_grib2_0.5deg/2013/05/02/gfs.t00z.pgrbf06
Name of NWP temp dump file: 21621
Read_AVN_File: ... loaded: 2013/05/02/gfs.t00z.pgrbf03 in 88 seconds
Name of NWP temp dump file: 21621
Read_AVN_File: ... loaded: 2013/05/02/gfs.t00z.pgrbf06 in 46 seconds

 NWP files used
----------------
 File Name 1 = framework_ancillary/gfs_grib2_0.5deg/2013/05/02/gfs.t00z.pgrbf03
 File Name 2 = framework_ancillary/gfs_grib2_0.5deg/2013/05/02/gfs.t00z.pgrbf06
----------------
Running: MDS_POLAR_CLD_MASK
Running: MDS_POLAR_CLD_PHASE
Running: MDS_POLAR_CLD_HEIGHT
Running: TEMPORAL_DATA_FIXED
Running: AWG_NOP_AMV
 satellite: NPP
 year:         2013
 day:          122
 hour:           3
 minute:           7
 channel to process:          14
Instrument Channel:           15
 time interval backward (min):    99.00000
 time interval forward (min):     99.00000
 target box size:          19
 Nested tracking flag:           1
 Flag to read cloud file:          1
 EE Filtering Not Active


……
```

VPW Unit Tests

VPW Product Generation

- **Validation: VPW Product Generation**
  - » **Generate polar wind product**
    - **Execute pcf_framework.exe**
      - **Log file content (cont.)**

```
Segment Number: 12 of 12
Running: AQUA_MODIS_POLAR
Running: LAND_MASK_NASA_1KM
Running: NWP_GFS
Running: MDS_POLAR_CLD_MASK
Running: MDS_POLAR_CLD_PHASE
Running: MDS_POLAR_CLD_HEIGHT
Running: TEMPORAL_DATA_FIXED
Running: AWG_NOP_AMV

Number of attempts:    24921
Error #1 ... Flat gradient failures:        881
Error #2 ... Earth edge failures:             4
Error #3 ... Cloud amount failures:        5678
Error #4 ... Median pressure not found:       9
Error #5 ... Bad pixel BrtTemp values:      785
Error #6 ... Multiple layers failures:        0
Error #7 ... Coherency failures:              0
Error #8 ... Correlation failures:            0
Error #9 ... u-acceleration failures:       196
Error #10 ... v-acceleration failures:      167
Error #11 ... u and v-accel failures:         5
Error #12 ... slow wind failures:           814
Error #13 ... terminator failures:            0
Error #14 ... height cutoff failures:        93
Error #15 ... match on boundary:              0
Error #16 ... Gross error failures          389
Error #17 ... Medians too different         640
Error #18 ... Search box not full             1
Error #19 ... EE Filtering Count              0
Error #20 ... Missing data in search box    5428
Error #21 ... Zero initial sample           2033
Error #22 ... Zero final sample             4006
```

VPW Unit Tests

VPW Product Generation

- **Validation: VPW Product Generation**
  - **Generate polar wind product**
    - **Execute pcf_framework.exe**
      - **Log file content (cont.)**

```
Number of targets processed:    24921

Number of good targets:         17486

Winds without errors:           3792
 get_best_buddy August 1998
 Calling sorting routine
 Sorting finished
 Total number of targets processed :    24921
 Total number of winds             :    3792
 --- Done EUMAQC ---

 EE MODIFICATION NOT IN EFFECT

 Name Weight
 ---- ------
_CONST        2.8715832233
QSIP         -0.0664423108
QIDR          0.1393737346
QIVS         -0.1762089282
_QILC        -0.2515231073
QIFC         -0.5091211796
SPD           0.0062573059
_PW          -0.0007419194
_SHEAR        0.0098076882
_TEMPG        0.0125888800
 _CONST =    2.871583
 QISP =  -6.6442311E-02 -0.9800000
 QIDR =   0.1393737     -0.9800000
 QIVS =  -0.1762089     -0.9800000
 _QILC =  -0.2515231     -0.9800000
 QIFC =  -0.5091212     -0.9800000
 SPD =   6.2573059E-03  -999.0000
 _PW =  -7.4191939E-04  -999.0000
 _SHEAR =   9.8076882E-03
 _TEMG =   1.2588880E-02
```

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - » **Generate polar wind product**
    - **Execute pcf_framework.exe**
      - **Log file content (cont.)**

```
AQUA_MODIS_POLAR: 0.92
LAND_MASK_NASA_1KM: 1.55
NWP_GFS: 10.6
MDS_POLAR_CLD_MASK: 0.14
MDS_POLAR_CLD_PHASE: 0.12
MDS_POLAR_CLD_HEIGHT: 2.34
TEMPORAL_DATA_FIXED: 2.88
AWG_NOP_AMV: 190.18
Total Process Time:   208.73
Total Time: 229.84
```

VPW Unit Tests

VPW Product Generation

- **Validation: VPW Product Generation**
  - » **Generate polar wind product**
    - **Log file location and name**

```
/data/data046/working/pw /NVPWPS_Proc_PW.pl.log
```

    - **Log file content**

```
NVPWPS_Proc_PW.pl is now starting at 20140304-18:23:09
  PSF_Remap_current = /data/data046/mfan/working/psf_remap_2
  PSF_Remap_before = /data/data046/mfan/working/psf_remap_1
  PSF_Remap_after = /data/data046/mfan/working/psf_remap_3
remap_file_sdr_1 = /data/data046/mfan/working/nps_VIIRS_2013122_0126_49_ALL_3305_3305.nc
remap_file_sdr_2 = /data/data046/mfan/working/nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc
remap_file_sdr_3 = /data/data046/mfan/working/nps_VIIRS_2013122_0448_49_ALL_3305_3305.nc
t1 =2013122_0126_49
t2 =2013122_0307_49
t3 =2013122_0448_49
deltT1 = -6060
deltT2 = 6060

/data/data046/mfan/dlv_2013/VPW/NVPWPS-OPS/bin/pcf_framework.exe
/data/data046/mfan/working/OUT/viirs_results_pw/configs/NH_poleG_2013122_0307_49.cfg
>/data/data046/mfan/working/OUT/viirs_results_pw/logs/NH_poleG_2013122_0307_49.log  OK


pw_name = /data/data046/mfan/working/OUT/viirs_results_pw/results/NPP_VIIRS_2013122_0307_49_AWG_NOP_AMV.nc
new_pw_name = /data/data046/mfan/working/OUT/viirs_results_pw/results/PAMV-VIIRS-CD-
NH_v1r1_npp_s201305020126490_e201305020448490_c201403041827370.nc
The 3 images are:
  /data/data046/mfan/working/nps_VIIRS_2013122_0126_49_ALL_3305_3305.nc
  /data/data046/mfan/working/nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc
  /data/data046/mfan/working/nps_VIIRS_2013122_0448_49_ALL_3305_3305.nc
NVPWPS_Proc_PW.pl is now completing successfully at: 20140304-13:27:40
```

VPW Unit Tests

VPW Product Generation

- **Validation: VPW Product Generation**
  - » **Generate polar wind product**
    - **PSF for NVPWPS_Proc_PW.pl**

```
/data/data046/working/pw /NVPWPS_Proc_PW.pl.PSF
```

    - **PSF Content**

```
/data/data046/mfan/working/OUT/viirs_results_pw/results/PAMV-VIIRS-CD-NH_v1r1_npp_s201305020126490_e201305020448490_c201403041827370.nc
```

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - **Create polar winds PSF file**
    - **Polar winds PSF file location and name**

```
/data/data046/working/VPW_PRODUCT.pl.PSF
```

    - **Polar winds PSF file content**

```
/data/data046/working/OUT/viirs_results_pw/results/PAMV-VIIRS-CD-NH_v1r1_npp_s201305020126490_e201305020448490_c201310151533470.nc
```

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - **Update log file VPW_PRODUCT.pl.log**
    - **Log file location and name**

```
/data/data046/working/VPW_PRODUCT.pl.log
```

    - **Log file content**

```
VPW_PRODUCT.pl is now starting at 20140304-18:23:08

VPW generating
======================
  Creating VPW generating working directory: /data/data046/mfan/working/pw

 The three remapped files are:
  - nps_VIIRS_2013122_0126_49_ALL_3305_3305.nc
  - nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc
  - nps_VIIRS_2013122_0448_49_ALL_3305_3305.nc

   Creating PCF file: /data/data046/mfan/working/pw/NVPWPS_Proc_PW.pl.PCF
  Running NVPWPS_Proc_PW.pl
   - /data/data046/mfan/working/pw/NVPWPS_Proc_PW.pl.log

real    4m31.799s
user    4m2.473s
sys     0m13.674s
running NVPWPS_Proc_PW.pl: OK

VPW_PRODUCT.pl is now completing successfully at: 20140304-13:27:40
```

VPW Unit Tests

VPW Product Generation

- **Validation: VPW Product Generation**
  - » **Generate polar wind product**
    - **PSF for NVPWPS_Proc_PW.pl**

```
/data/data046/working/pw /NVPWPS_Proc_PW.pl.PSF
```

    - **PSF Content**

```
/data/data046/mfan/working/OUT/viirs_results_pw/results/PAMV-VIIRS-CD-NH_v1r1_npp_s201305020126490_e201305020448490_c201403041827370.nc
```

VPW Unit Tests
VPW Product Generation

- **Validation: VPW Product Generation**
  - **Create polar winds PSF file**
    - **Polar winds PSF file location and name**

```
/data/data046/working/VPW_PRODUCT.pl.PSF
```

    - **Polar winds PSF file content**

```
/data/data046/working/OUT/viirs_results_pw/results/PAMV-VIIRS-CD-NH_v1r1_npp_s201305020126490_e201305020448490_c201310151533470.nc
```

VPW Unit Tests

VPW Product Generation

- **Validation: VPW Product Generation**
  - **Update log file VPW_PRODUCT.pl.log**
    - **Log file location and name**

```
/data/data046/working/VPW_PRODUCT.pl.log
```

    - **Log file content**

```
VPW_PRODUCT.pl is now starting at 20140304-18:23:08

VPW generating
======================
  Creating VPW generating working directory: /data/data046/mfan/working/pw

 The three remapped files are:
  - nps_VIIRS_2013122_0126_49_ALL_3305_3305.nc
  - nps_VIIRS_2013122_0307_49_ALL_3305_3305.nc
  - nps_VIIRS_2013122_0448_49_ALL_3305_3305.nc

   Creating PCF file: /data/data046/mfan/working/pw/NVPWPS_Proc_PW.pl.PCF
  Running NVPWPS_Proc_PW.pl
   - /data/data046/mfan/working/pw/NVPWPS_Proc_PW.pl.log

real    4m31.799s
user    4m2.473s
sys     0m13.674s
running NVPWPS_Proc_PW.pl: OK

VPW_PRODUCT.pl is now completing successfully at: 20140304-13:27:40
```