



VIIRS Flood Mapping

Algorithm Readiness Review

February 26, 2021

Presented by:

Sanmei Li¹, Priyanka Roy², Peter Keehn³, Eric Buzan²

¹George Mason University,

²I.M. Systems Group, Inc.

³GAMA-1 Technologies



Review Agenda

Introduction	10.00 – 10.10am	Roy
UTRR Risks & Actions	10.10 – 10.15am	Roy
Requirements	10.15 – 10.25am	Roy
Algorithm Validation	10.25 – 11.00am	Li
Software Architecture	11.00 – 11.30am	Keehn and Buzan
Delivered Algorithm Package	11.30 – 11.40am	Buzan
ARR Risks & Actions	11.40 – 11.45am	Roy
Summary	11.45 – 11.50am	Roy



ARR Outline

- Introduction
- UTRR Risks & Actions
- Requirements
- Algorithm Validation
- Software Architecture
- Delivered Algorithm Package
- ARR Risks & Actions
- Summary



Introduction

Presented by:
Priyanka Roy



Contents

- Project Background
- Project Objectives
- Integrated Product Team
- Stakeholders
- Customers and Users
- Project Schedule
- Entry and Exit Criteria



Background

- **SPSRB requirement number, title and summary**
 - 1805-0003 VIIRS Flood Inundation
 - High resolution remotely sensed data on flood inundation and extent over large spatial domains in the CONUS and OCONUS.
- Accurate spatial information of inundated areas, especially in remote or sparsely populated regions, provides forecasters with valuable data to adjust river models in real time, improving forecasters accuracy and decision support services for flood mitigation efforts.



Project Objective – Products

The following VIIRS Flood detection products will be generated using inputs from S-NPP and NOAA-20 :

- Near Real time Granules and NWS Domain products
 - Resolution: 375m
 - Coverage:
 - Granule: Global
 - NWS Domain: CONUS and Alaska regions
 - Format: Netcdf4, png, ~~geotiff and shapefile~~



Project Objective – Products

- Daily and Rolling 5 day Composite Flood detection Product
 - Resolution: 375 m
 - Coverage: 136 AOIs covering Global land area
 - Format: Netcdf4, png, geotiff and shapefile



Integrated Product Team (IPT)

- **IPT Lead: Walter Wolf (STAR)**
- **IPT Backup Leads: Liqun Ma(OSPO)**
- **NESDIS team:**
 - STAR: Satya Kalluri, Ivan Csiszar
 - OSPO: Zhaohui Cheng, Antonio Irving, Donna McNamera, Chris Sisko
 - OSGS: Jonathan Doran, Ame Fox, Rick Vizbulis
 - ESPDS: Timothy Harline, David Snyder, Jonathan Hansford, Wei Yu, Krishna Tewari
 - JPSS: Lihang Zhou, Mitch Goldberg
 - Others: Sanmei Li (George Mason University), Donglian Sun (George Mason University), Peter Keehn (GAMA1), Priyanka Roy(IMSG), Eric Buzan (IMSG)
- **User team**
 - Lead: Kevin Schrab (NWS), Brian Connelly (NWS North Central River Forecast Center), Madeline Jones (FEMA)



Project Stakeholders

- Development:
 - George Mason University
 - STAR
- Operation Implementation:
 - OSGS
 - ESPDS
 - OSPO
- Users:
 - NWS, NWS/North Central River Forecast Center
 - FEMA



Project Schedule

- Critical Design Review: 11/22/2019
- Unit Test Readiness Review: 10/16/2020
- Initial DAP Delivery: 10/30/2020
- Software Code Review: 11/20/2020
- Algorithm Readiness Review: 2/26/2021
- Final DAP Delivery : March 2021
- Operational Readiness Review: May 2021



ARR Entry Criteria

- Requirements Document
- Review Item Disposition (tracks risks and actions)
- UTRR Report
- Algorithm Readiness Review Sections:
 - Requirements
 - Algorithm Validation Plan
 - Software Architecture & Interfaces
 - Algorithm Package
 - Risks and Actions



ARR Exit Criteria

- Algorithm Readiness Review Report
 - The ARR Report (ARRR) will be compiled after the ARR
 - The report will contain:
 - Updated RID
 - Updated Requirements Document
 - Updated ARR presentation



ARR Outline

- Introduction
- UTRR Risks & Actions
- Requirements
- Algorithm Validation
- Software Architecture
- Delivered Algorithm Package
- ARR Risks & Actions
- Summary



UTRR Report

Presented by:
Priyanka Roy



CDR Risks and Actions

- **Risk #1:** This product has not been identified for operational monitoring using OSPO's Product Monitoring Tool.
- **Impact:** OSPO cannot ensure the quality of the product distributed.
- **Risk Assessment:** Low
- **Risk Mitigation:**
 - PAL will need to work with Product Monitoring team to have this added for being funded.
 - Science team will need to work with ASSISTT integrators to identify metadata for monitoring.
- **Status:** *Closed*

		CONSEQUENCES				
LIKELIHOOD		1	2	3	4	5
	5					
	4					
	3					
	2			X		
	1					



CDR Risks and Actions

- **Action item #4** : OSPO should investigate the process for archiving DAPs at NCEI. This product uses non-enterprise versions of static data that is delivered with the DAP.



ARR Risks and Actions

- **Risk #1:** The NWS mosaic product may not meet the requested timeliness of 40 minutes.
- **Impact:** NWS may not be able to use the outputs.
- **Risk Assessment:** Medium
- **Risk Mitigation:**
 - NDE will need to implement geographic filtering and set up the trigger to process such that the domain is covered and timeliness is met.
- **Status:** Open

		CONSEQUENCES				
		1	2	3	4	5
LIKELIHOOD	5					
	4					
	3					
	2				X	
	1					



Risks and Action Summary

- 1 CDR Risk with Low assessment - Closed.
- 1 CDR Open Action
- 1 ARR Risk with Medium assessment.



ARR Outline

- Introduction
- UTRR Risks & Actions
- Requirements
- Algorithm Validation
- Software Architecture
- Delivered Algorithm Package
- ARR Risks & Actions
- Summary



Requirements

Priyanka Roy



Requirements Documentation

- Requirements are obtained from:
 - » The Flood Mapping project plan
 - » NDE DAP requirements documents
 - » SPSRB process standards
 - » OSPO Technical Reference Manual
 - » Meetings and communications with integrators, science teams, and users
 - » Archive requirements
- Requirements are documented in an associated Requirements Document kept in google drive with other project-related documents.
- Requirements updated/changed since the last review are highlighted in green.



Requirements Organization

1. Processes and Documentation
 - A high-level overview of the process-related guidelines, the document package, production plan, and roles and responsibilities
2. System Requirements
 - Details regarding the system requirements for the software package
3. Software and Hardware
 - Details of the software, software security checklist, and hardware.
4. Algorithm
 - Algorithm and upgrades.
5. Data Products
 - Level 1 requirements and data requirements for the data product.



Basic Requirement: 1.0

- **Section 1: Process and Documentation**
 - **Process Guidelines**
 - **Delivered Algorithm Package**
 - **Production**
 - **Roles and Responsibilities**
 - **Archive Plan**



Basic Requirement: 1.0

Process Guidelines

- **VIIRS-FMPS-R 1.0:** *The VIIRS Flood Mapping Product System (VIIRS FMPS) development project shall adopt the standard practices of the Satellite Product and Services Review Board (SPSRB).*
 - Driver: SPSRB reviews add value to product development.



Basic Requirement: 1.0

- **VIIRS-FMPS-R 1.0.1:** *The VIIRS FMPS development project practices shall be tailored from the SPSRB process.*
 - This requirement should be met by following the SPSRB process, as long as the tailoring does not introduce an incompatibility.



Basic Requirement: 1.0

Delivered Algorithm Package

- **VIIRS-FMPS-R 1.1:** *STAR shall deliver a Delivered Algorithm Package (DAP) to OSPO.*
- **VIIRS-FMPS-R 1.1.1:** *The VIIRS FMPS DAP shall include a document package.*
- **VIIRS-FMPS-R 1.1.1.1:** *The VIIRS FMPS document package shall include a README text file*
 - The README file shall list each item in the final pre-operational system baseline, including code, test data, and documentation



Basic Requirement: 1.0

- **VIIRS-FMPS-R 1.1.1.2:** *The VIIRS FMPS document package shall include an Algorithm Theoretical Basis Document (ATBD).*
- **VIIRS-FMPS-R 1.1.1.3:** *The VIIRS FMPS document package shall include a System Maintenance Manual (SMM).*
- **VIIRS-FMPS-R 1.1.1.4:** *The VIIRS FMPS document package shall include an External Users Manual (EUM).*



Basic Requirement: 1.0

- **VIIRS-FMPS-R 1.2:** *STAR shall maintain a VIIRS FMPS document package.*
- **VIIRS-FMPS-R 1.2.2:** *The VIIRS FMPS document package shall include a Review Item Disposition (RID) document*
 - The RID shall describe the final status of all development project tasks, work products, and risks
- **VIIRS-FMPS-R 1.2.3:** *The VIIRS FMPS document package include a Requirements Allocation Document (RAD).*



Basic Requirement: 1.0

- **VIIRS-FMPS-R 1.2.4:** *The VIIRS FMPS document package shall include a Critical Design Review Report (CDRR).*
- **VIIRS-FMPS-R 1.2.5:** *The VIIRS FMPS document package shall include a Software Code Review Report (SCRR).*
- **VIIRS-FMPS-R 1.2.6:** *The VIIRS FMPS document package shall include a Unit Test Readiness Review Report (UTRRR).*



Basic Requirement: 1.0

- **VIIRS-FMPS-R 1.2.7:** *The VIIRS FMPS document package shall include a Algorithm Readiness Review Report (ARRR).*
 - The ARRR shall document the approved readiness of the VIIRS FMPS system for transition to operations.



Basic Requirement: 1.0

Production

- **VIIRS-FMPS-R 1.3:** *The VIIRS FMPS shall write product files in NetCDF4 format.*
 - SPSRB Requirement
- **VIIRS-FMPS-R 1.4:** *The VIIRS FMPS system shall generate metadata for each retrieved product.*
 - Driver: Metadata will be used by the Product Monitoring Project
- **VIIRS-FMPS-R 1.5:** *The VIIRS FMPS system shall write metadata into the NetCDF4 files associated with the retrieved products.*



Basic Requirement: 1.0

- **VIIRS-FMPS-R 1.5.1:** *The metadata shall include overall quality and summary level metadata.*
- **VIIRS-FMPS-R 1.5.2:** *The metadata shall include Granule metadata.*
- **VIIRS-FMPS-R 1.5.3:** *The metadata shall include Geographic metadata.*
- **VIIRS-FMPS-R 1.5.4:** *The metadata shall include product specific metadata as Quality Information Attributes .*



Basic Requirement: 1.0

- **VIIRS-FMPS-R 1.6:** *The VIIRS FMPS system shall have QC monitoring capability*
 - Driver: This basic requirement is traced to an OSPO need for QC monitoring.
- **VIIRS-FMPS-R 1.6.1:** *The VIIRS FMPS Product files shall include overall quality control flags and quality summary level metadata.*
 - Needed for distribution, quality control and post-processing. VIIRS FMPS code will generate metadata for this purpose.



Basic Requirement: 1.0

- **VIIRS-FMPS-R 1.6.2:** *The VIIRS FMPS system shall be capable of monitoring input data latency and overall quality.*
 - Need to import metadata from input file and create code for generating metadata.
- **VIIRS-FMPS-R 1.6.3:** *The VIIRS FMPS system shall be capable of monitoring product latency.*
 - Run status file will include processing time.



Basic Requirement: 1.0

- **VIIRS-FMPS-R 1.6.4:** *The VIIRS FMPS system shall be capable of monitoring product distribution status to ensure that the data/products are successfully available for transfer to the user community.*
 - A run status file will be produced. Work with OSPO to determine needs
- **VIIRS-FMPS-R 1.6.5:** *Each run status file shall include all runtime error messages.*
 - Error messages will include system messages and error conditions written by the code



Basic Requirement: 1.0

- **VIIRS-FMPS-R 1.6.6:** *Each run status file shall indicate whether or not the run was completed without error.*
 - Code will write this message. This indication will be the last message in the file, so that operators can find it easily
- **VIIRS-FMPS-R 1.6.7:** *The VIIRS FMPS system shall write a log file for each production run.*
 - Used by OSPO for QC monitoring and troubleshooting.



Basic Requirement: 1.0

Roles and Responsibilities

- **VIIRS-FMPS-R 1.7:** *Algorithm Products shall be validated and verified.*
- ~~**VIIRS-FMPS-R 1.7.1:** *The VIIRS FMPS system shall plot datasets for verification of the Algorithm Products.*~~
- **VIIRS-FMPS-R 1.7.2:** *The VIIRS FMPS system shall verify that Algorithm Products files are generated correctly.*
 - Will be included in the unit tests described in the UTR and the system test described in the ARR



Basic Requirement: 1.0

- **VIIRS-FMPS-R 1.7.3:** *The VIIRS FMPS system shall perform routine data range checks to flag anomalous values in the input data*
 - Anomalous values will be flagged. These checks will be included in the code and described in the ARR
- **VIIRS-FMPS-R 1.7.4:** *The VIIRS FMPS system shall perform routine data range checks to flag anomalous values in the Algorithm Products.*
 - Out-of-range values will be flagged. These checks will be included in the code. UTR will address
- ~~**VIIRS-FMPS-R 1.7.5:** *The VIIRS FMPS system shall generate matchup datasets between Algorithm Products retrievals and in situ measurements*~~



Basic Requirement: 1.0

Archive Plan

- **VIIRS-FMPS-R 1.8:** *The VIIRS Flood Detection granules, daily composite, ~~daily flood mask~~, 5-day rolling composite, and ~~5-day flood mask~~ products in netcdf4 format shall be archived at NCEI.*



Basic Requirement: 2.0

- **Section 2: System**
 - **Demonstration Mode**
 - **Development Environment Testing**
 - **Delivered Algorithm Package**



Basic Requirement: 2.0

Demonstration Mode

- **VIIRS-FMPS-R 2.0:** *The VIIRS FMPS shall produce a fully functional pre-operational demonstration system in the STAR Development Environment.*
 - Driver: This basic requirement is traced to an NDE need for a unit-tested, fully functional system delivered to its Test Environment.



Basic Requirement: 2.0

- **VIIRS-FMPS-R 2.0.1:** *The STAR Development Environment shall be capable of hosting the conversion of VIIRS FMPS science code to VIIRS FMPS pre-operational code.*
- **VIIRS-FMPS-R 2.0.2:** *The STAR Development Environment shall include the GNU C and C++ compiler.*
 - Needed for the science code. Development Environment servers have this



Basic Requirement: 2.0

- **VIIRS-FMPS-R 2.0.3:** *The VIIRS FMPS processing code shall be able to run in the STAR Development Environment (Linux with 10 dual core 3.2 GHz CPUs) and IDL for Validation Storage: 100 TB)*
 - C code, C++ code, and Fortran code can run in this environment
- **VIIRS-FMPS-R 2.0.4:** *The STAR Development Environment shall be capable of hosting unit tests and a system test*
 - Unit tests and system test required prior to delivery of pre-operational demonstration system to OSPO



Basic Requirement: 2.0

- **VIIRS-FMPS-R 2.0.5:** *The STAR Development Environment shall have access to OSPO PDA*
 - For ingest of input products for the JPSS RR products
- **VIIRS-FMPS-R 2.0.6:** *The STAR Development Environment shall have access to the GRAVITE server.*
 - For ingest of input data for VIIRS Imagery products
- **VIIRS-FMPS-R 2.0.7:** *The STAR Development Environment shall host the pre-operational demonstration system.*
 - For development and unit testing. Complete unit test of the pre-operational system is expected before delivery to NDE



Basic Requirement: 2.0

- **VIIRS-FMPS-R 2.0.8:** *The pre-operational demonstration system shall include all processing code and ancillary files needed to conduct unit tests*
 - Complete unit test of the pre-operational system is expected before delivery to NDE. The UTRR will provide a detailed description of the source code units and ancillary files
- **VIIRS-FMPS-R 2.0.9:** *The pre-operational demonstration system shall include all input test data needed to conduct unit tests.*
 - Complete unit test of the pre-operational system is expected before delivery to NDE. The UTRR will provide a detailed description of the unit test data



Basic Requirement: 2.0

- **VIIRS-FMPS-R 2.0.10:** *The VIIRS FMPS pre-operational demonstration system baseline shall be established and maintained with the Git CM tool.*
 - CM of the pre-operational system is expected throughout its development



Basic Requirement: 2.0

Development Environment Testing

- **VIIRS-FMPS-R 2.1:** *The VIIRS FMPS integrated pre-operational demonstration system shall be transitioned from the STAR Development Environment to the NDE.*
 - Driver: This basic requirement is traced to an NDE need for a system-tested, integrated pre-operational system delivered to its Test Environment.
- **VIIRS-FMPS-R 2.1.1:** *The STAR Development Environment shall host the VIIRS FMPS integrated pre-operational demonstration system*
 - For system testing. A complete system test of the integrated pre-operational system is expected before delivery to NDE



Basic Requirement: 2.0

- **VIIRS-FMPS-R 2.1.2:** *The integrated pre-operational demonstration system shall include all processing code and ancillary files needed to conduct the system test*
 - Complete system test of the integrated pre-operational system is expected. The ARR will provide a description of the processing software system and ancillary files
- **VIIRS-FMPS-R 2.1.3:** *The integrated pre-operational demonstration system shall include all input data needed to conduct a system test*
 - Complete system test of the integrated pre-operational system is expected. The ARR will provide a description of the system test data



Basic Requirement: 2.0

- **VIIRS-FMPS-R 2.1.4:** *The integrated pre-operational demonstration system shall include all output data produced by the system test*
 - Needed by NDE to verify the system test in its Test Environment. Comparison of outputs from system test in STAR and NDE environments will be part of the NDE system test. Specific items will be listed in the ARR
- **VIIRS-FMPS-R 2.1.5:** *The VIIRS FMPS integrated pre-operational demonstration system baseline shall be established and maintained with the Git CM tool.*
 - CM of the integrated pre-operational system is expected throughout its development



Basic Requirement: 2.0

Delivered Algorithm Package

- **VIIRS-FMPS-R 2.3:** *The integrated pre-operational demonstration system shall be delivered to NDE via google drive or FTP as a Delivered Algorithm Package (DAP).*
- **VIIRS-FMPS-R 2.3.1:** *The VIIRS FMPS development team shall ensure that the NDE integrators and OSPO PAL has the information needed to acquire the VIIRS FMPS DAP.*



Basic Requirement: 3.0

- **Section 3: Software and Hardware**
 - **Software**
 - **Security Checks**
 - **Hardware**



Basic Requirement: 3.0

Software

- **VIIRS-FMPS-R 3.0:** *The VIIRS Flood detection software will be implemented to produce the Flood detection products.*
 - Driver: This basic requirement is traced to user needs for high resolution remotely sensed data on flood inundation.



Basic Requirement: 3.0

- **VIIRS-FMPS-R 3.0.1:** *The VIIRS FMPS Algorithms shall be implemented by processing codes written in C, C++ and Python.*
- **VIIRS-FMPS-R 3.0.2:** *The VIIRS FMPS science code will be wrapped in Python scripts.*
- **VIIRS-FMPS-R 3.0.3:** *The VIIRS FMPS processing code shall be able to run in the NDE Test Environment (Linux machine with 6 quad core 3.2 GHz CPUs S/W: Intel and GNU Compilers (C/C++/Fortran) and IDL for Validation Storage: 30 TB)*
 - C code, C++ code, and Fortran code can run in this environment



Basic Requirement: 3.0

- **VIIRS-FMPS-R 3.0.4:** *The VIIRS FMPS processing code shall be able to run in the OSPO Operations Environment: (Linux machine with 6 quad core 3.2 GHz CPUs S/W: Intel and GNU Compilers (C/C++/Fortran) and IDL for Validation Storage: 30 TB)*
 - C code, C++ code, and Fortran code can run in this environment



Basic Requirement: 3.0

Security Checks

- **VIIRS-FMPS-R 3.1:** *The VIIRS FMPS system shall undergo an OSPO Code Review for security compliance.*
 - Driver: OSPO Security
- **VIIRS-FMPS-R 3.1.1:** *The VIIRS FMPS system shall comply with OSPO data integrity check list.*
 - OSPO data integrity check list is part of the OSPO Code Review Security check lists
- **VIIRS-FMPS-R 3.1.2:** *The VIIRS FMPS 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: 3.0

- **VIIRS-FMPS-R 3.1.3:** *The VIIRS FMPS system shall comply with OSPO code check list.*
 - OSPO code check list is part of the OSPO Code Review Security check lists



Basic Requirement: 3.0

Hardware

- **VIIRS-FMPS-R 3.2:** *IT resource needs for operations shall be specified.*
 - Driver: OSPO IT Capacity Planning
- **VIIRS-FMPS-R 3.2.1:** *The VIIRS FMPS system shall run on Redhat Linux*
 - Servers are available



Basic Requirement: 3.0

- **VIIRS-FMPS-R 3.2.2:** *Operational server shall have 30 TB of disk space*
 - Available servers have this capability
- **VIIRS-FMPS-R 3.2.3:** *Each operational server shall have 8 GB of RAM for each core.*
 - Available servers have this capability



Basic Requirement: 4.0

- **Section 4: Algorithm and Data Products**
 - **Algorithm Product**



Basic Requirement: 4.0

- **VIIRS-FMPS-R 4.0:** *The VIIRS Flood detection algorithm shall generate the Flood detection products.*
 - Driver: SPSRB requirements: 1805-0003 VIIRS Flood Inundation
 - High resolution remotely sensed data on flood inundation and extent over large spatial domains in the CONUS and OCONUS.



Basic Requirement: 5.0

- **Section 5: NRT Data Products**
 - **Level 1 Requirements**
 - **Data Ingest Requirements**
 - **Tailored Formats**



Basic Requirement: 5.0

- **VIIRS-FMPS-R 5.0:** *The Flood Detection algorithm shall generate near-real time (NRT) Flood detection Products.*



Basic Requirement: 5.0

- **VIIRS-FMPS-R 5.0.1:** *The Flood Detection algorithm shall generate NRT Flood detection products at granule level.*
- **VIIRS-FMPS-R 5.0.2:** *The Flood Detection algorithm shall generate NRT Flood Detection products for 8 NWS domains.*



Basic Requirement: 5.0

Level 1 Requirements

- **VIIRS-FMPS-R 5.1:** *The NRT Flood Detection products shall have following coverage: Global coverage for granule level product and CONUS and Alaska coverage for NWS domain product.*
- **VIIRS-FMPS-R 5.2:** *The NRT Flood Detection Products shall have a latency: 40 mins for granule product and 40 mins after the arrival of the last granule covering the domain for NWS domain product*



Basic Requirement: 5.0

- **VIIRS-FMPS-R 5.3:** *The NRT Flood Detection Products shall have timeliness of ≤ 2 hours.*
- **VIIRS-FMPS-R 5.4:** *The NRT Flood Detection Products shall have Horizontal Resolution of 375m*
- **VIIRS-FMPS-R 5.5:** *The NRT Flood Detection Products shall have Measurement Range of: Water fractions between 25% and 100%.*
- **VIIRS-FMPS-R 5.6:** *The NRT Flood Detection Products shall have Mapping Accuracy: 80%*



Basic Requirement: 5.0

Data Ingest Requirements

- **VIIRS-FMPS-R 5.7:** *The VIIRS FMPS system shall use S-NPP and NOAA-20 VIIRS Imager Bands (SVI01, SVI02, SVI03, SVI05) and Geolocation (GITCO) data.*
 - Will be ingested from IDPS via PDA
- **VIIRS-FMPS-R 5.8:** *The VIIRS FMPS system shall use S-NPP and NOAA-20 Enterprise ~~Snow Cover~~ and Cloud Mask data.*
 - Will be ingested from ESPDS/NDE



Basic Requirement: 5.0

- **VIIRS-FMPS-R 5.9:** *The VIIRS FMPS system shall use static ancillary data.*
 - Will be included in the DAP
- **VIIRS-FMPS-R 5.9.1:** *The VIIRS FMPS system shall use global land cover data at 1km resolution.*
- **VIIRS-FMPS-R 5.9.2:** *The VIIRS FMPS system shall use global digital elevation model at 375m resolution.*
- **VIIRS-FMPS-R 5.9.3:** *The VIIRS FMPS system shall use a sun-glint look-up table.*



Basic Requirement: 5.0

- **VIIRS-FMPS-R 5.9.4:** *The VIIRS FMPS system shall use a global land/sea mask at 1km resolution.*
- **VIIRS-FMPS-R 5.9.5:** *The VIIRS FMPS system shall use global water mask resampled at 375m resolution.*
- **VIIRS-FMPS-R 5.9.6:** *The VIIRS FMPS system shall use land/sea surface temperature 16-day climatology at 5km resolution.*



Basic Requirement: 5.0

- **VIIRS-FMPS-R 5.9.7:** *The VIIRS FMPS system shall use global albedo monthly climatology at 5km resolution in visible channel.*
- **VIIRS-FMPS-R 5.9.8:** *The VIIRS FMPS system shall use pre-trained decision trees and tree attribute files.*
- **VIIRS-FMPS-R 5.9.9:** *The VIIRS FMPS system shall use user defined AOI definition files for listing the geographic information of each subset.*



Basic Requirement: 5.0

Tailored Formats

- ~~**VIIRS-FMPS-R 5.10:** *The VIIRS FMPS system shall generate products in geotiff, and shapefile formats.*~~
- ~~**VIIRS-FMPS-R 5.10.1:** *The VIIRS FMPS system shall generate tailored granule products in geotiff formats.*~~
- ~~**VIIRS-FMPS-R 5.10.2:** *The VIIRS FMPS system shall generate tailored domain products in geotiff, and shapefile formats.*~~



Basic Requirement: 6.0

- **Section 6: Composite Data Products**
 - **Level 1 Requirements**
 - **Data Ingest Requirements**
 - **Tailored Formats**



Basic Requirement: 6.0

- **VIIRS-FMPS-R 6.0:** *The Flood Detection algorithm shall generate Flood detection composite products.*
 - Driver: Timely product for global users.



Basic Requirement: 6.0

- **VIIRS-FMPS-R 6.0.1:** *The Flood Detection algorithm shall generate 6 hr rolling 24hr Flood detection composite products.*
- ~~VIIRS-FMPS-R 6.0.2:~~ ~~*The Flood Detection algorithm shall generate a daily composite flood mask product.*~~
- **VIIRS-FMPS-R 6.0.3:** *The Flood Detection algorithm shall generate rolling 5 day Flood detection composite products.*
- ~~VIIRS-FMPS-R 6.0.4:~~ ~~*The Flood Detection Product shall generate a 5-day composite flood mask product.*~~



Basic Requirement: 6.0

Level 1 Requirements

- **VIIRS-FMPS-R 6.1:** *The Flood Detection composite products shall have global coverage.*
- **VIIRS-FMPS-R 6.2:** *The Flood Detection composite products shall have Horizontal Resolution of 375m.*
- **VIIRS-FMPS-R 6.3:** *The Flood Detection Product composites shall have a latency of 3 hours after the last granule is available.*
- **VIIRS-FMPS-R 6.4:** *The Flood Detection Product composites shall have timeliness of ≤ 3 hours.*



Basic Requirement: 6.0

- **VIIRS-FMPS-R 6.5:** *The Flood Detection Product composites shall have Measurement Range of: Water fractions between 25% and 100%.*
- **VIIRS-FMPS-R 6.6:** *The Flood Detection Product composites shall have Mapping Accuracy: 80%*
- **VIIRS-FMPS-R 6.7:** *The Flood Detection algorithm shall generate flood detection daily and 5 day composite products in 136 AOIs.*



Basic Requirement: 6.0

- **VIIRS-FMPS-R 6.8:** *The Flood Detection daily composite ~~and mask products~~ shall be generated every 6 hours.*
- **VIIRS-FMPS-R 6.9:** *The Flood Detection 5-day composites ~~and mask products~~ shall be generated every 24 hours.*



Basic Requirement: 6.0

Data Ingest Requirements

- **VIIRS-FMPS-R 6.10:** *The VIIRS FMPS system shall use S-NPP and NOAA-20 NRT Flood Detection reprojected granules to generate the flood detection composite products*
 - Primary Input Data, generated by the NRT processing
- **VIIRS-FMPS-R 6.10.1:** *The VIIRS FMPS system shall use S-NPP and NOAA-20 daily composite AOIs as input to generate the rolling 5 day flood detection composite products.*



Basic Requirement: 6.0

- **VIIRS-FMPS-R 6.11:** *The VIIRS FMPS system shall use static ancillary data.*
 - Will be included in the DAP
- **VIIRS-FMPS-R 6.11.1:** *The VIIRS FMPS system shall use a global land/sea mask at 1km resolution.*
- **VIIRS-FMPS-R 6.11.2:** *The VIIRS FMPS system shall use global water mask resampled at 375m resolution.*
- **VIIRS-FMPS-R 6.11.3:** *The VIIRS FMPS system shall use user defined AOI definition files for listing the geographic information of each subset.*



Basic Requirement: 6.0

Tailored Formats

- **VIIRS-FMPS-R 6.12:** *The VIIRS FMPS system shall generate Flood detection composite tailored products in geotiff and shapefile formats.*



Requirement Summary

- Product requirements have been reviewed.
- Individual product requirements have been documented in the Requirements Document (RAD).



Requirements Summary

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



ARR Outline

- Introduction
- CDR Risks and Actions
- Requirements
- Algorithm Validation
- Software Architecture
- Unit Tests
- Risk & Actions
- Summary



Algorithm Validation

Presented by
Sanmei Li

George Mason University



Validation Data Sets

We requested the following datasets from ASSISTT/T4 to test the results:

- VIIRS flood maps in the west gulf region from Aug. 27 to 29, 2020 for hurricane Laura
- VIIRS flood maps in the west gulf region from Oct. 10 to 15 for hurricane Delta
- VIIRS flood maps with geographic range in three regions outside of the CONUS:
 - 122E~ 142E, 42N~54N, Oct. 8 to Oct. 13, 2020;
 - 3E~22E, 3N~15N, Oct. 8 to Oct. 13, 2020;
 - 58W~48W, 38S ~ 25S, Oct. 8 to 13, 2020



Validation Methods

- Output formats and contents check
 - NetCDF4 check
 - Geotiff check
 - Shapefile check
- Algorithm performance check
 - Visual inspection with the VIIRS false-color images
 - Validation with Sentinel-2 images
 - Quantitative analysis with Sentinel-2 images



NetCDF4 check

VIIRS-Flood_v1r0_n20_s202010111913117_e2020101119

- QualityFlag
- WaterDetection
- col
- row

Properties - /WaterDetection

General Attributes

Number of attributes = 13

Name	Value	Type	Array Size
DIMENSION_LIST	514, 239	Variable-length of Objec...	2
ProjectionMaxLatitude	33.7609	32-bit floating-point	1
ProjectionMaxLongitude	-76.30884	32-bit floating-point	1
ProjectionMinLatitude	24.1879	32-bit floating-point	1
ProjectionMinLongitude	-108.299	32-bit floating-point	1
ProjectionResolution	0.003372	32-bit floating-point	1
Type	1, 15, 16, 17, 20, 27, 30,...	32-bit integer	11
TypeDescription	1: no_valid_data; 15: FI...	String, length = 250	Scalar
_FillValue	1	8-bit unsigned character	1
add_offset	0.0	32-bit floating-point	1
long_name	WaterDetection	String, length = 14	Scalar
scale_factor	1.0	32-bit floatino-point	1

Close

General Attributes User Block

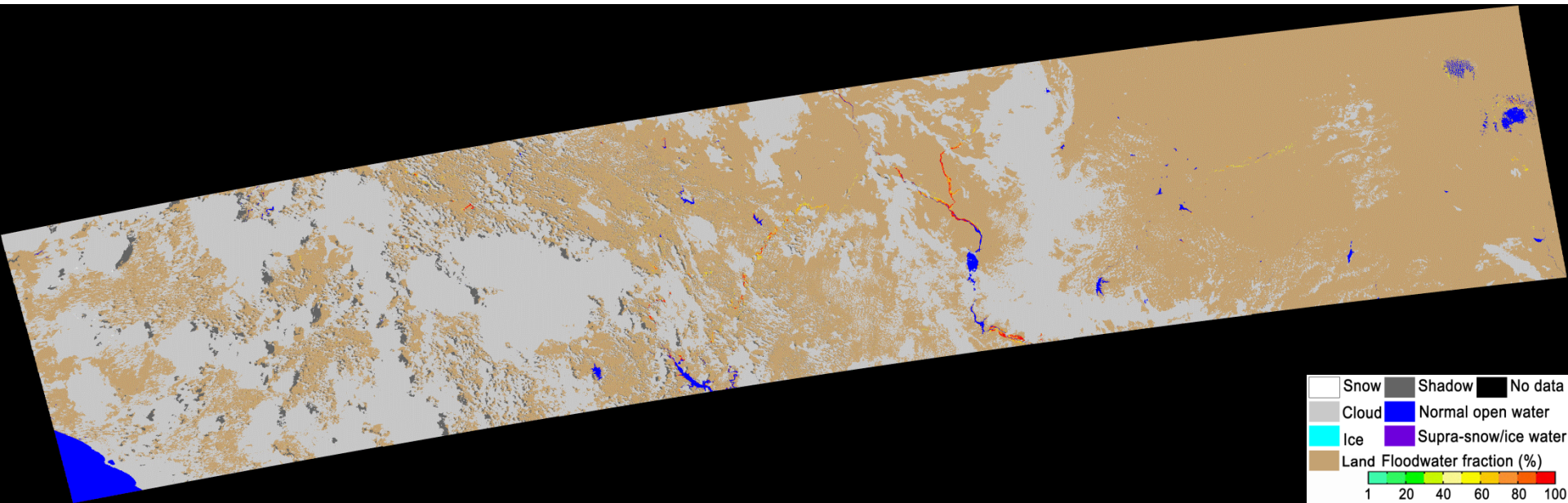
Number of attributes = 45

Name	Value	Type	Array Size
Conventions	CF-1.6, ACDD-1.3	String, length = 16	Scalar
DaynightFlag	1	32-bit integer	1
Observingdate	20201011	String, length = 8	Scalar
Observingtime	191311.756027Z	String, length = 14	Scalar
Resolution	0.003372	32-bit floating-point	1
Satellitename	n20	String, length = 3	Scalar
SensorIdentifyCode	VIIRS	String, length = 5	Scalar
ascent_descend_data_fl...	0	32-bit integer	1
cdm_data_type	Grid	String, length = 4	Scalar
creator_email	slia@gmu.edu	String, length = 12	Scalar
creator_name	The VIIRS Flood Team in...	String, length = 76	Scalar
creator_url	https://jpsfflood.gmu.edu	String, length = 25	Scalar
date_created	2021-02-09T20:32:00Z	String, length = 20	Scalar
day_night_data_flag	day	String, length = 3	Scalar
end_orbit_number	15017	32-bit integer	1
flood_pixel_count	75822	32-bit integer	1
flood_pixel_percent	0.5394278	32-bit floating-point	1
geospatial_lat_max	33.7609	32-bit floating-point	1
geospatial_lat_min	24.1879	32-bit floating-point	1
geospatial_lat_units	degrees_north	String, length = 13	Scalar
geospatial_lon_max	-76.30884	32-bit floating-point	1
geospatial_lon_min	-108.299	32-bit floating-point	1
geospatial_lon_units	degrees_east	String, length = 12	Scalar
history	VIIRS Flood Detection Ve...	String, length = 35	Scalar
id	VIIRS-Flood_v1r0_n20_s...	String, length = 74	Scalar
institution	DOC/NOAA/NESDIS/OSP...	String, length = 109	Scalar
instrument	VIIRS	String, length = 5	Scalar
metadata_link	VIIRS-Flood_v1r0_n20_s...	String, length = 74	Scalar
naming_authority	gov.noaa.nesdis.ncei	String, length = 20	Scalar
platform	n20	String, length = 3	Scalar
processing_level	NOAA Level 2	String, length = 12	Scalar
production_environment	Unknown	String, length = 7	Scalar
production_site	Unknown	String, length = 7	Scalar
project	NPP Data Exploitation	String, length = 21	Scalar
publisher_email	espcoperations@noaa.g...	String, length = 23	Scalar
publisher_name	DOC/NOAA/NESDIS/ND...	String, length = 87	Scalar
publisher_url	http://www.ospo.noaa.gov	String, length = 24	Scalar
references	https://www.ssec.wisc.ed...	String, length = 56	Scalar
source	GITCO, VIIRS I-bands: 01...	String, length = 36	Scalar
standard_name_vocabul...	CF Standard Name Tabl...	String, length = 52	Scalar
start_orbit_number	15017	32-bit integer	1
summary	VIIRS 375-m flood water f...	String, length = 40	Scalar
time_coverage_end	2020-10-11T19:14:36Z	String, length = 20	Scalar
time_coverage_start	2020-10-11T19:13:11Z	String, length = 20	Scalar
title	VIIRS_Flood_Detection	String, length = 21	Scalar

The datasets and the attributes are output correctly.

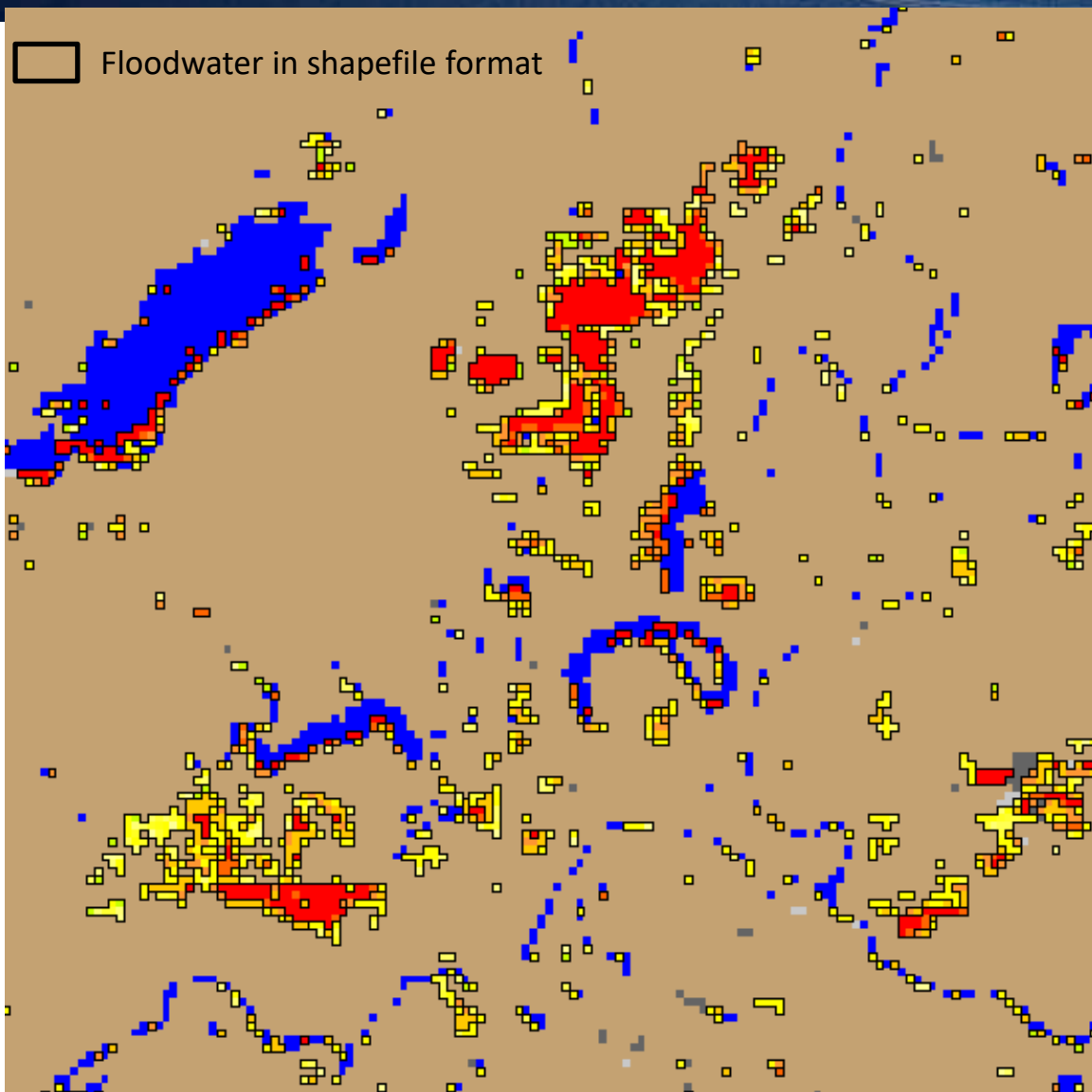


Output in geotiff format



RGB 24-bit images

Output in shapefile format



FID	Shape	DN
0	Polygon	6
1	Polygon	6
2	Polygon	8
3	Polygon	4
4	Polygon	8
5	Polygon	6
6	Polygon	8
7	Polygon	6
8	Polygon	6
9	Polygon	6
10	Polygon	8
11	Polygon	8
12	Polygon	6
13	Polygon	4
14	Polygon	4
15	Polygon	8
16	Polygon	8
17	Polygon	10
18	Polygon	6
19	Polygon	4
20	Polygon	4
21	Polygon	10
22	Polygon	6
23	Polygon	4
24	Polygon	6
25	Polygon	4
26	Polygon	4
27	Polygon	4
28	Polygon	6
29	Polygon	4
30	Polygon	6
31	Polygon	4

Floodwater data in shapefile format (5 attributes in one layer):

- 2: 1~20%
- 4: 21~40%
- 6: 41~60%
- 8: 61~80%
- 10: 81~100%



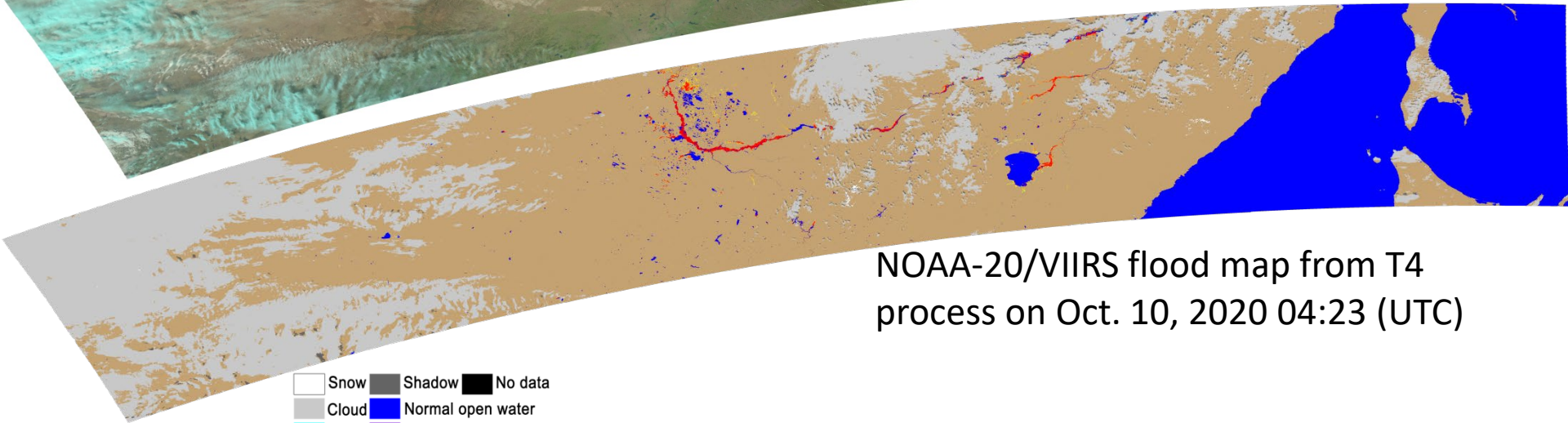
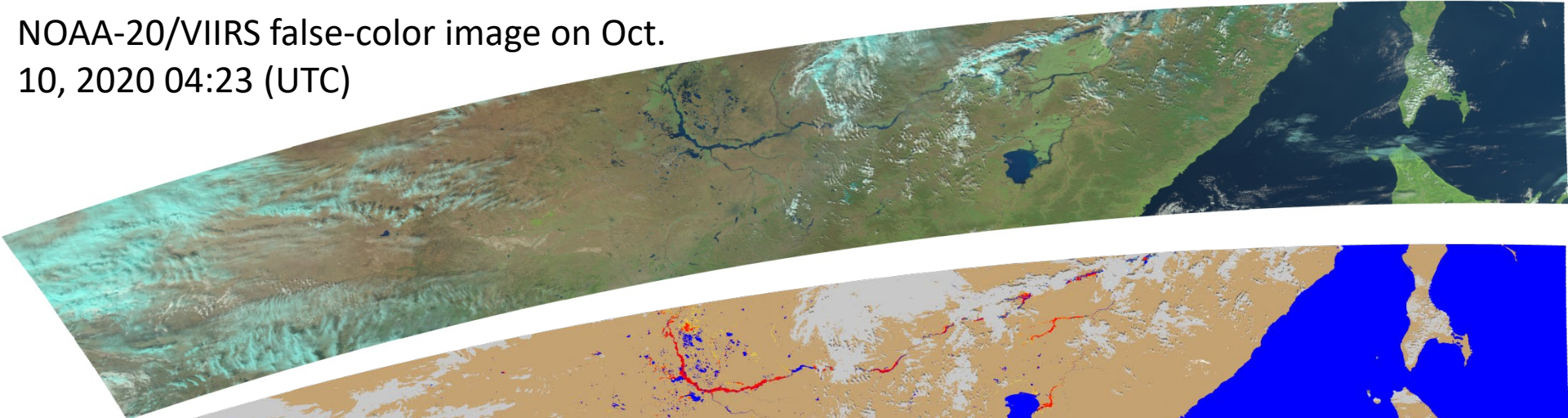
Algorithm Performance – Visual Inspection

- Altogether 453 VIIRS flood maps from ASSISTT/T4 are visually inspected with the corresponding VIIRS false-color images.
- These images cover Asia, Africa, North America and South America
- Visual inspection shows reliable results from the VIIRS flood detection.

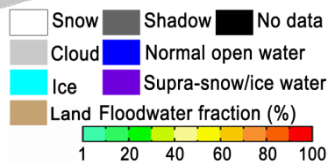


Algorithm Performance – Visual Inspection

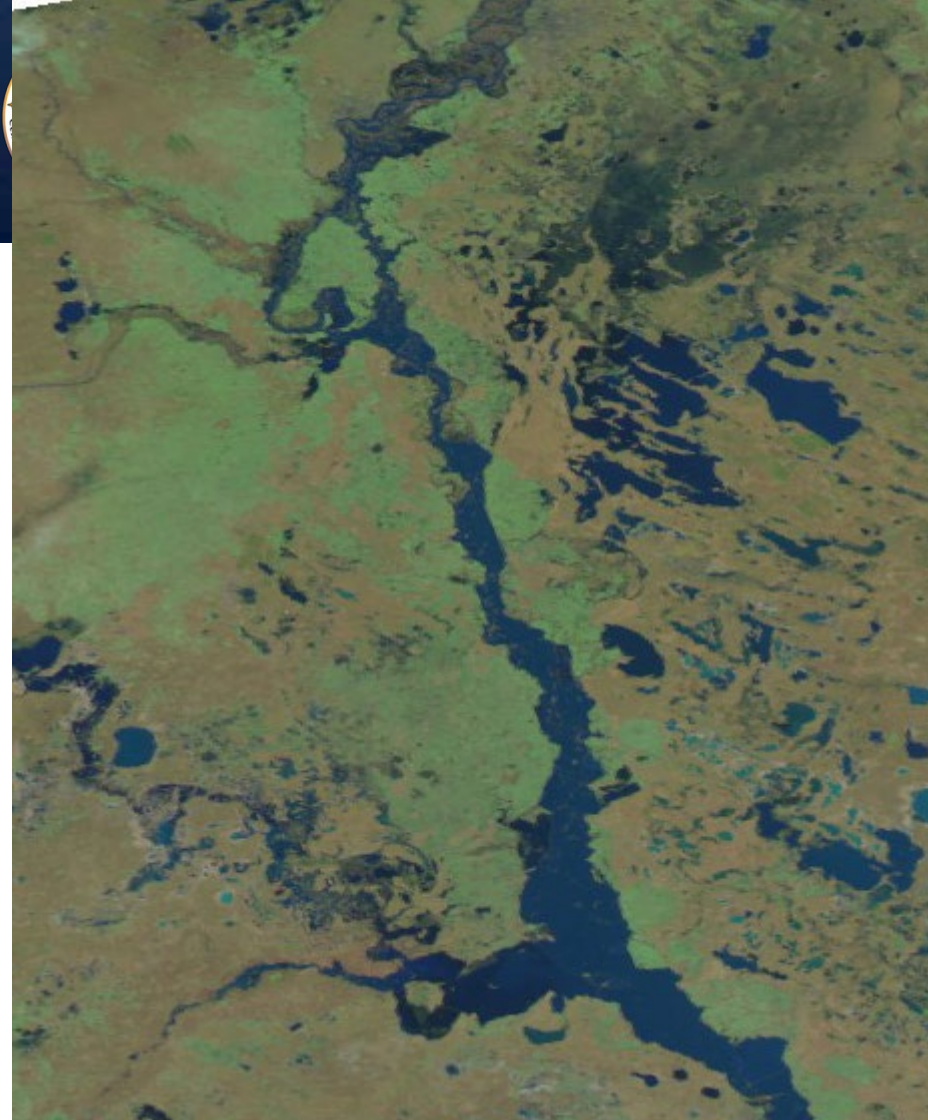
NOAA-20/VIIRS false-color image on Oct.
10, 2020 04:23 (UTC)



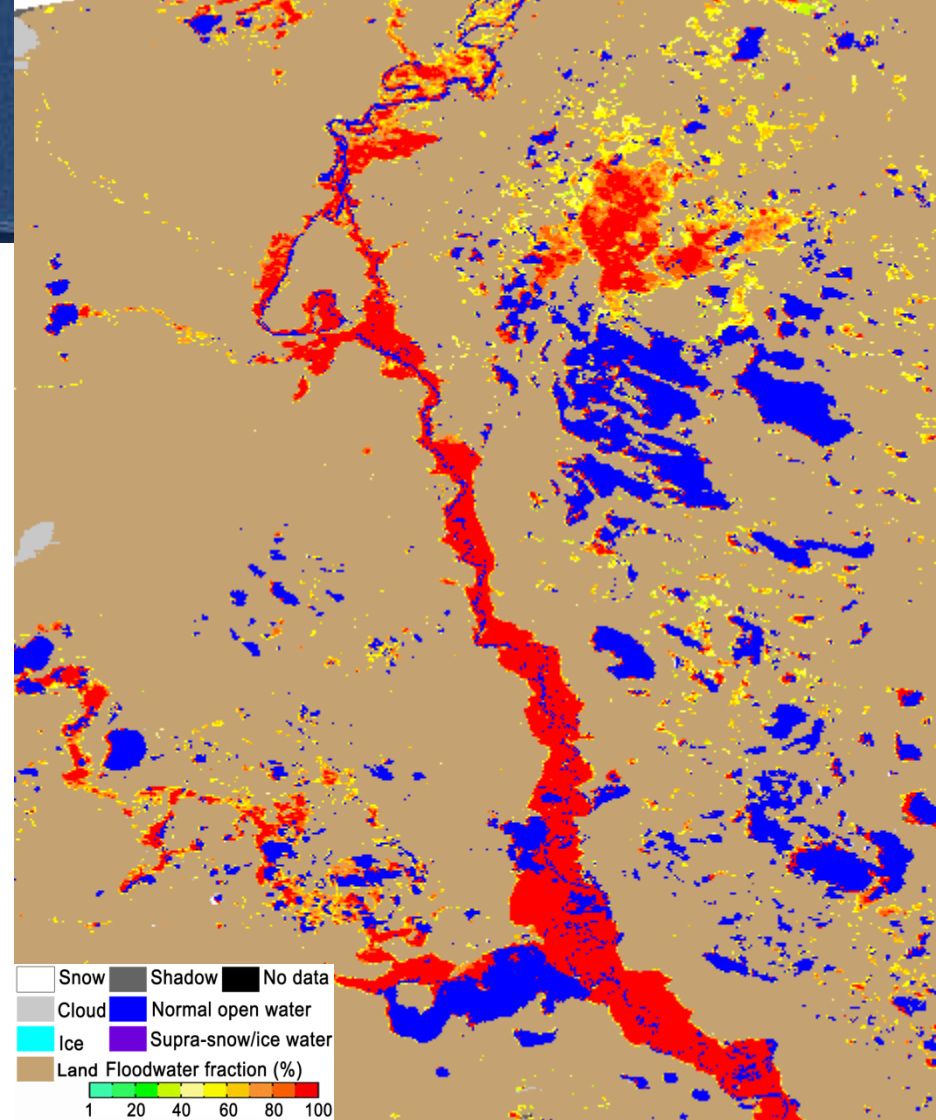
NOAA-20/VIIRS flood map from T4
process on Oct. 10, 2020 04:23 (UTC)



- VIIRS flood maps from T4 process are visually inspected with the VIIRS false-color images one by one.

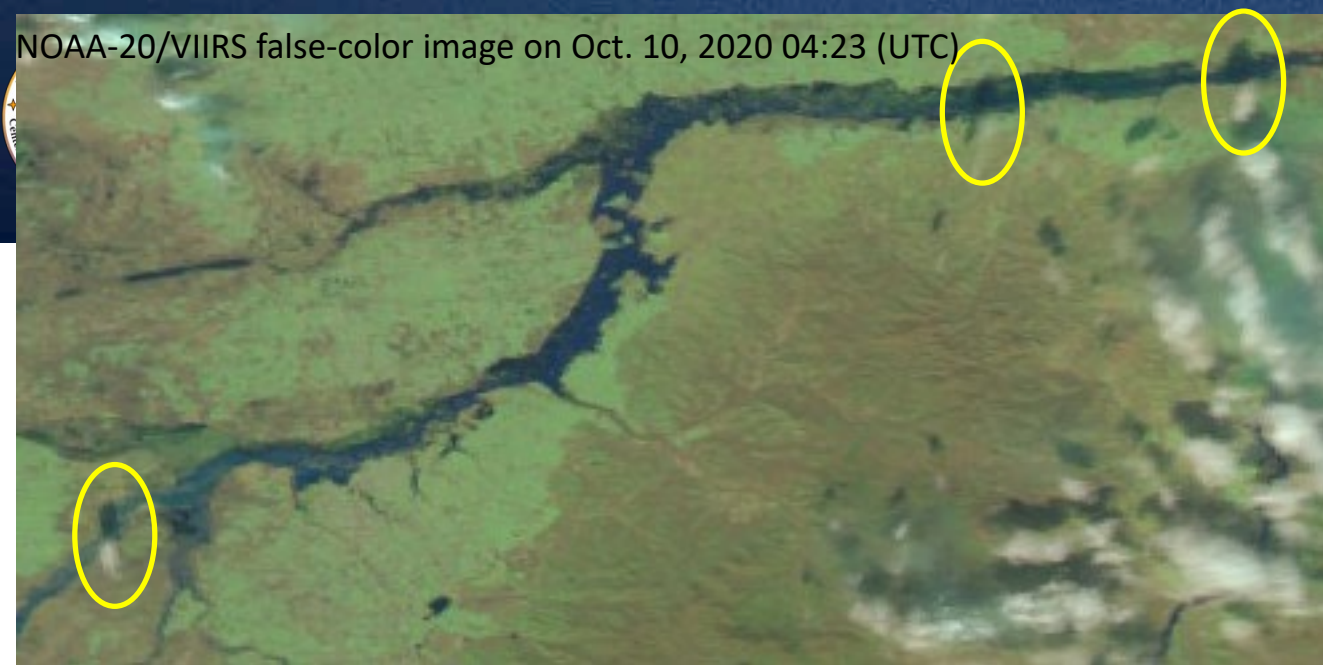


NOAA-20/VIIRS false-color image on Oct.
10, 2020 04:23 (UTC)



NOAA-20/VIIRS flood map from T4
process on Oct. 10, 2020 04:23 (UTC)

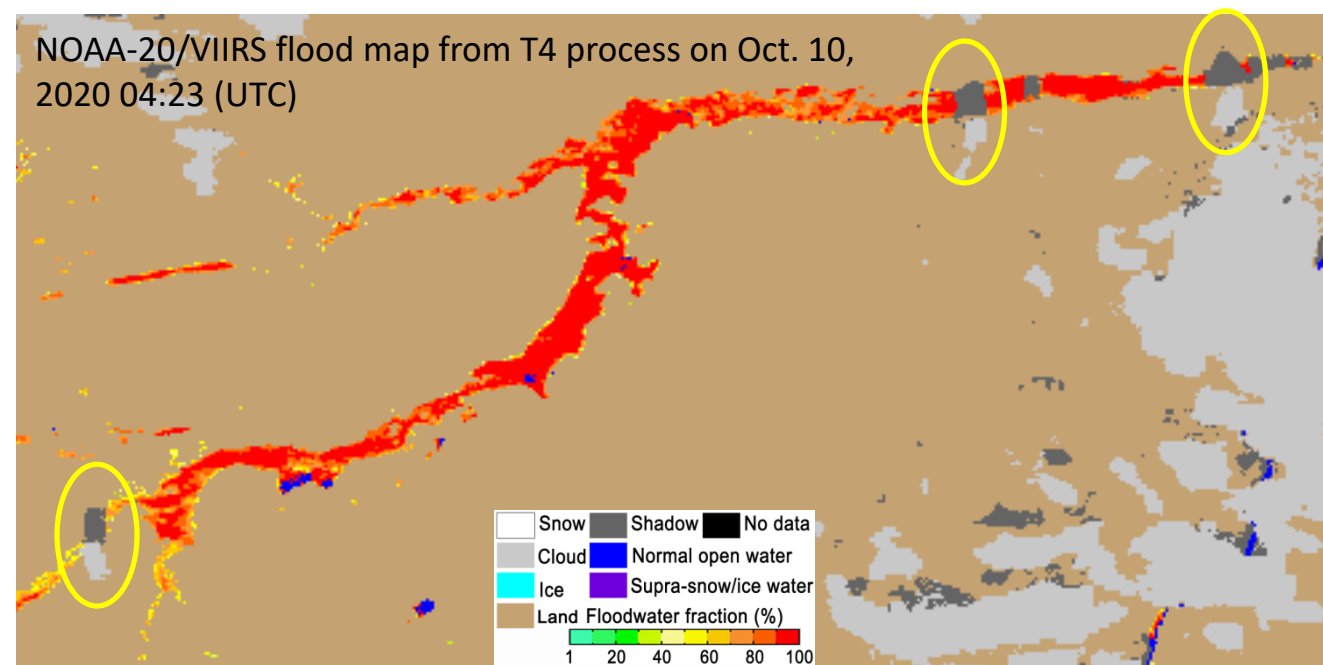
NOAA-20/VIIRS false-color image on Oct. 10, 2020 04:23 (UTC)

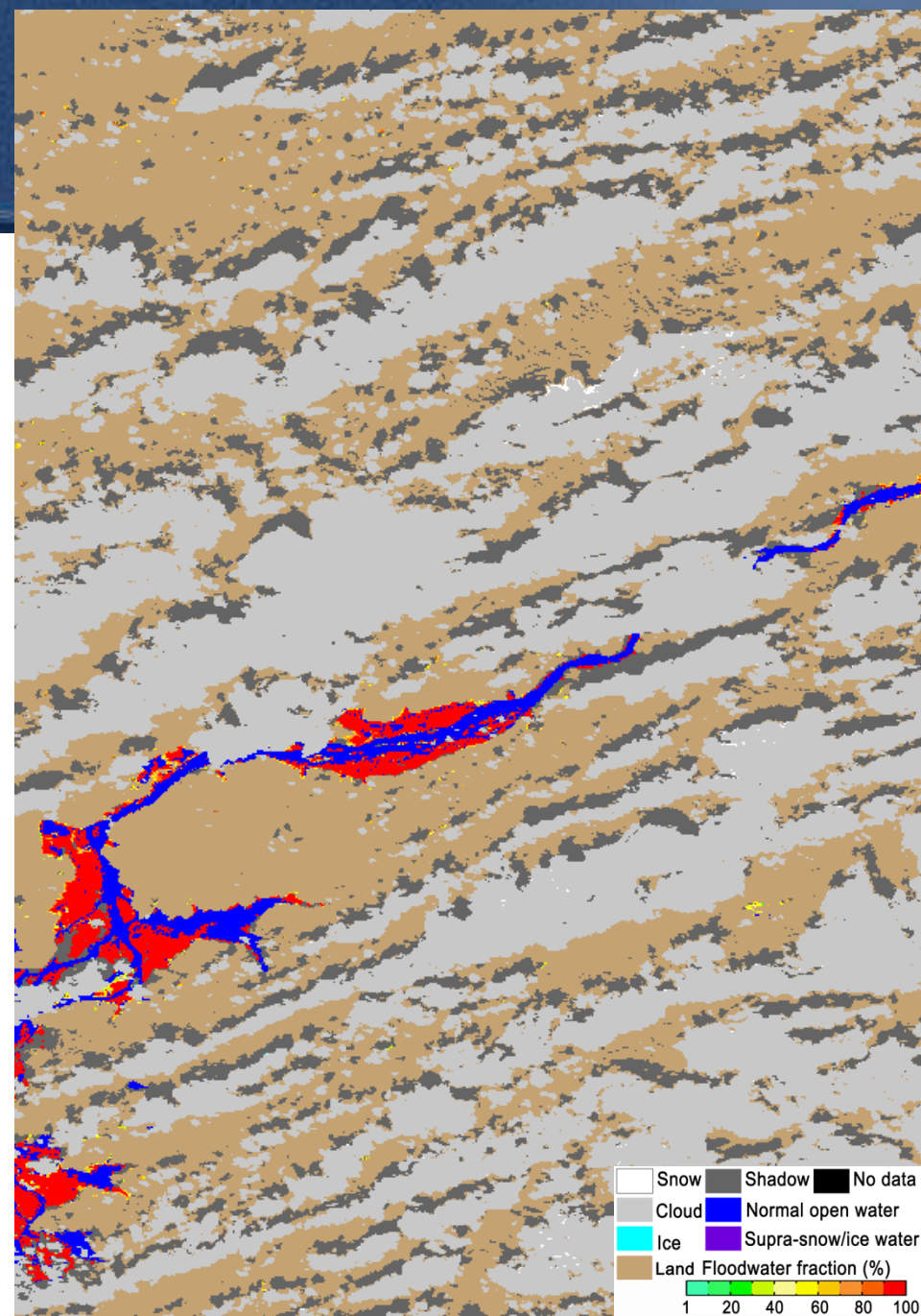
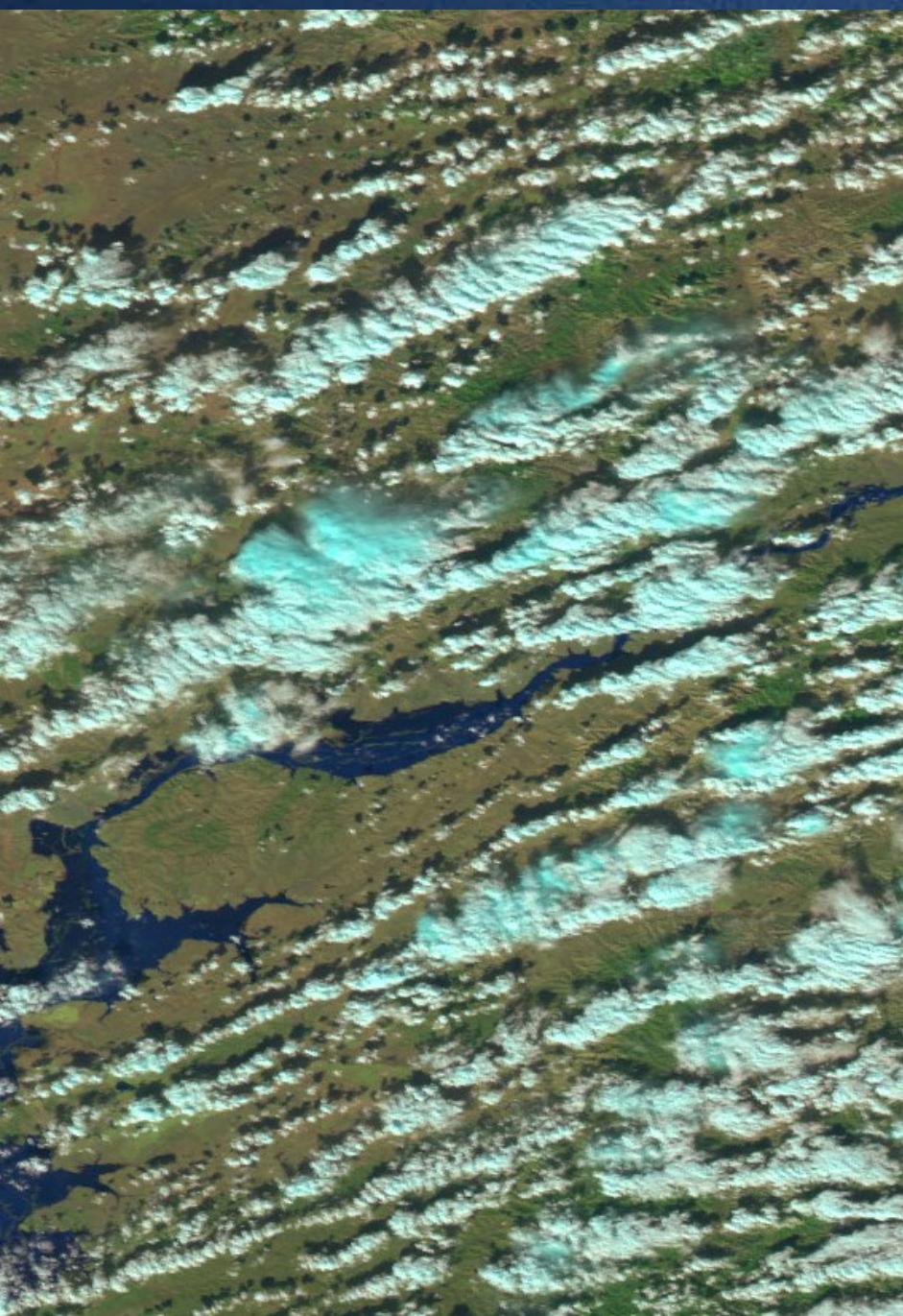


Cloud shadow removal

- Accurate discrimination between cloud shadows and floodwater from the test results.
- Some shadows cast by thin clouds may not be removed due to the uncertainty of cloud detection and cloud height estimation.

NOAA-20/VIIRS flood map from T4 process on Oct. 10, 2020 04:23 (UTC)





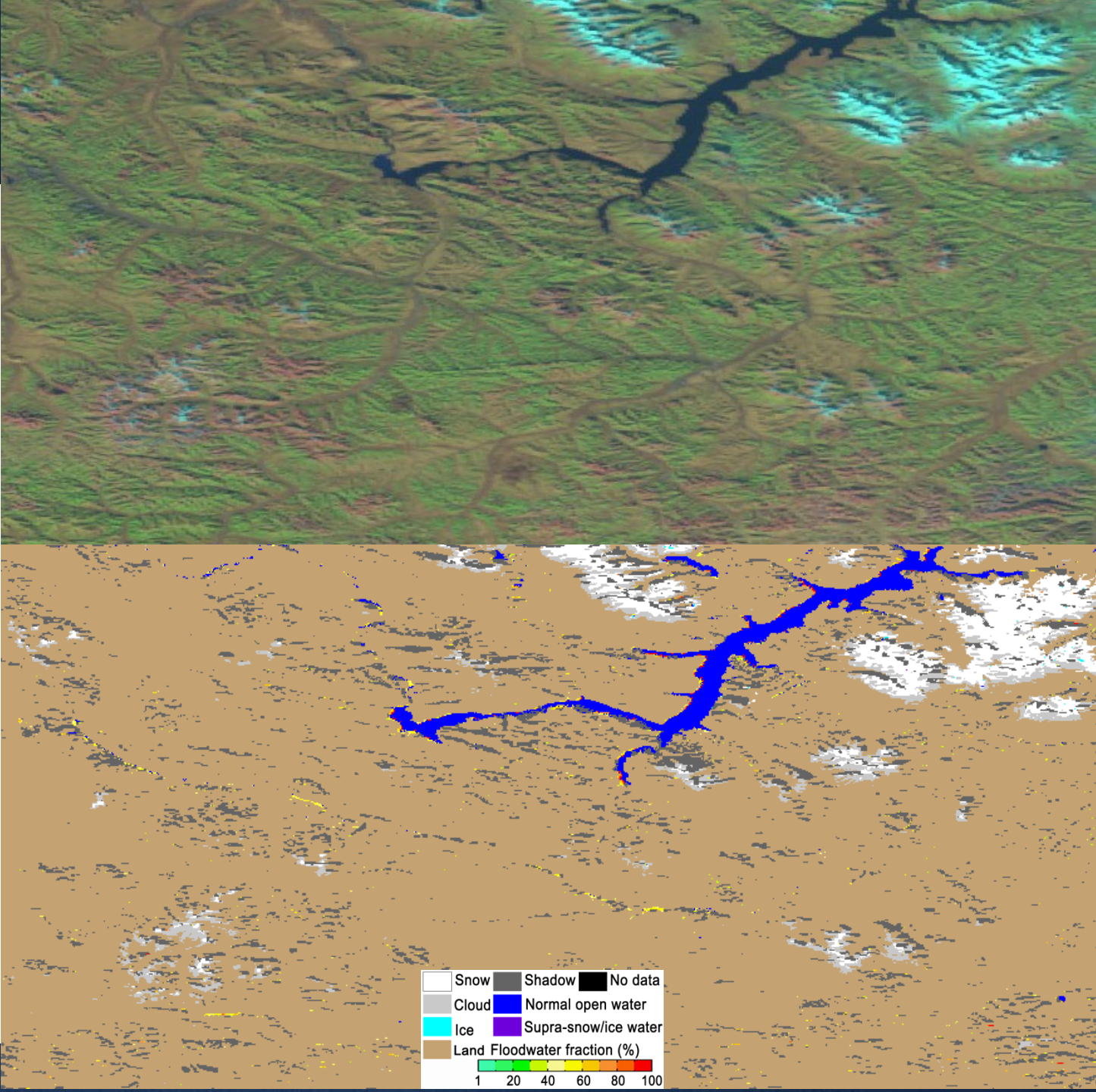
NOAA-20/VIIIRS false-color image on Oct. 13, 2020 03:27 (UTC)

NOAA-20/VIIIRS flood map on Oct. 13, 2020 03:27 (UTC)

Terrain shadow removal

NOAA-20/VIIRS false-color image on Oct. 11, 2020 03:18 (UTC)

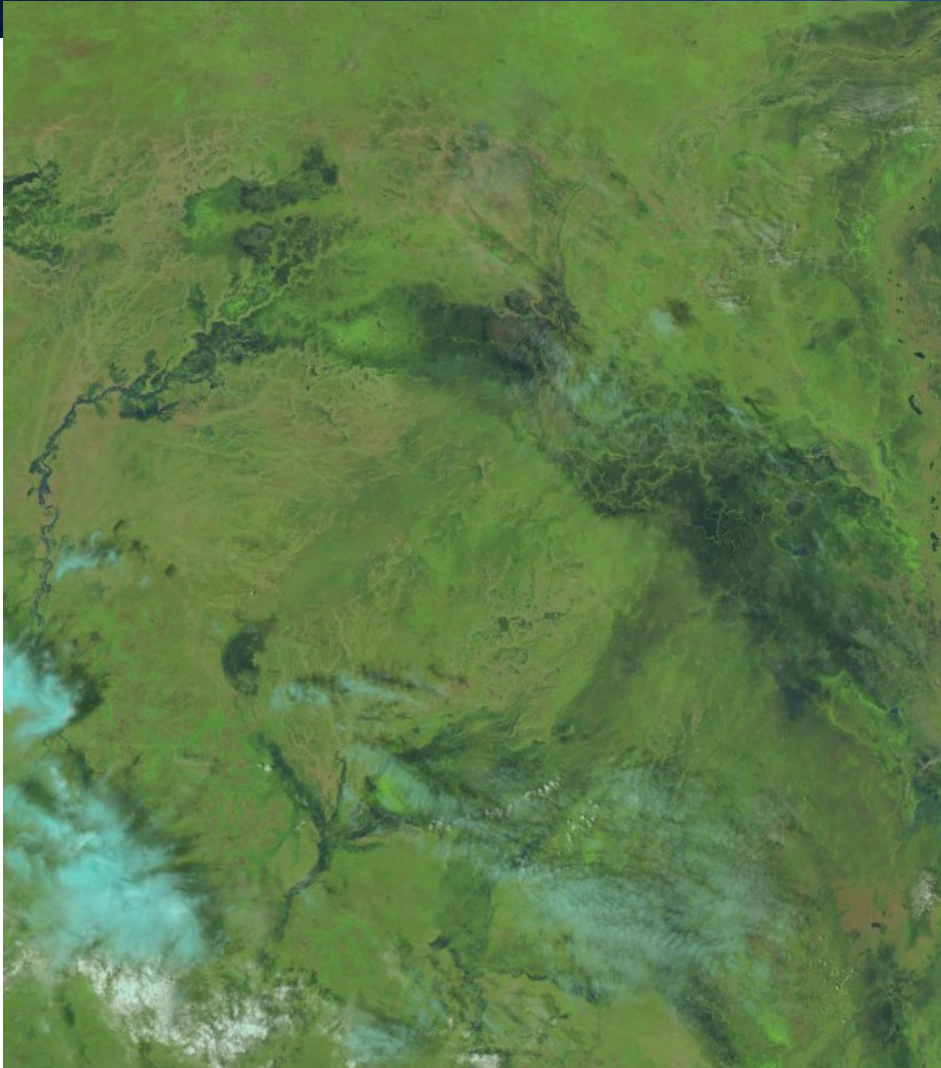
- Accurate discrimination between terrain shadows and floodwater



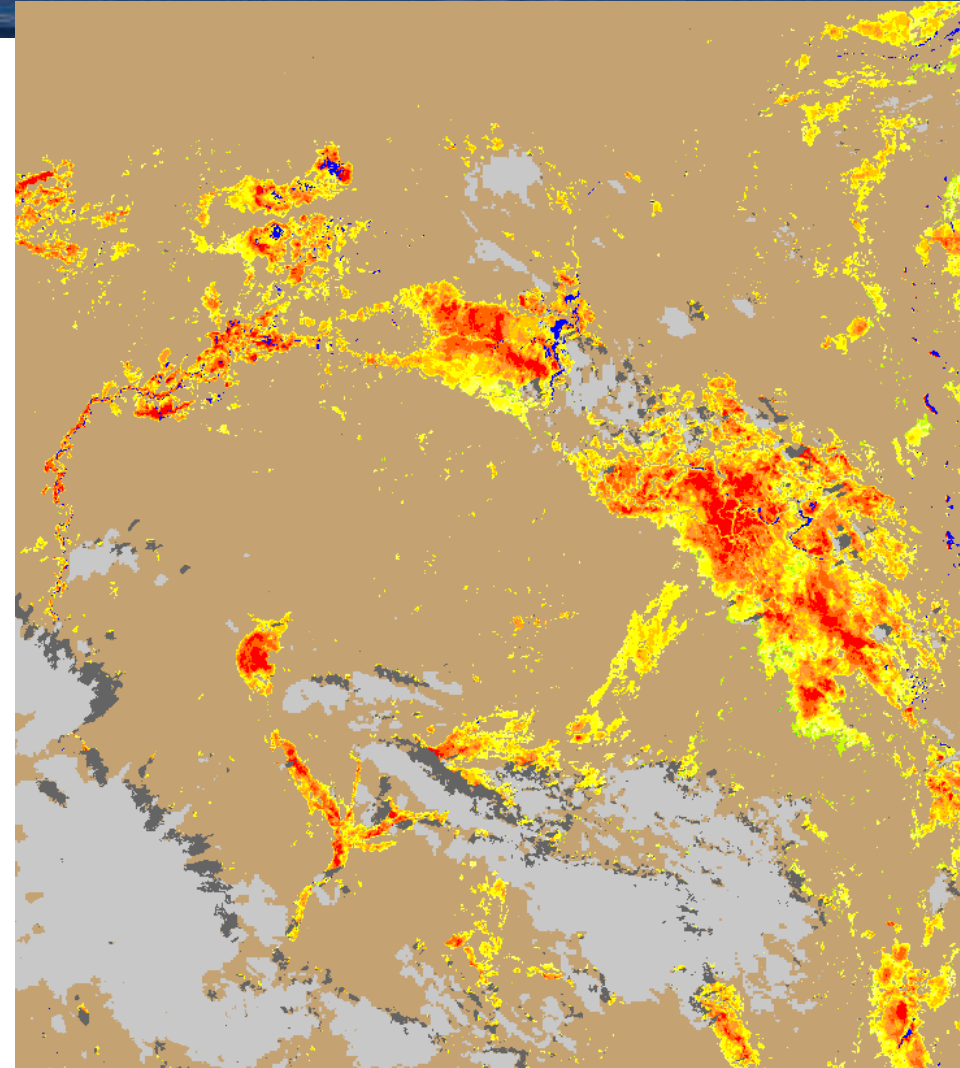
NOAA-20/VIIRS flood map from T4 process on Oct. 11, 2020 03:18 (UTC)



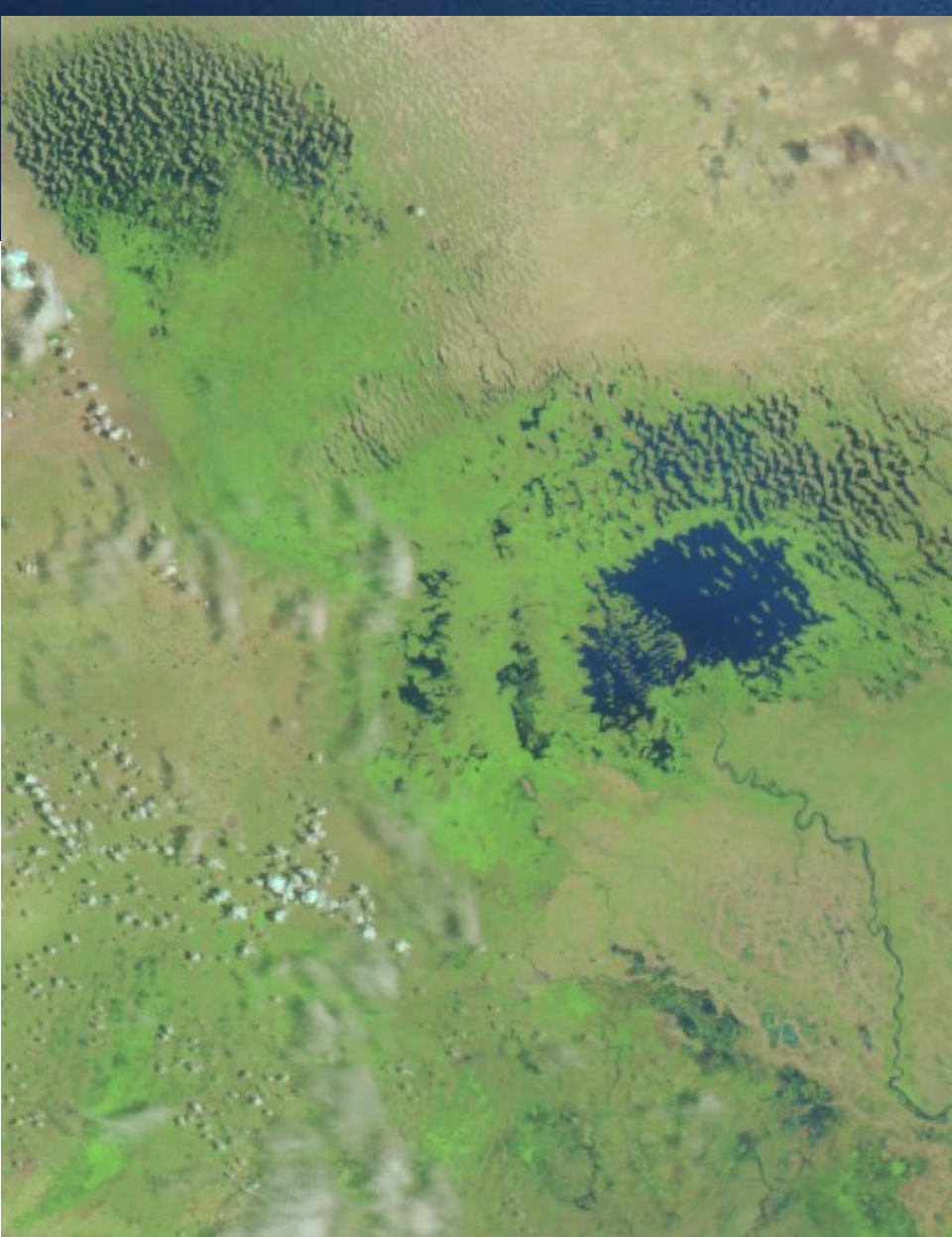
Wetland & Flooding



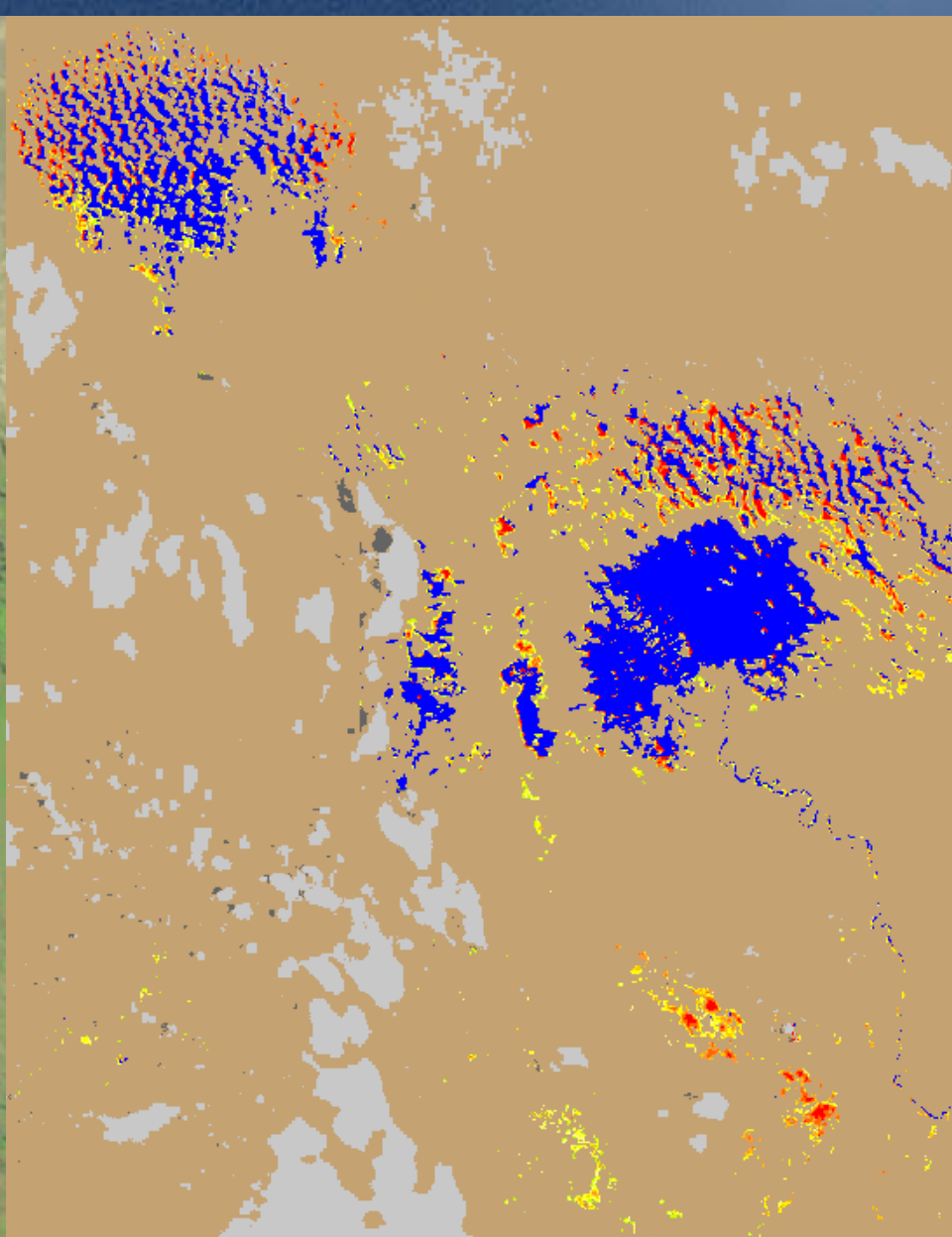
NOAA-20/VIIRS false-color image on Oct. 09, 2020
11:17 (UTC)



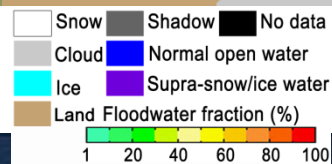
NOAA-20/VIIRS flood map from T4
process on Oct. 09, 2020 11:17 (UTC)



NOAA-20/VIIIRS false-color image on Oct. 09, 2020
12:59 (UTC)



NOAA-20/VIIIRS flood map from T4
process on Oct. 09, 2020 12:59 (UTC)

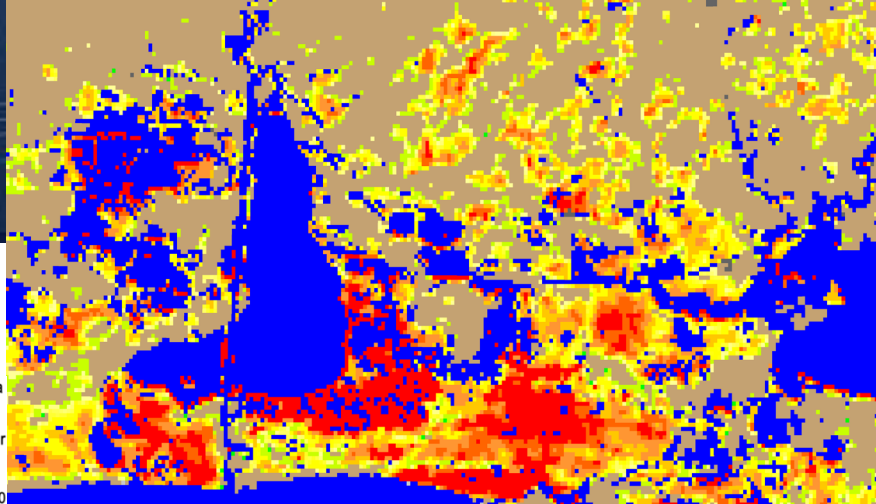
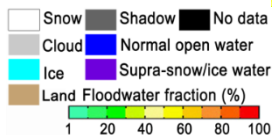


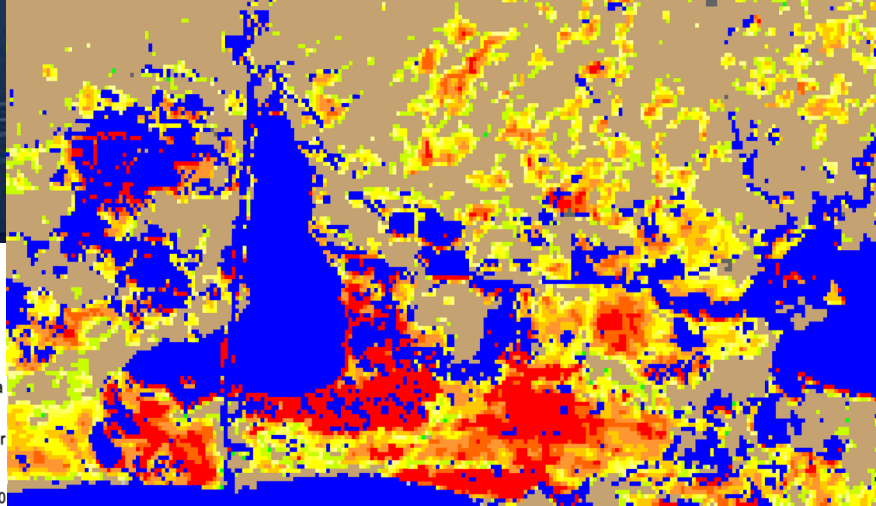


Validation with Sentinel-2 images

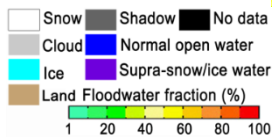
Suomi-NPP/VIIRS flood map on Oct. 10, 2020

Sentinel-2 image on Oct. 10, 2020

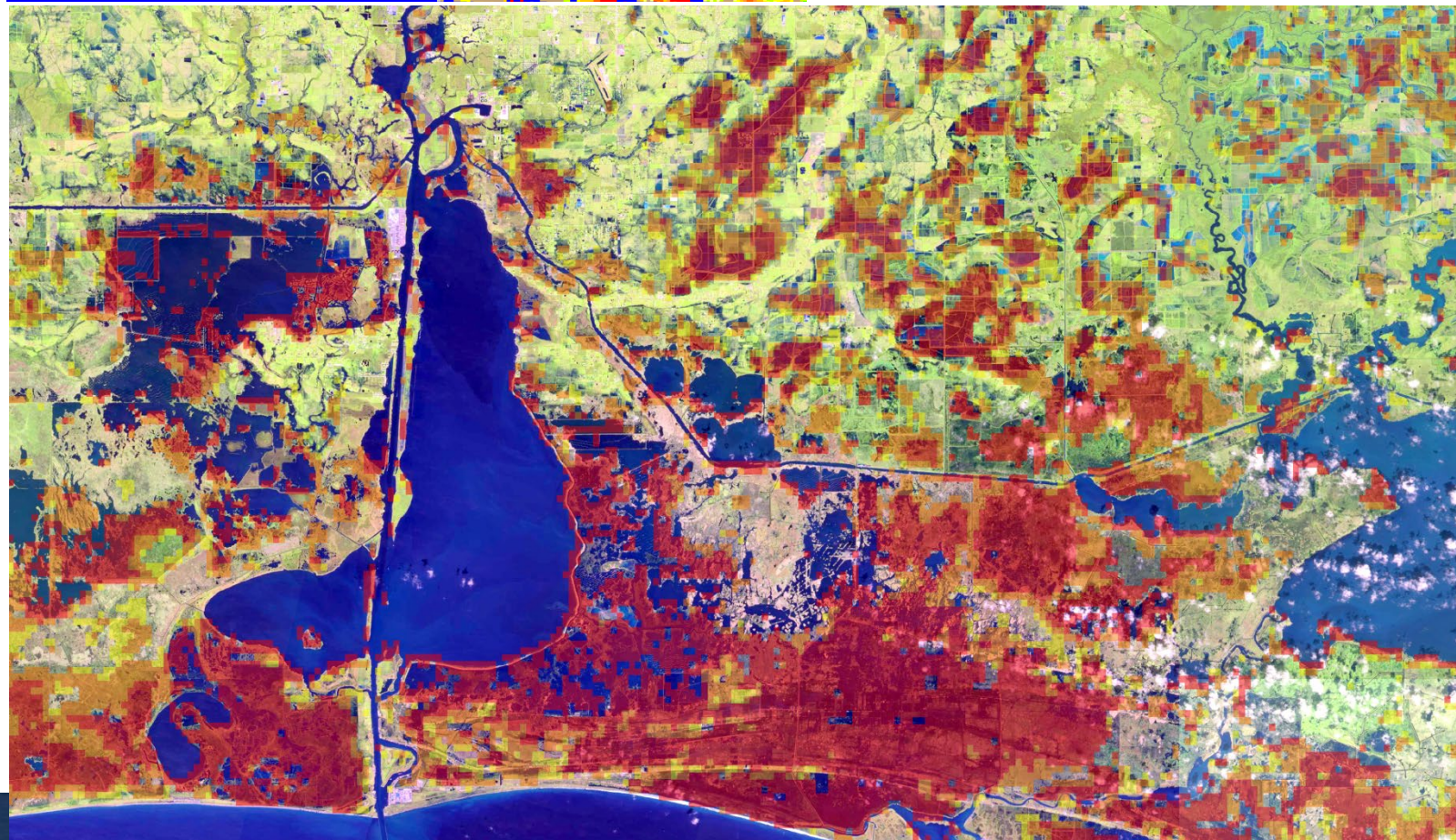




Suomi-NPP/VIIRS flood map
on Oct. 10, 2020

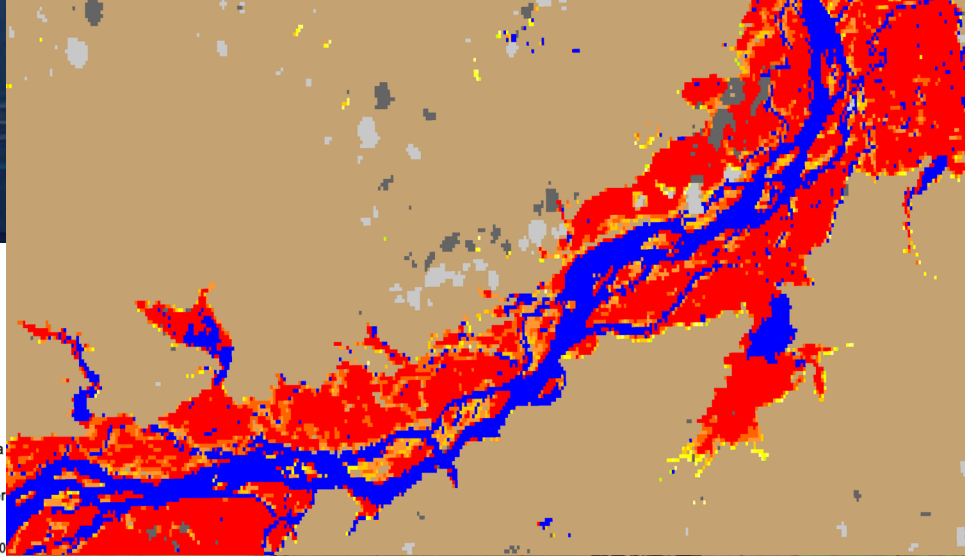


Sentinel-2 image with VIIRS
floodwater overlapped on Oct.
10, 2020





NOAA-20/VIIRS flood map on Oct. 13, 2020

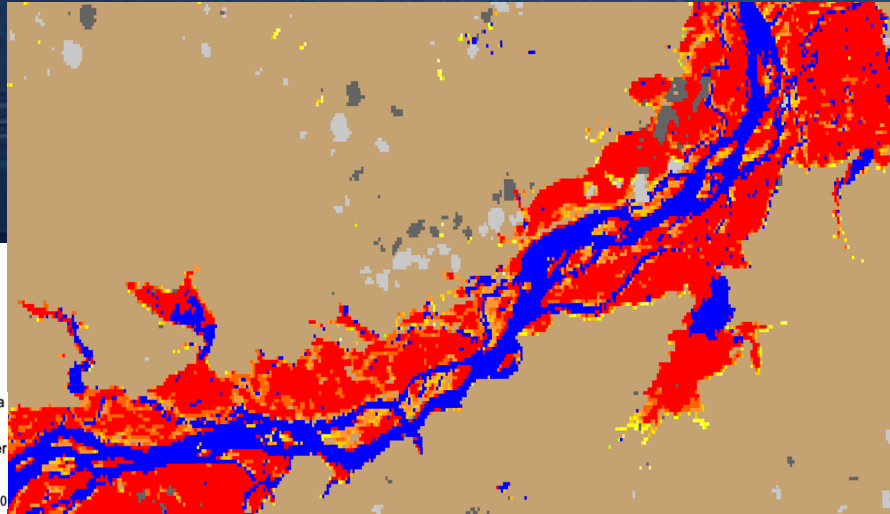


Sentinel-2 image on Oct. 1, 2020

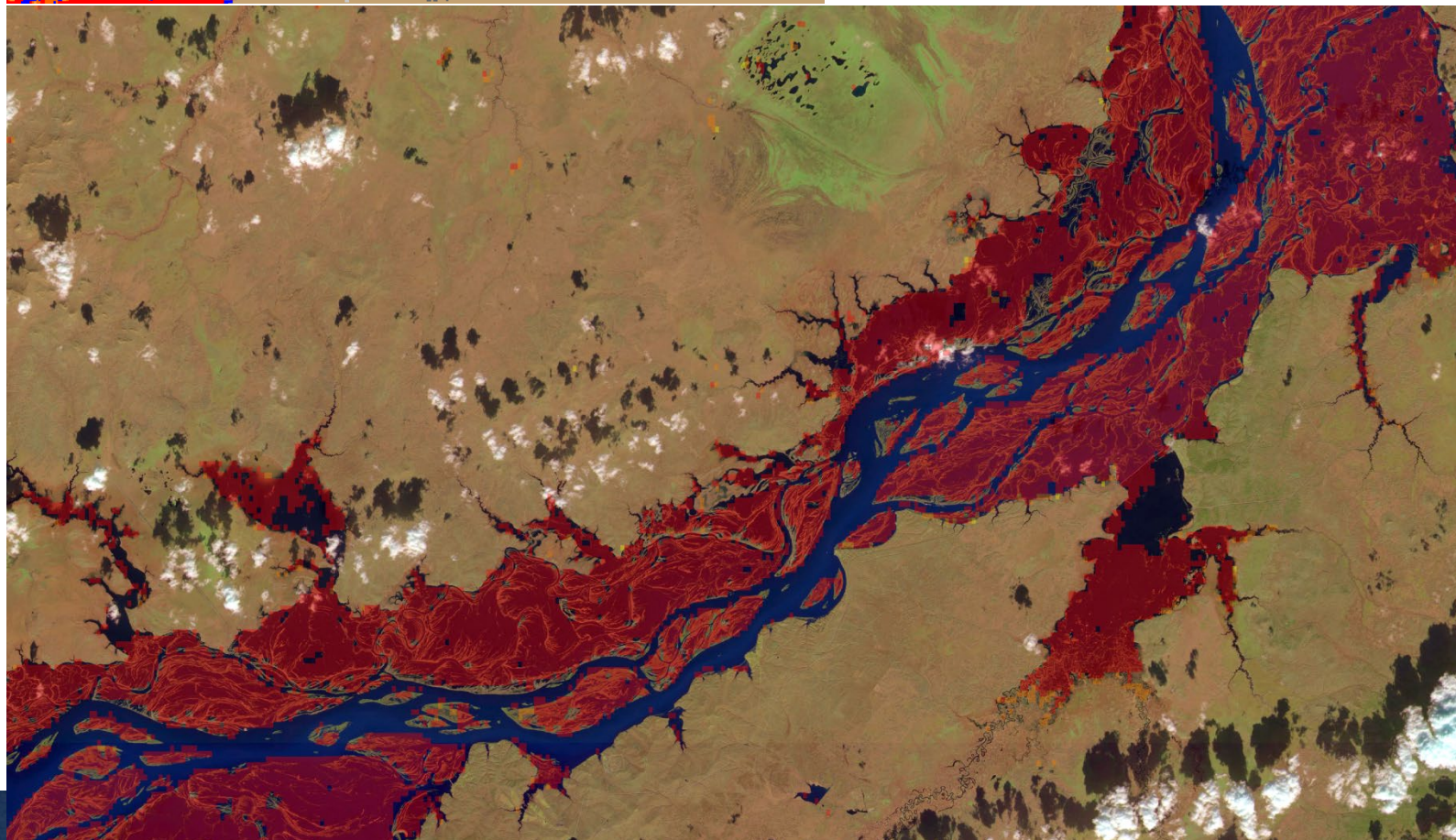




NOAA-20/VIIIRS flood map on
Oct. 13, 2020

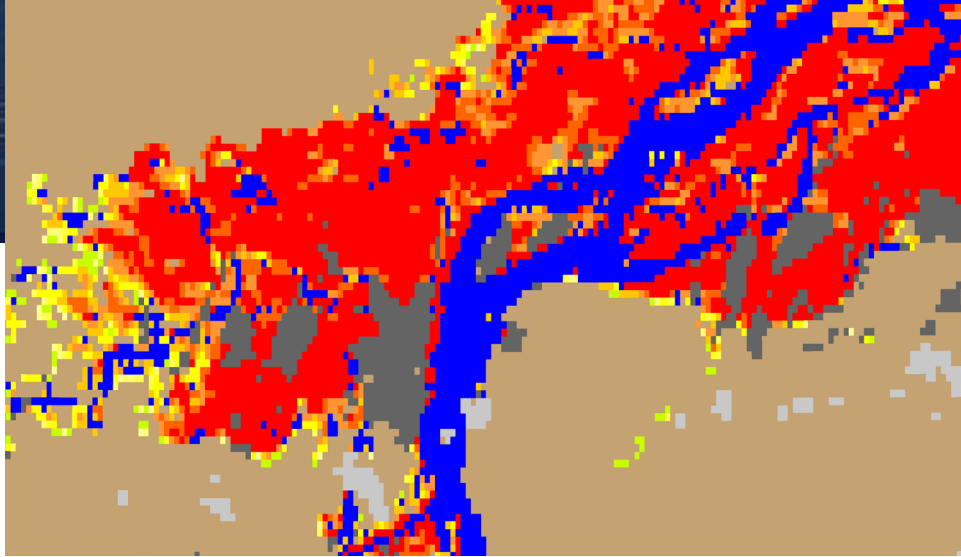


Sentinel-2 image with VIIIRS
floodwater overlapped on
Oct. 13, 2020

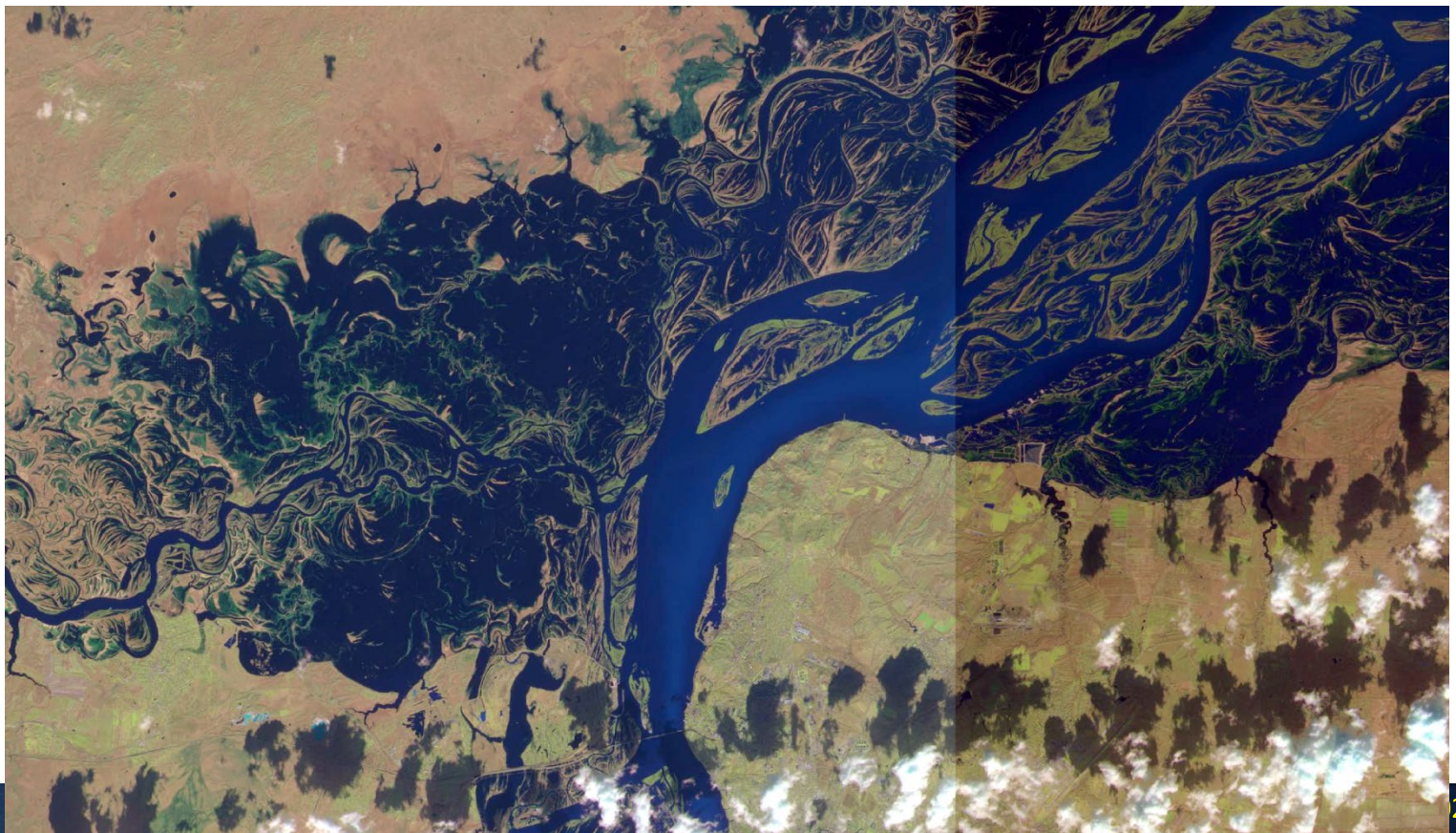
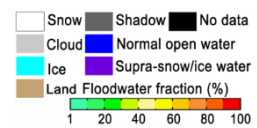




NOAA-20/VIIRS flood map on Oct. 13, 2020

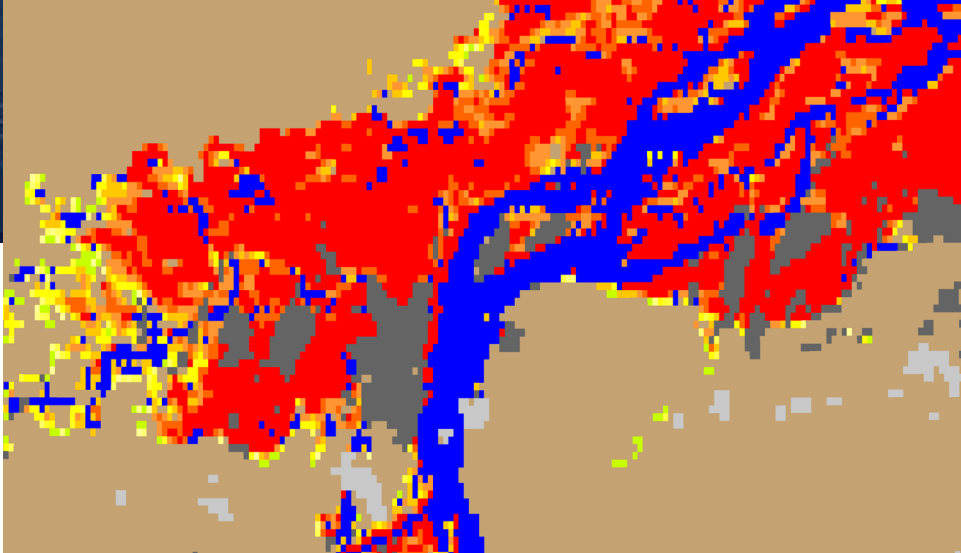


Sentinel-2 image on Oct. 13, 2020

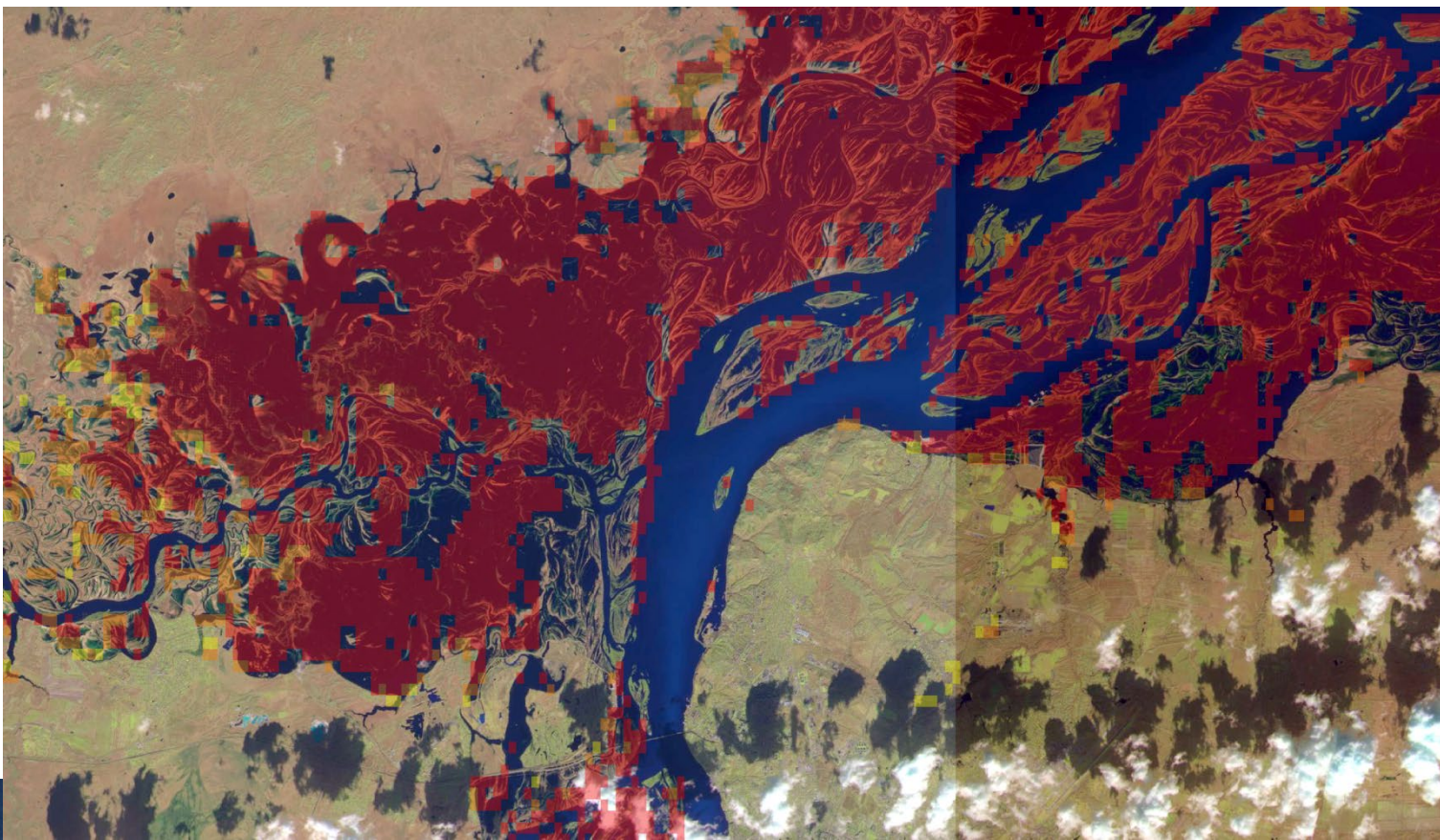
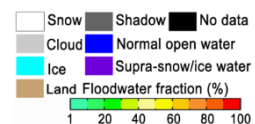




NOAA-20/VIIRS flood map on
Oct. 13, 2020



Sentinel-2 image with VIIRS
floodwater overlapped on Oct.
13, 2020

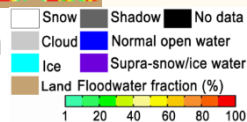
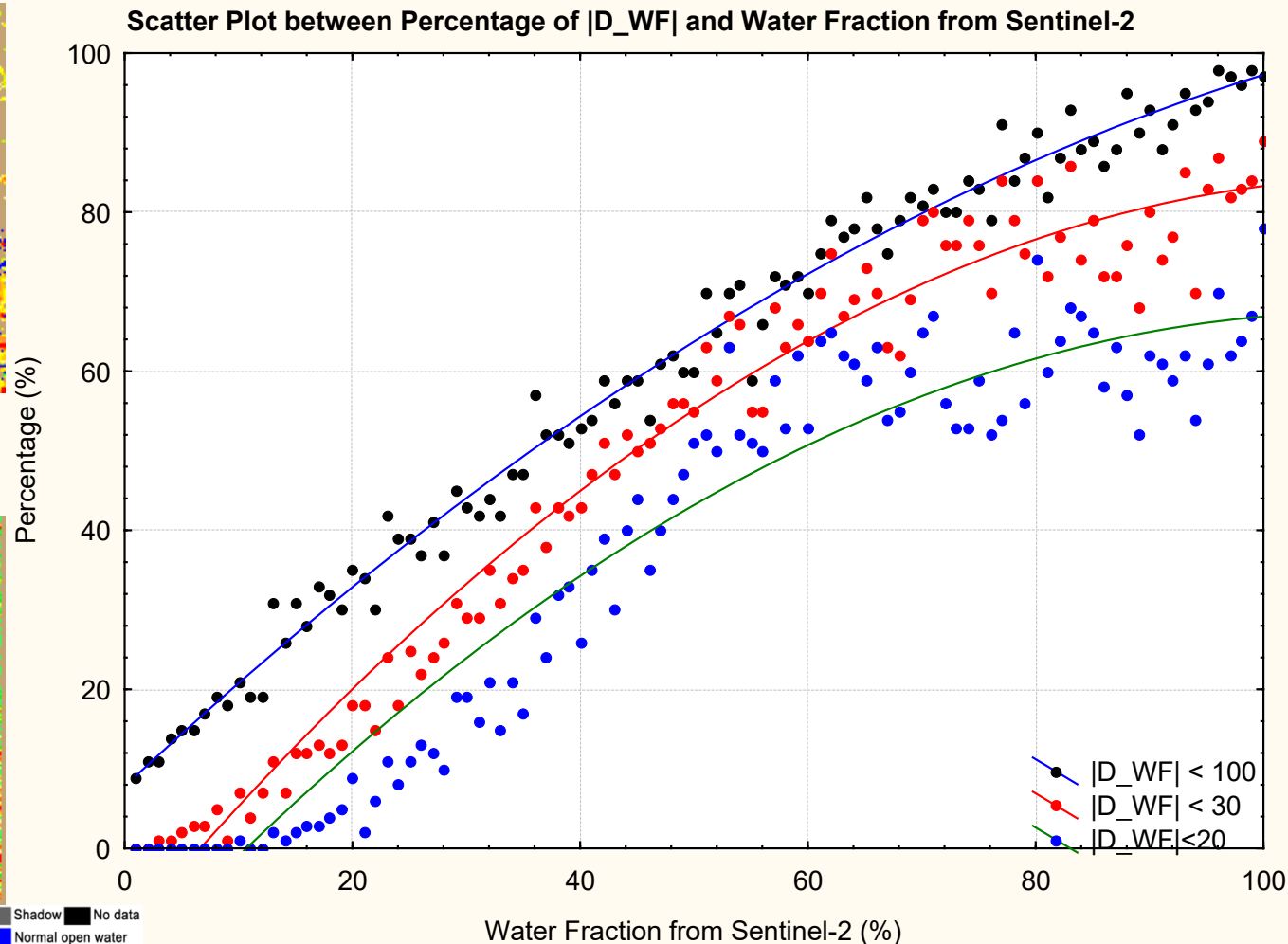
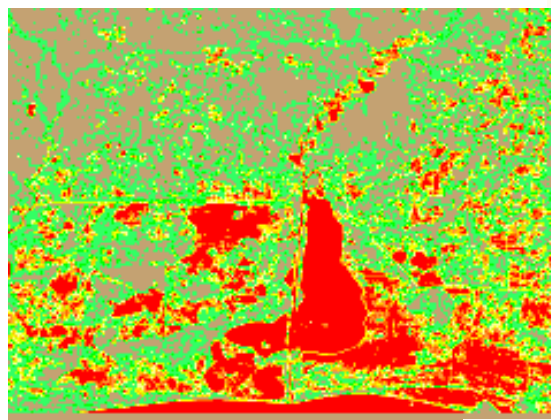
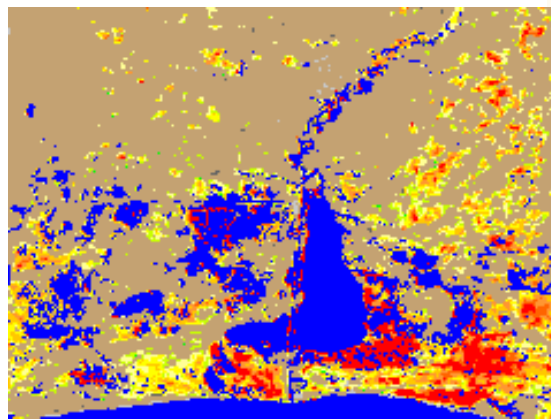




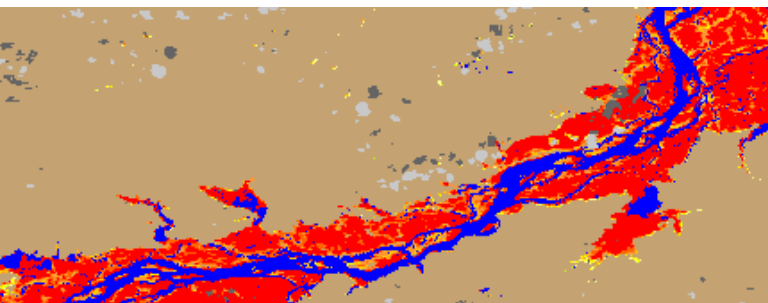
Algorithm Performance – Quantitative Analysis

- About 10 Sentinel-2 images were used to quantitatively validate the VIIRS water detection and water fraction retrieval results.
- $|D_WF|$, defined as the absolute water fraction difference between VIIRS and Sentinel-2, is used to evaluate the results by calculating the percentage of flood pixel with $|D_WF|$ in different range:
 - $P_{|D_WF| \leq 100}$: Percentage of flood pixels both detected in VIIRS and Sentinel-2 image ignoring the water fraction difference
 - $P_{|D_WF| \leq 20}$: Percentage of flood pixels both detected in VIIRS and Sentinel-2 images with $|D_WF|$ less than 20%
 - $P_{|D_WF| \leq 30}$: Percentage of flood pixels both detected in VIIRS and Sentinel-2 images with $|D_WF|$ less than 30%

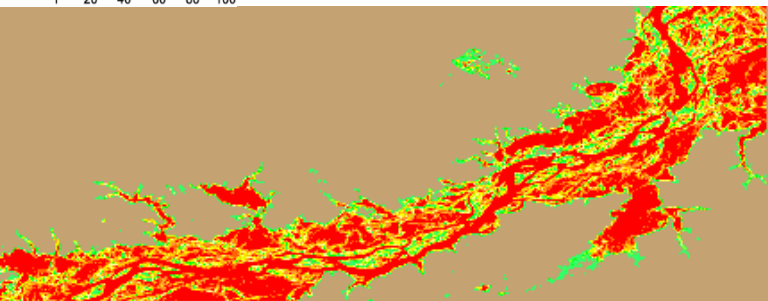
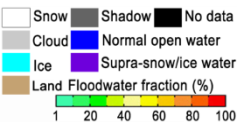
Quantitative Analysis



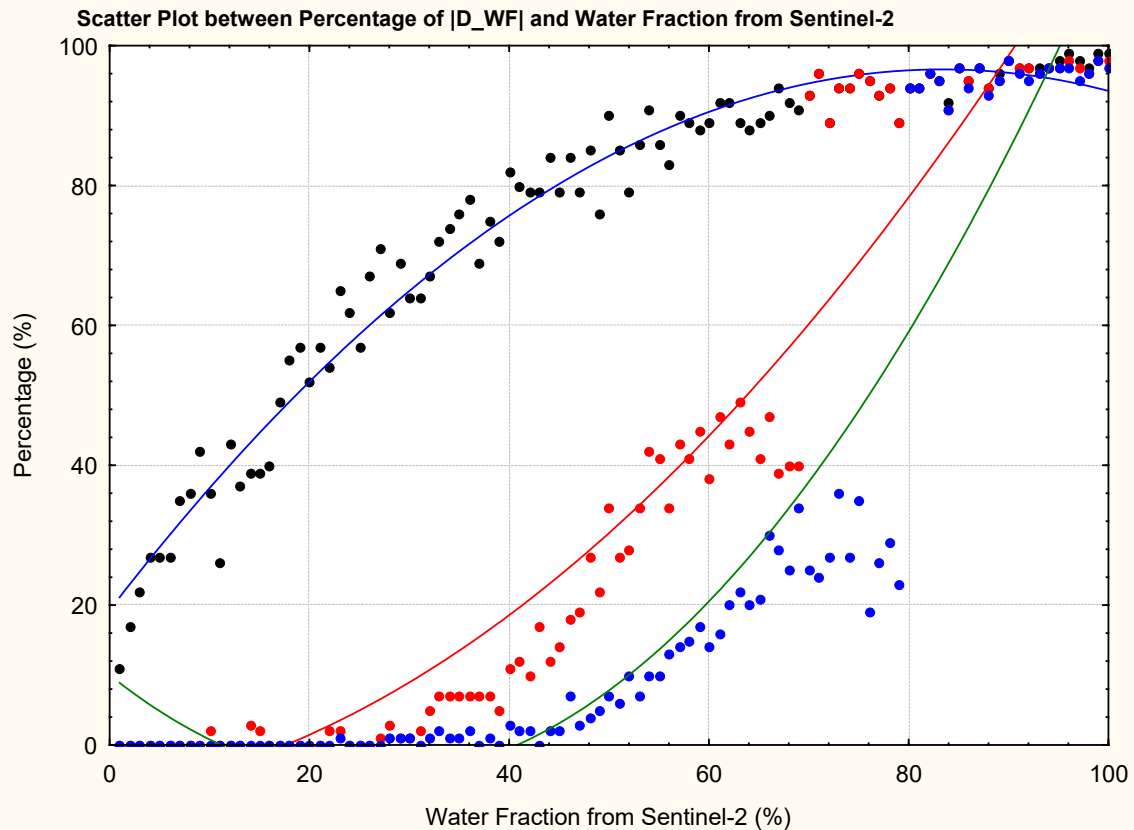
Quantitative Analysis



VIIRS flood map on Oct. 13, 2020



Sentinel-2 water fraction map on Oct. 13, 2020





Issues

- Some cloud shadows (less than 10%), mainly cast by thin clouds, are not removed completely from the VIIRS flood maps due to the uncertainty of cloud detection and cloud height estimate.
 - The cirrus channel may be integrated in future for better shadow removal from cirrus clouds.
- Some terrain shadows (less than 5%) are not removed completely from VIIRS flood maps.
- Occasionally, some muddy land can be confused with small-fraction floodwater pixels from the minor flood detection algorithm.
- Small-fraction water pixels are not detected well with larger uncertainty in water fraction retrieval.



Summary

- The VIIRS flood maps from T4 process have been validated in different means:
 - Outputs are correct in formats and contents
 - Visual inspection indicates consistent results to the VIIRS false-color images
 - Comparison with high-resolution Sentinel-2 images show accurate detection.
 - Quantitative analysis indicates high-quality water detection and water fraction retrieval.
- Overall, the VFM algorithm performs reliably and consistently in global flood detection and floodwater fraction retrieval. Despite of some minor issues, the product meets the requirements for operations.



ARR Outline

- Introduction
- UTRR Risks & Actions
- Requirements
- Algorithm Validation
- Software Architecture
- Delivered Algorithm Package
- ARR Risks & Actions
- Summary



Software Architecture

Presented by:
Peter Keehn and Eric Buzan



Processing Architecture Outline

- **Hardware Environment**
 - GMU Science Development
 - ASSISTT Integration and Unit Testing
 - ASSISTT NRT Integration and Testing
- **Software Description**
 - External-Level Flow
 - Unit Level Flow
- **Data Files**
 - Input Files
 - Static/Ancillary Files
 - Output Files
 - Log Files
 - Resource Files
 - File Formats



Software Architecture - Algorithm

Peter Keehn (PLAIT)



STAR Integration Software and Hardware

ASSISTT PLAIT Integration environment:

Hardware

- CPU Model: 2 x Intel(R) Xeon(R) CPU E5-2643 v3 @ 3.40GHz (24 thread)
- Memory: 251.65 GiB | SWAP: 4.00 GiB

Software

- Operating System: CentOS 7.8.2003
- Compiler: gcc/g++/gfortran version 4.8.5 20150623 (Red Hat 4.8.5-39) (GCC)
- Interpreters:
 - perl 5, version 16, subversion 3 (v5.16.3) built for x86_64-linux-thread-multi
 - Python 2.7.5
 - Python 3.6.8



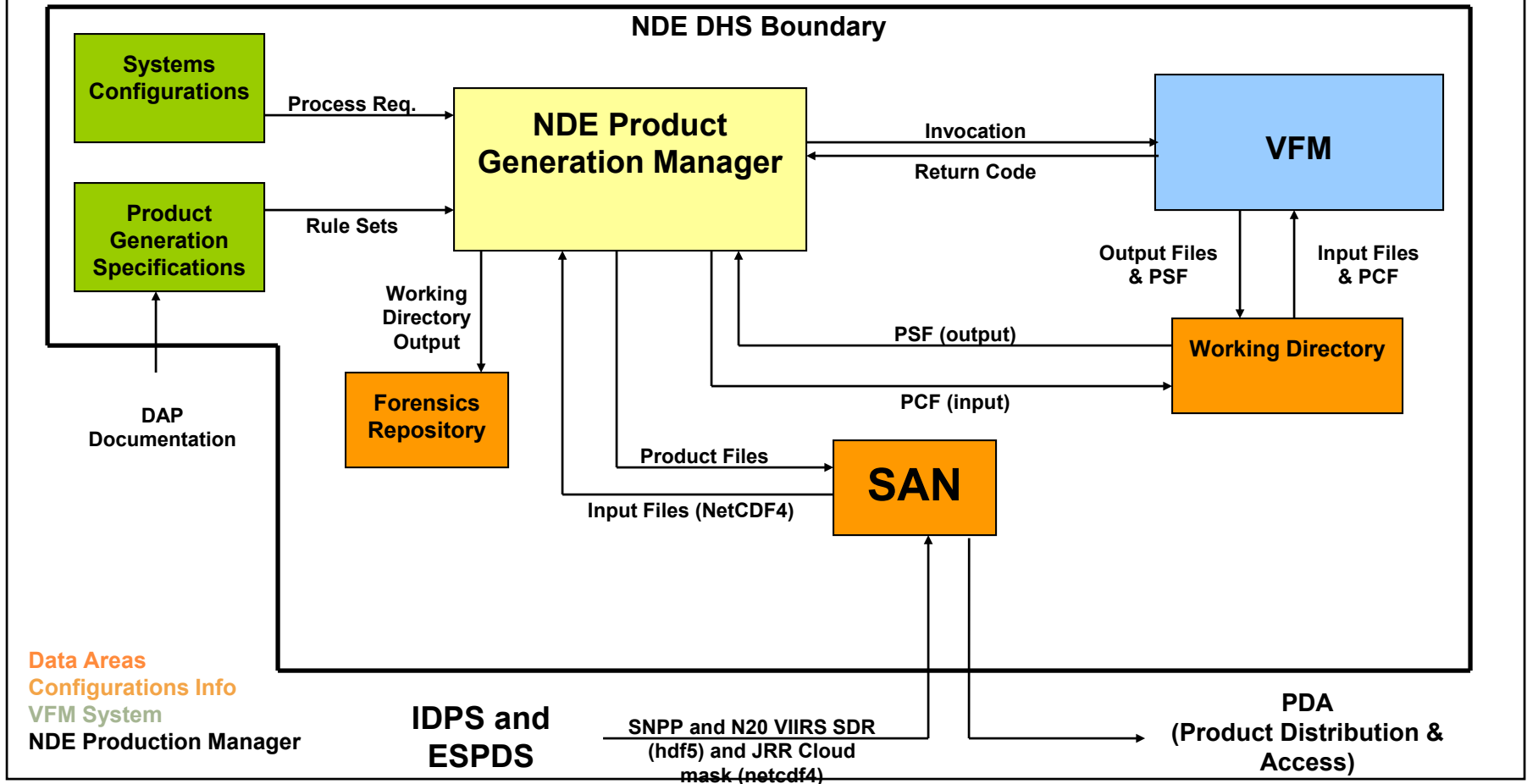
Software/Compilers

- GNU C/C++ version 4.8x
- HDF4
- HDF5
- NetCDF4
- ms2gt
- Code Size
 - C++ code: 42862 lines
 - shell script: 1052 lines
 - Python: 362 lines



External Interfaces

VFM External Interfaces





External Input Files

VFM External Dynamic Input Data – Granule Processing

Input File	Name Pattern	Source	File Size	Format
VIIRS Geo. File NPP or J01	GITCO_j01_d20200225_t1909150_e1910395_b11768_c20200226160606186943_nobc_op s.h5	OSPO PDA	Varies ~110 MB	H5
VIIRS SVI01 NPP or J01	SVI01_j01_d20200225_t1909150_e1910395_b11768_c20200226160556032790_nobc_op s.h5	OSPO PDA	Varies ~30 MB	H5
VIIRS SVI02 NPP or J01	SVI02_j01_d20200225_t1909150_e1910395_b11768_c20200226160559483258_nobc_op s.h5	OSPO PDA	Varies ~30 MB	H5
VIIRS SVI03 NPP or J01	SVI03_j01_d20200225_t1909150_e1910395_b11768_c20200226160600612879_nobc_op s.h5	OSPO PDA	Varies ~30 MB	H5
VIIRS SVI05 NPP or J01	SVI05_j01_d20200225_t1909150_e1910395_b11768_c20200226160534051648_nobc_op s.h5	OSPO PDA	~25 MB	H5
Cloud Mask NPP or J01	JRR- CloudMask_v2r1_j01_s202002251909150_e202002251910395_c202002251935310.nc	OSPO PDA	~17 MB	NetCDF



Static Files

External Static Input Data

Input File	Name Pattern	Source	Format	Used by
Global land cover at 1km resolution	Global_land_cover_IGBP_2017_USGS_types.raw	VIIRS surface type and AVHRR land cover 2000	raw	Flood Detection
Global Digital Elevation Model at 375-m resolution	Global_DEM375m_W180_W090_N90_S90.raw,... Global_DEM375m_E090_E180_N90_S90.raw	SRTM/DEM ASTER DEM	raw	Flood Detection
Sun-glitter mask	Sun_Glitter_mask_005.dat		raw	Flood Detection
Global land/sea mask at 1km resolution	Lw_geo_2001001_v03m_1km.raw		raw	Flood Detection, Composition
Global water mask	VIIRS_Global_MOD44W_Water_Mask.raw	MOD44W, ESA's global water mask, NLCD 2006	raw	Flood Detection, Composition
Land/sea surface temperature 16-day climatology at 5km resolution	AQUA_Daytime_LST_SST_Climatology_NNN.raw		raw	Flood Detection
Global Albedo monthly climatology at 5km resolution in visible channel	CMG-SMT-POB1_ch1_{Channel number}.raw		raw	Flood Detection
Pre-trained decision trees and tree attribute files	Tree_{tree number}_attr.txt Tree{tree number}_J48graft_{description}.txt		txt	Flood Detection
User AOI definition file: to list the geographic information of each subset	User_AOI_Definition.txt	User defined	txt	Swath Projection, Subset & Mosaic



External Outputs

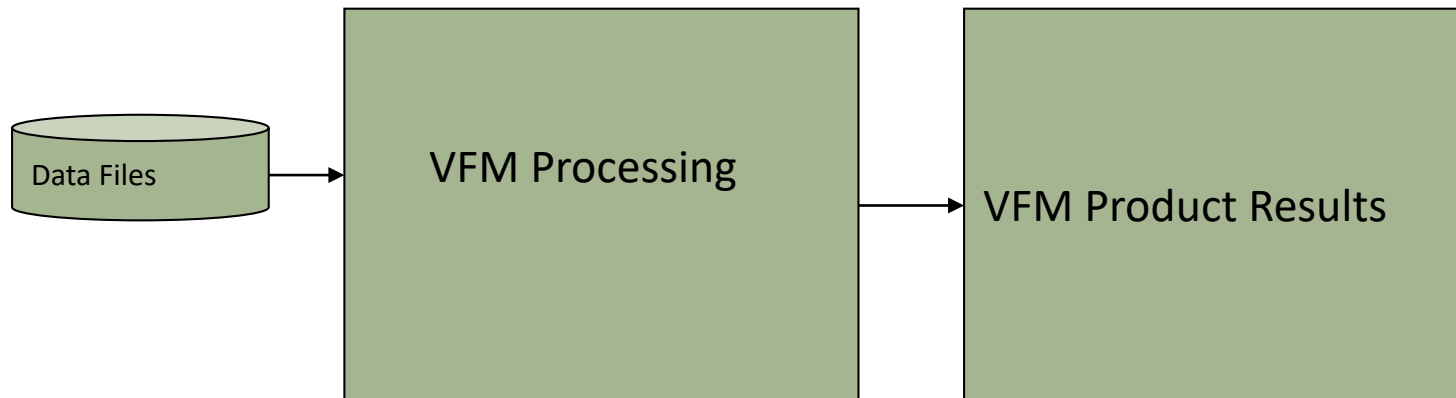
System Level Output Data					
Output File	Name Pattern (all times are in YYYYMMDDHHMMSS)	Archived/Distributed	Update Frequency	Format	File Size
VIIRS Flood Map Granule NPP, J01	VIIRS-Flood-v1r0_<npp n20>_s<start_time>_e<end_time>_c<creation_time>.<nc png tif shapefiles.zip>	All Distributed, Archived Netcdf4	NRT	NetCDF, PNG, GeoTIFF, Shapefile	44 - 292 MB ~ 1 MB 20 - 150 MB 0.1 - 3 MB
VIIRS Flood Map Mosaic 8 NWS Areas NPP J01	VIIRS-Flood-NWS???_v1r0_<npp n20>_s<start_time>_e<end_time>_c<creation_time>.<nc png tif shapefiles.zip>	Distributed	NRT	NetCDF, PNG, GeoTIFF, Shapefile	34 - 127 MB 500 KB 21 MB 0.009 - 1 MB
VIIRS Flood Map 1-Day Composite 136 Global Areas	VIIRS-Flood-1day-GLB???_v1r0_blend_s<start_time>_e<end_time>_c<creation_time>.<nc png tif shapefiles.zip>	All Distributed, Archived Netcdf4	6 hourly	NetCDF, PNG, GeoTIFF, Shapefile	40 MB 0.003 - 1.2MB 19 MB .009 - 10 MB
VIIRS Flood Map 5-Day Composite 136 Global Areas	VIIRS-Flood-5day-GLB???_v1r0_blend_s<start_time>_e<end_time>_c<creation_time>.<nc png tif shapefiles.zip>	All Distributed, Archived Netcdf4	6 hourly	NetCDF, PNG, GeoTIFF, Shapefile	40 MB 0.04 - 2.5 MB 20 MB 0.009 - 23 MB

The daily and 5 day composite outputs for the 136 AOIs will be tarred to have 4 tarballs for the 4 formats, in addition to the individual netcdf4 files.



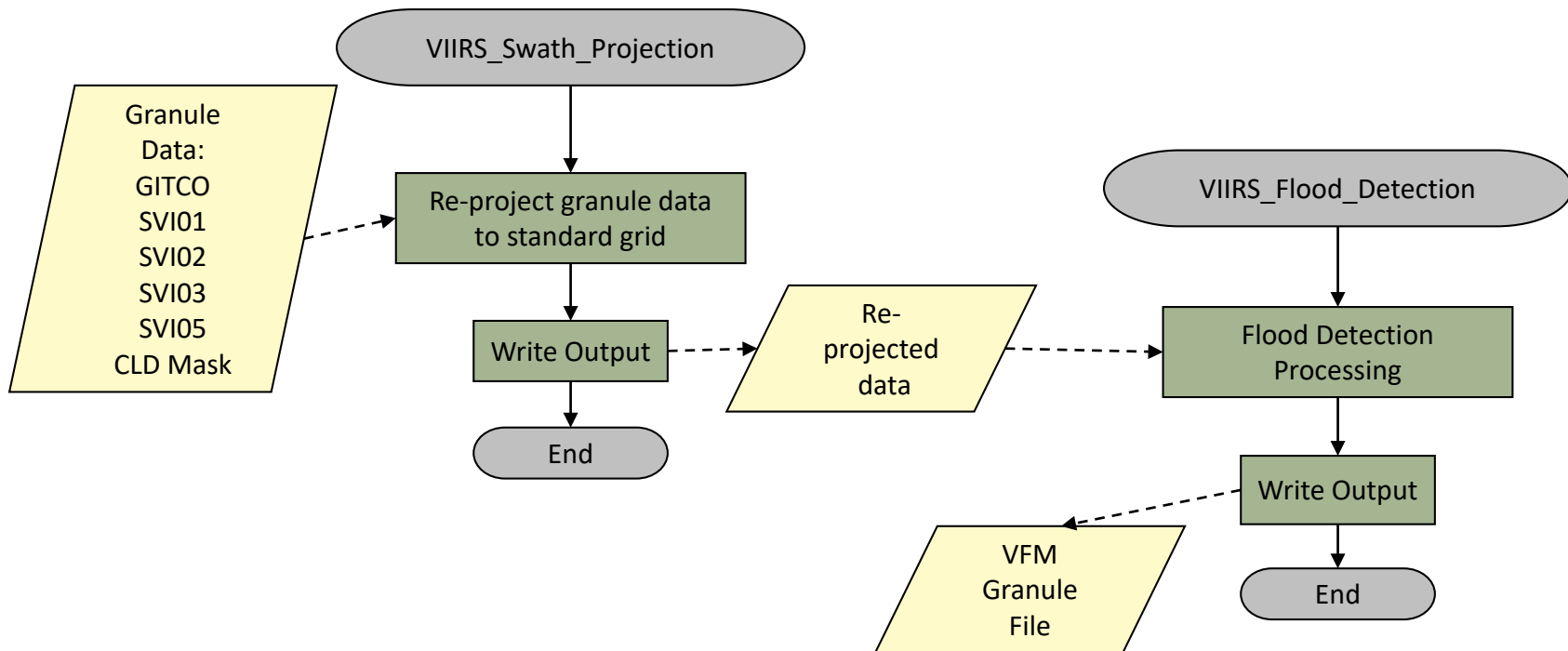
System Flow Chart

Simplified Operational VFM System Diagram



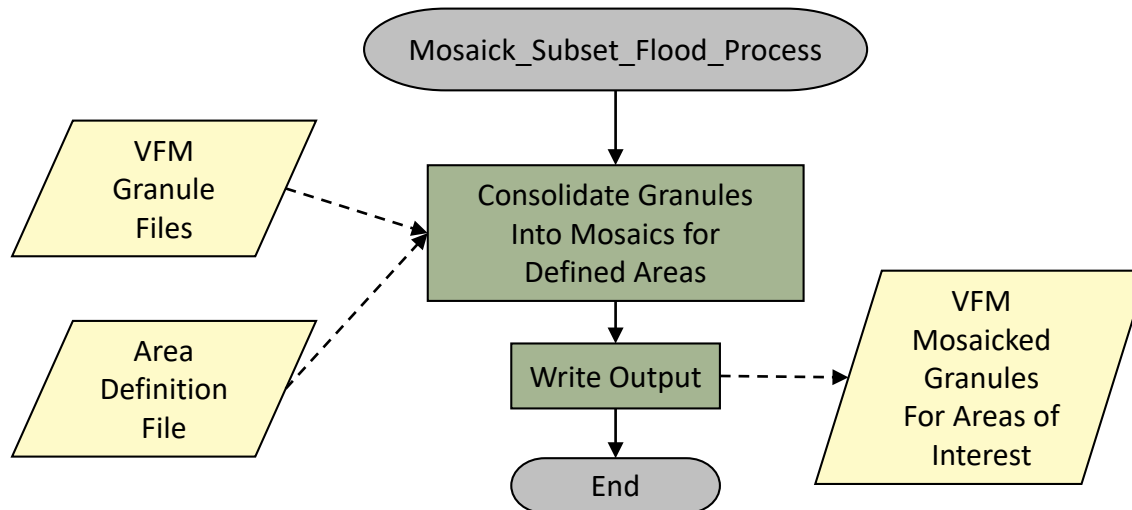


VFM - Granule Flow Chart



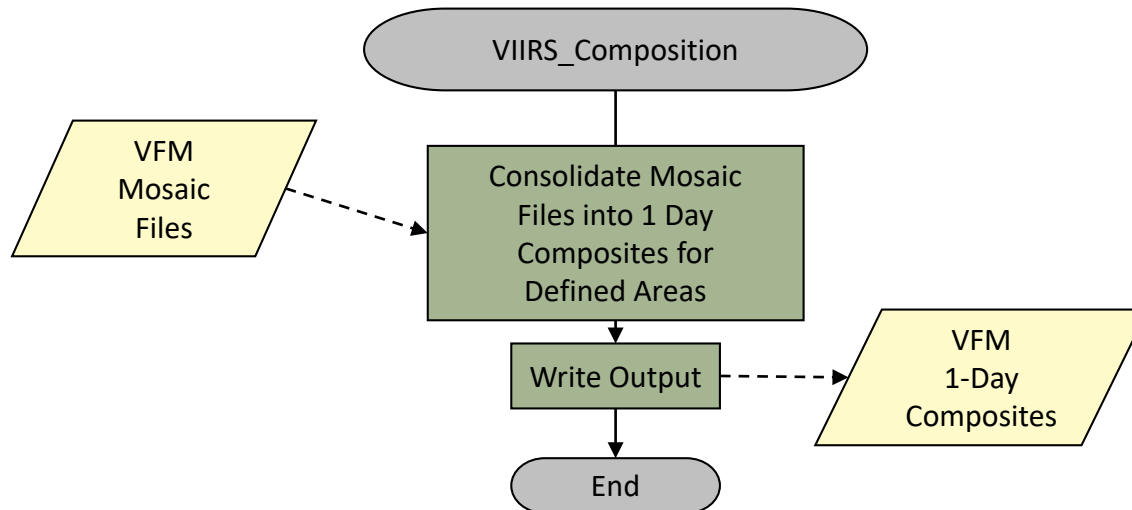


VFM - Mosaic Flow Chart



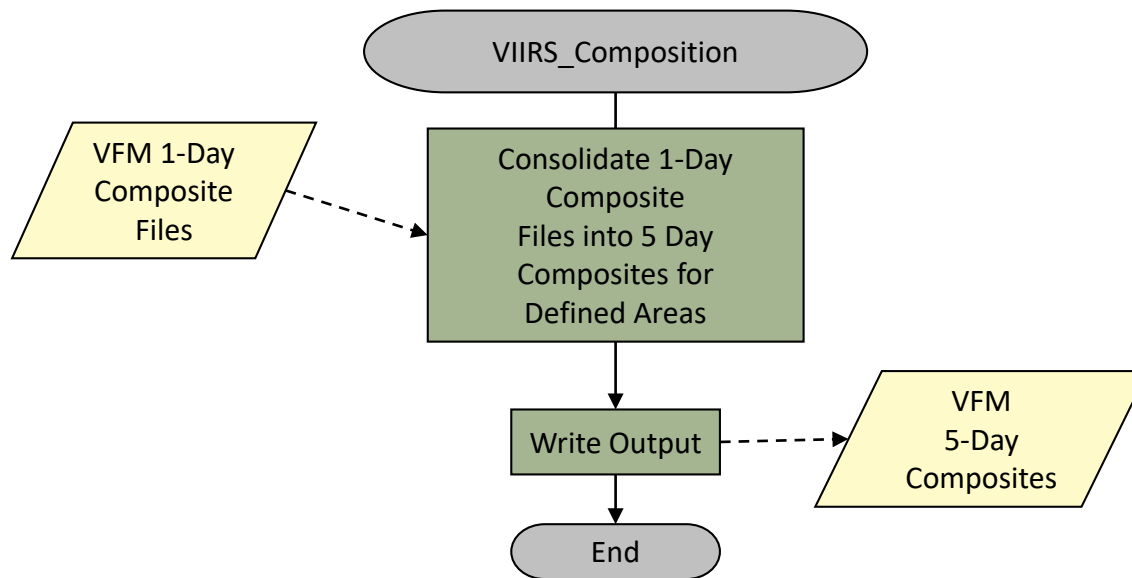


VFM – 1-Day Composite Flow Chart





VFM – 5-Day Composite Flow Chart





VFM Overview: Production Rules

VIIRS Flood Mapping consists of three parts

- Granule processing
 - Generate Flood Mapping results for each granule of VIIRS data
 - NPP and J01
 - Daytime granules only
 - No Polar granules (north or south of ~75 degrees latitude)
- Mosaic processing
 - Combine granule output files corresponding to a given Area Of Interest (AOI)
 - Two defined AOI Lists
 - Global (136 locations)
 - NWS (8 locations)
 - NPP and J01
 - Run every 6 hours for Global AOI
 - Run ASAP for NWS AOI
- Composite processing
 - Combine mosaic output files corresponding to a given Global AOI
 - 1 and 5 day composites
 - Combines NPP and J01 data
 - Run every 6 hours for 1-day composite
 - Run twice daily for 5-day composite



VFM Overview: Outputs

- All output is NetCDF format
 - PNG, GEOTIFF, Shapefile can/will be generated as well from the NetCDF files
- Quality information in each NetCDF file
 - Variable name: QualityFlag
 - Each pixel is flagged as...
 - 0: high quality detection
 - 1: moderate quality detection
 - 2: low quality detection
 - 255: FillValue



Product Monitoring Metadata

- Product Monitoring Metadata has been added to the granule files
 - fillvalue_pixel_percent
 - % of pixels containing the fill value
 - lowquality_pixel_percent
 - % of pixels flagged as low quality
 - moderatequality_pixel_percent
 - % of pixels flagged as moderate quality



Target Platform Required Metadata

Target system is ESPDS and the netcdf4 outputs contain all the NDE required metadata.



VIIRS Flood Mapping Delivered Algorithm Package Processing

Presented by:
Eric Buzan



Software Architecture

- A general description of the VIIRS Flood Mapping product processing software architecture is presented in this section



Development Hardware

The VFM scientific development occurs in the STAR Collaborative Environment

- Science development hardware:
 - Dell Intel
 - OS Version: Centos 7
 - Fortran Compiler: Intel 19.0.4, GNU 8.3
 - C/C++ Compiler: Intel 19.0.4, GNU 8.3
- STAR machines
 - rhw1141 – DAP delivery testing for NDE with Intel 19.0.4
 - rhw1185 – Kubernetes job testing on ASSISTT Main Cluster
 - rhw1185 – System Validation and Near Real-time processing on ASSISTT Main Cluster



Development Environment

- The development occurs on ASSISTT's Main Kubernetes cluster (v 1.16.0) located in the STAR Collaborative Environment
- The Main Kubernetes cluster consists of:
 - 18 identical physical servers
 - Each Dell server has:
 - CentOS version 7.5 (x86_64)
 - 128GB RAM
 - 24 cores at 3.5 GHz
 - gcc and gfortran are version 8.3
 - icc and ifort are version 19.0.4
 - 8TB local drives (RAID)
 - 10Gb ethernet
- Upgrades: No plans for upgrades at this time



STAR Test Environment

- All jobs are currently ingested into the Main Kubernetes cluster (v 1.16.0) and run with Argo workflows (v 2.4.2)
- The Main Kubernetes cluster consists of 3 master nodes and 15 worker nodes
- The total statistics of the cluster are:
 - 2.3 TB RAM
 - 432 cores
 - 144 TB local drive storage



Operational Hardware

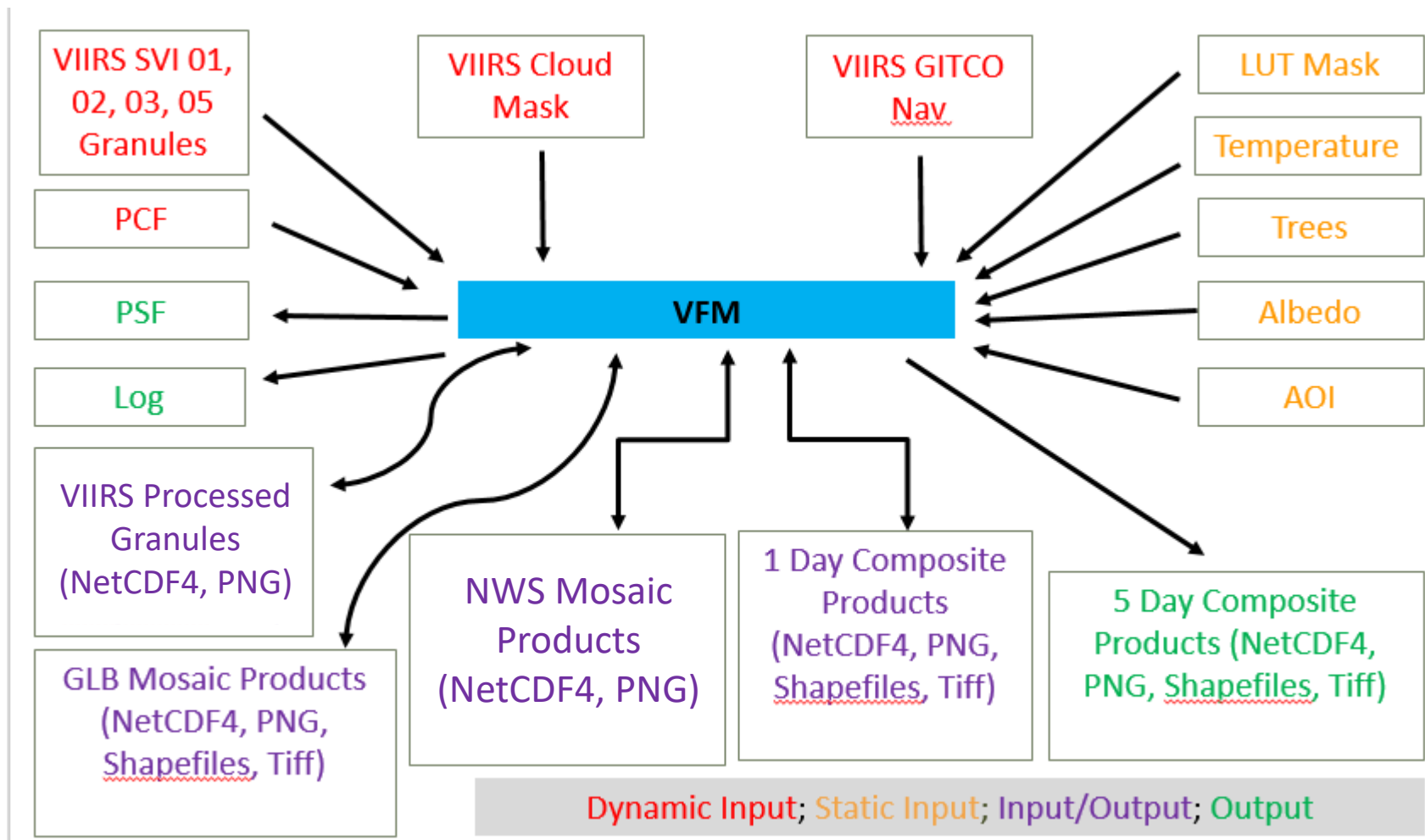
- The VFM official test machine is located within the NDE2.0 environment
- The VFM operational machine is located within the NDE2.0 environment



External Interfaces

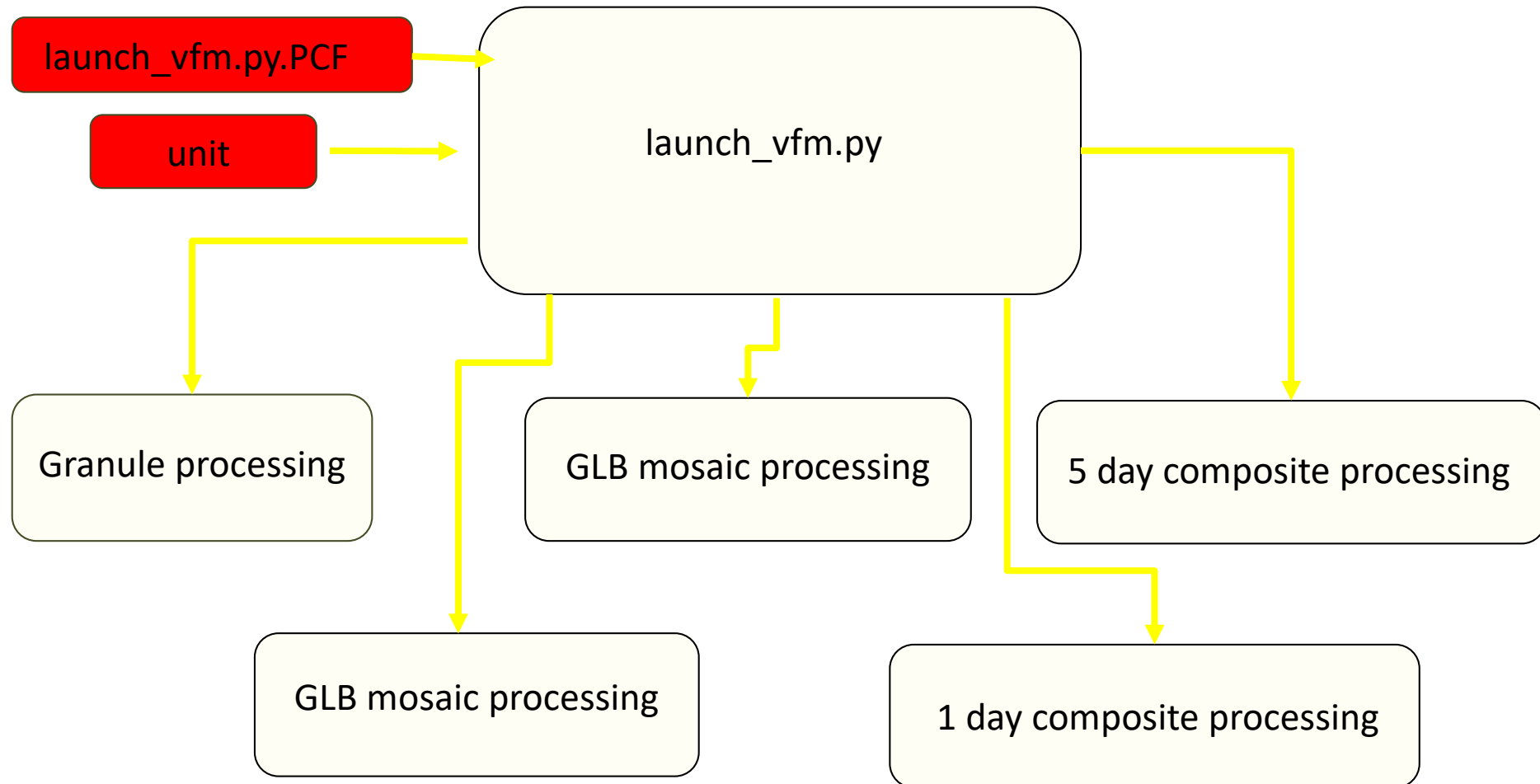
- The operational processing will occur in ESPDS. Satellite data and other dynamic ancillary inputs are received from PDA
- The VFM processing unit will be run via execution of top level driver script that will be invoked, monitored, and managed by the NDE Data Handling System(DHS)
- Each processing unit will receive a Process Control File (PCF) from the NDE Product Generation Manager (PGM) containing all input file locations, parameters and other information needed by the unit
- The processing run will produce a PSF containing information about processing unit product files

Context Diagram



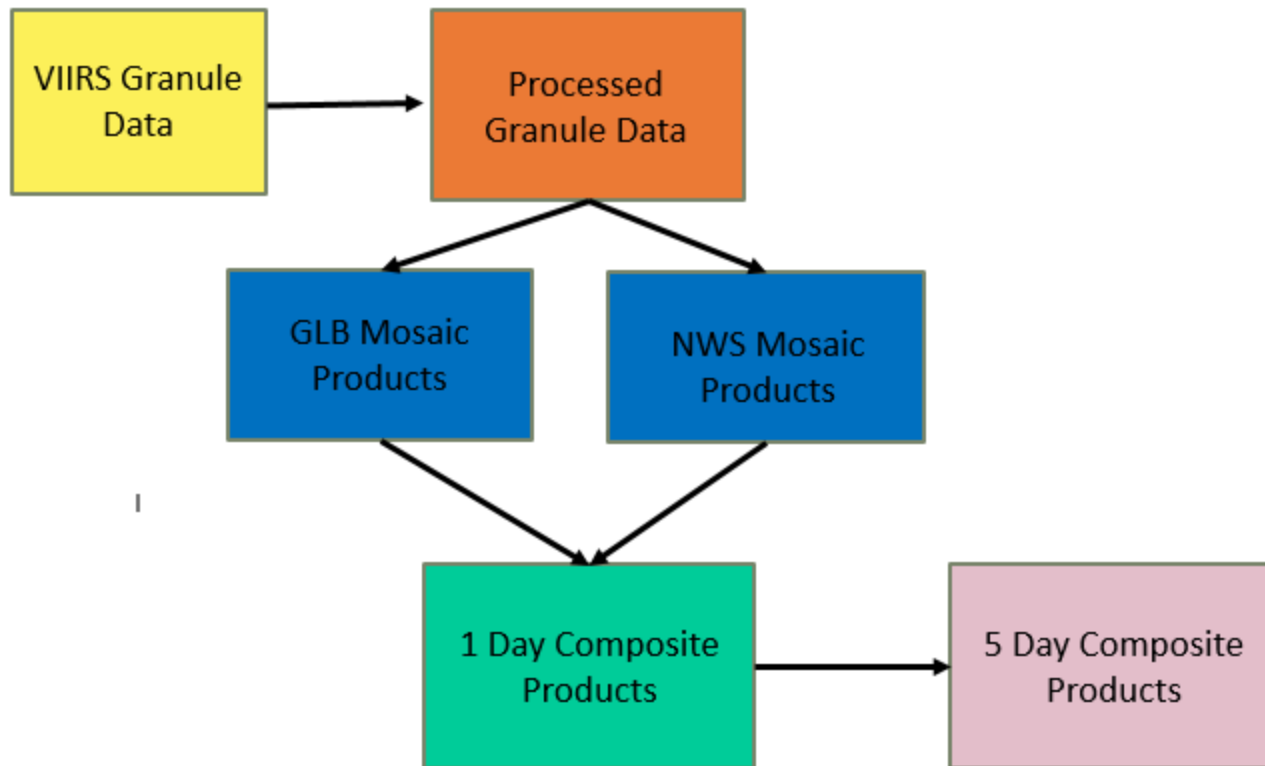


System Interface Overview





Product Precedence





System Interface Overview

- The driver script `launch_vfm.py` runs all 5 types of VFM processing
- PCF and run type (unit) are passed to `launch_vfm.py` at runtime
- Unit input parameter used to determine which type of processing should be run for a given invocation of `launch_vfm.py`



PCF

- Driver script requires path to the PCF at runtime
- Contains all variables necessary to successfully execute processing system
 - Will contain directory locations of input data, executables, configuration files, etc.
- PCF's are generated by NDE PGM for every granule being processed



Sample PCF (1/2)

```
working_dir=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/working
```

```
### define file names -- will not be changed after the system has been setup###
```

```
-----  
RUN_PLATFORM=NDE  
PROD_VERSION=v1r0  
SATELLITE=n20  
GRANULE_START=202010181843101  
GRANULE_DATE=20201018  
PCF_NAME=launch_vfm.py.PCF
```

```
--- Below are VFM algorithm specific directories ---
```

```
STATIC_DIR=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/data/static  
SCRIPTS_DIR=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/scripts
```

```
--- Below are VFM algorithm specific files/options ---
```

```
RUN_VFM_GRANULE=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/scripts/run_VFM_Granule.sh  
RUN_VFM_MOSAIC_GLB=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/scripts/run_VFM_Mosaic_GLB.sh  
RUN_VFM_MOSAIC_NWS=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/scripts/run_VFM_Mosaic_NWS.sh  
RUN_VFM_COMPOSITE_1DAY=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/scripts/run_VFM_Composite_1Day.sh  
RUN_VFM_COMPOSITE_5DAY=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/scripts/run_VFM_Composite_5Day.sh
```

```
SWATH_PROJECTION_EXE=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/src/vfm/VIIRS_Swath_Projection.exe  
FLOOD_DETECTION_EXE=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/src/vfm/VIIRS_Flood_Detection.exe  
MOSAIC_EXE=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/src/vfm/Mosaick_Subset_Flood_Process.exe  
COMPOSITE_EXE=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/src/vfm/VIIRS_Composition.exe  
FORNAV_EXE=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/src/ms2gt0.26/bin/fornav  
LL2CR_EXE=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/src/ms2gt0.26/bin/ll2cr  
LL2XY_EXE=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/exe_src/src/ms2gt0.26/bin/ll2xy
```



Sample PCF (2/2)

--- Below are input/output directories ---

```
SVI_DIR=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/sample_data/granule_input/n20
GITCO_DIR=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/sample_data/granule_input/n20
CLOUD_MASK_DIR=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/sample_data/granule_input/n20
PROCESSED_GRANULE_INPUT_DIR=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-
FloodMap/sample_data/processed_granule_input/n20
MOSAIC_INPUT_DIR=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/sample_data/mosaic_input/
COMPOSITE_INPUT_DIR=/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/sample_data/composite_input/
```

define NDE required metadata to be included in .cfg files

```
-----
production_site=nsof
production_environment=ite
INSTITUTION=DOC/NOAA/NESDIS/NDE+>+S-NPP+Data+Exploitation,+NESDIS,+NOAA,+U.S.+Department+of+Commerce
NAMING_AUTHORITY=gov.noaa.nesdis.nde
```

#END-of-PCF



Unit

- Driver script can be used to run all 5 different types of processing
- Unit parameter will allow the driver script to decide which unit to run
 - Granule
 - GLB mosaic
 - NWS mosaic
 - 1 day composite
 - 5 day composite



All Processing

- All VFM processing requires:
 - A bin/, data/, and working/ folder within the working directory
 - Environment variable with location of root processing directory
 - ms2gt0.26 executables
 - VIIRS_Flood_Detection.pcf file
 - Static ancillary files



Granule Processing

- When unit is set to 'granule'
- Requires VIIRS SVI (01, 02, 03, 05), Cloud Mask, and GITCO input
- Calls run_VFM_Granule.sh
 - VIIRS_Swath_Projection.exe
 - VIIRS_Flood_Detection.exe
- Produces flood products for each set of granule data
 - NetCDF4 output used in mosaic processing



Mosaic Processing

- When unit is set to either 'mosaic_glb' or 'mosaic_nws'
- Both mosaic processes can make use of any available processed granule data
- Calls run_VFM_Mosaic_GLB.sh or run_VFM_Mosaic_NWS.sh
 - Uses predefined area of interest static files
 - Mosaick_Subset_Flood_Process.exe
- Produces mosaic products for respective unit
 - Output for any area of interest that granule data falls within



Composite Processing

- When unit is set to either 'composite_1day' or 'composite_5day'
- 1 day composite used to create daily composites of mosaicked data files
- Calls run_VFM_Composite_1Day.sh or
 - VIIRS_Composition.exe
- Produces composite products for respective unit
 - Output for any area of interest that granule data falls within



Composite Processing

- 5 day composite used to create a composite of the 1 day composite product files
- Calls `run_VFM_Composite_5Day.sh` or
 - `VIIRS_Composition.exe`
- Produces composite products for respective unit
 - Output for any area of interest that granule data falls within



Logging

- All actions carried out within the Python scripts are captured in a log file
 - launch_vfm.py.log
- The stdout/stderr of each shell script is redirected to a text file with the name of the unit
 - For example: granule_processing.log



PSF

- At the end of each invocation of `launch_vfm.py` a Process Status File (PSF) is generated with all output produced
- Has the name `launch_vfm.py.PSF`



Sample PSF

```
/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/working/VIIRS-  
Flood_v1r0_n20_s202010181843101_e202010181844346_c202010301437059.nc  
/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/working/VIIRS-  
Flood_v1r0_n20_s202010181843101_e202010181844346_c202010301437059.shapefiles.zip  
/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/working/VIIRS-  
Flood_v1r0_n20_s202010181843101_e202010181844346_c202010301437059.tif  
/data/data513/kurtis.pinkney/NDE_DAPS/NDE-VIIRS-FloodMap/working/VIIRS-  
Flood_v1r0_n20_s202010181843101_e202010181844346_c202010301437059.png
```

#END-OF-PSF



DAP Software Description

- DAP Python scripts
 - 6 files
 - 504 SLOC



Test Environment & Timing Information

- The algorithm was tested on a Linux machine (Intel(R) Xeon(R) CPU E5-2643 v2 @ 3.50GHz machine, 2 GB disk space)
 - XL C/C++ and Fortran compilers was used
 - Algorithm was compiled as 64 bit
- Runtimes:
 - Granule – 4-12 minutes
 - Mosaic and Composite – 15-50 minutes



Resource Details

Runtime statistics - Memory and CPU

- All units run on a single thread and use 1 CPU
- Memory requirements vary by locations being processed

Executable	Units	Memory Usage
VIIRS_Swath_Projection.exe	Granule	500 MB
VIIRS_Flood_Detection.exe	Granule	700 MB – 4.5 GB
Mosaick_Subset_Flood_Process.exe	Mosaic	200 MB
VIIRS_Composition.exe	Composite	250 MB



Storage Information

Runtime statistics - Storage

Unit	Input Size	Output Size
Granule	~225 MB	60-450 MB
GLB Moasic	30-60 MB per granule Up to 1013 granules/day	50 MB per area of interest (up to 136)
NWS Moasic	30-60 MB per granule Up to 1013 granules/day	55-150 MB per area of interest (up to 8)
1 Day Composite	34-127 MB per input file (AOI/mosaic run/satellite)	55-150 MB per area of interest (up to 136)
5 Day Composite	34-127 MB per AOI per day (5 days)	55-150 MB per area of interest (up to 136)



Validation Datasets

- Validation dataset has been processed – granules only
 - Aug 27-29, 2020
 - Gulf of Mexico – Hurricane Laura
 - Oct 10-15, 2020
 - Gulf of Mexico – Hurricane Delta
 - Oct 8-13, 2020
 - 122-142°E, 42-54°N – Eastern China/Russia
 - 3-22°E, 3-15°N – Central Africa
 - 58-48°W, 38-25°S – SE South America



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 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.
- The start and completion of all steps in Python scripts are logged
- All errors or noteworthy conditions are trapped and logged



Process Monitoring

- The Process Status File contains information about data readiness. The PSFs provide information about the successfully generated product files from each unit run. The files may be used in operations for process monitoring in order to tracking the process completion and checking the availability of the product files



Product Monitoring Tool

Peter Keehn



Product Monitoring

- The 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.



VIIRS Flood Mapping Metadata

- Table

VFM		
ProdID	Smallint	Pk fk
dtFlag	Smallint	Pk fk
RecordDt	Timestamp	Pk
fillValuePixPct	Float	
lowQaulityPixPct	Float	
moderateQualityPixPct	Float	
endDT	Timestamp	
prodStatus	Smallint	
errMsg	text	



Summary

- The updated software architecture has been presented for generating the VFM products:
 - Algorithm software - updated to generate the expected outputs and to meet OSPO's coding standards.
 - DAP software architecture which includes the python wrapper scripts implemented to interface with ESPDS/NDE's DHS.



ARR Outline

- Introduction
- UTRR Risks & Actions
- Requirements
- Algorithm Validation
- Software Architecture
- **Delivered Algorithm Package**
- ARR Risks & Actions
- Summary



Delivered Algorithm Package

Presented by:
Eric Buzan



DAP

- The product systems and documents will be delivered to NDE/ OSPO through Google drive
- The input and test results are delivered via secured ftp
- The [Software Code Review](#) was conducted on [November 20, 2020](#)
- The [Final DAP](#) delivery will be made in [March 2021](#).



DAP

- The DAP will consist of:
 - *data* – Test case working directories
 - *docs* - All SPSRB / OSPO documentation
 - *ops_src*- All Python scripts used to run the production system
 - *exe_src*- All scripts, static system files, Fortran 77/90, C/C++ code
 - *sample_data* – test case data
 - *sample_pcf* – sample PCF files



DAP Script Execution Tree

```
— data
  — working_composite_1day
  — working_composite_5day
  — working_n20_glb_mosaic
  — working_n20_granule
  — working_n20_nws_mosaic
  — working_npp_glb_mosaic
  — working_npp_granule
  — working_npp_nws_mosaic
— docs
  — DAP_README_VFM.docx
  — DelivMemo_VFM.docx
  — gdal_install_notes.txt
  — VFM_PCF_PSF.docx
  — VFM_ProductionRules.docx
— exe_src
  — data
  — README
  — scripts
  — src
  — VIIRS_Flood_Detection.pcf
```

```
— ops_src
  — base_processing.py
  — composite_processing.py
  — granule_processing.py
  — launch_vfm.py
  — launch_vfm.py.log
  — logging_functions.py
  — mosaic_processing.py
— sample_data
  — composite_input
  — granule_input
  — mosaic_input
  — processed_granule_input
— sample_pcf
  — j01
  — launch_vfm.py.PCF
  — npp
— tests
  — test_granule_processing.py
```



DAP Checklist(1/3)

Required DAP Item	Location in Delivered DAP
Science algorithm source code, including make files and build scripts.	./exe_src/src/ms2gt0.26 ./exe_src/src/vfm
Processing Scripts, bin, templates	./ops_src
Template files for framework/ algorithm PCFs/ configuration files.	./sample_pcf
Test plans, test description, test procedures, and detailed performance testing results	./docs/
Test input/ancillary static/dynamic data	./exe_src/data/static ./sample_data
Test plans, test description, test procedures, and detailed performance testing results	./data



DAP Checklist(2/3)

Required DAP Item	Location in Delivered DAP
Sample intermediate/ output data files	./data
Production rule-set definitions	./docs/VFM_ProductionRules.docx
Quality monitoring information (quality flags, quality flag values).	SMM
Data flow diagrams.	SMM
Product file specifications – layout, content, and size.	SMM
List of exit codes and their associated messages.	SMM
List of expected compiler warnings	SMM



DAP Checklist(3/3)

Required DAP Item	Location in Delivered DAP
Estimates of resources required for execution.	SMM
Algorithm Theoretical Basis Documents (ATBDs) or reference to where the ATBDs can be obtained.	./docs/ATBD/*
Delivery Memo.	./docs/DelivMemo_VFM.docx (To be sent via email at time of delivery to NDE)
READMEs on System Configuration File, Job scheduling, etc.	./docs/DAP_README_VFM.docx



Documentation

- The VFM DAP contains the following [documents](#):
 - DAP_README_VFM.docx
 - DelivMemo_VFM.docx
 - gdal_install_notes.txt
 - VFM_PCF_PSF.docx
 - VFM_ProductionRules.docx



Algorithm Readiness Summary

- The software has been tested. The results have been presented
- The DAP contents have been verified
- The Final DAP will be delivered to ESPDS in March 2021
- The developers will work with OSPO to address any issues that arise during integration and OSPO system testing



ARR Outline

- Introduction
- UTRR Risks & Actions
- Requirements
- Algorithm Validation
- Software Architecture
- Delivered Algorithm Package
- ARR Risks & Actions
- Summary



ARR Risks & Actions

Presented by:
Priyanka Roy



CDR Risks and Actions

- **Action item #4** : OSPO should investigate the process for archiving DAPs at NCEI. This product uses non-enterprise versions of static data that is delivered with the DAP.



ARR Risks and Actions

- **Risk #1:** The NWS mosaic product may not meet the requested timeliness of 40 minutes.
- **Impact:** NWS may not be able to use the outputs.
- **Risk Assessment:** Medium
- **Risk Mitigation:**
 - NDE will need to implement geographic filtering and set up the trigger to process such that the domain is covered and timeliness is met.
- **Status:** Open

		CONSEQUENCES				
		1	2	3	4	5
LIKELIHOOD	5					
	4					
	3					
	2				X	
	1					



Risks and Action Summary

- 1 CDR Open Action – To be closed at ORR
- 1 ARR Risk with Medium assessment – To be closed at ORR .



ARR Outline

- Introduction
- UTRR Risks and Actions
- Requirements
- Algorithm Validation
- Software Architecture
- Unit Tests
- Risks & Actions
- Summary



Summary

Presented by:
Priyanka Roy



Review Objectives Addressed

- The following have been discussed:
 - Introduction
 - Risks & Actions
 - Requirements
 - Algorithm Validation
 - Software Architecture
 - Delivered Algorithm Package
 - ARR Risks & Actions
 - Summary



Next Steps

- Deliver the final DAP to NDE
- Support integration on NDE



Open Discussion

- The Review is now open for discussion