System Verification Report (SVR)

version 1.0, 27 November 2019

Contents

Author	Approved	Signature	Date
JF Piolle	Fabrice Ardhuin, Ellis Ash		27/11/2019
ESA Acceptance	Digitally signed by Craig Donlon Date: 2020.02.19 16:00:56 +01'00'		

Issue	Date	Comments
1.0	27/11/2019	First version submitted to ESA

List of Acronyms

CCI	Climate Change Initiative
CLS	Collecte Localisation Satellite
CMEMS	Copernicus Marine Environment Monitoring Service
CPU	Central Processing Unit
ECV	Essential Climate Variable
ESA	European Space Agency
GDR	Geophysical Data Record
HPC	High Performance Computing
L1A	Level 1A
L1B	Level 1B
L4	Level 4
LRM	Low Rate Measurement
RA	Radar Altimeters
S3A	Sentinel-3A
S3B	Sentinel-3B
SAR	Synthetic Aperture Radar
SMOS	Soil Moisture and Ocean Salinity satellite
SSH	Secure Shell
SVR	System Verification Report
SWH	Significant Wave Height
WV	Wave (mode for SAR)

1. Introduction

This document presents the System Verification Report (SVR) for **Sea_State_cci**, deliverable 3.3 of the project. This first version is prepared in the first year of the project in order to set the scene for system verification activity. An updated version will be produced towards the end of each year of the project. In this first version it is not yet possible to elaborate details of system verification tests and results as the processing system for the Sea State CCI version 2 data is not fully defined. Sea State CCI version 1.1 data has been produced using the GlobWave heritage processing system, with verification considerations presented here in section 3.

The remainder of this System Verification Report contains two further sections:

- Section 2: Overview of the Processing Chain
- Section 3: Overview of System Verification

2. Overview of the Processing Chain

At the end of Year 1 of the project, the processing chain for future production of CCI data sets for Sea State is still to be finalised. For version 1 of this SVR this section summarises the options of system architectures to be deployed, as presented in section 6 of the System Specification Document.

Keeping long and massive mission archives alive by raising the level of data revisiting through multiple applications, demonstration products or services, or extensive data reprocessing are major requirements for projects such as the CCI Sea State in order to combine numerous sources of data, test new ideas and perform reprocessing over a significant amount of data.

The system architecture for the project will therefore rely on existing infrastructures providing these capacities.

Among the key technical drivers identified to provide an efficient and cost-effective access to historical massive multi-mission archives are:

- fast and online access to massive collections of data
- avoiding data duplication and transfer to users
- minimizing time from algorithm development to processing
- allowing fast and easy to manage large scale reprocessing
- improving data storage and management

There may be several target infrastructures fitting these criteria and the CCI Sea State project may use one or several of them, depending on their respective strengths and limitations for a particular mission retracking. Possible infrastructures to be used include:

- Ifremer “Datarmor” platform, originally intended for most of the processing
- ESA “SARvatore”, which may prove more adapted for ESA missions (thanks to the availability of the input data) such as Sentinel-3A & B, CryoSat-2 or Envisat
- ESA CNES, which hosts all Jason mission data and has already integrated retrackers

Contacts have been engaged with all these possible resource providers and will have to be discussed more deeply once the project has settled the processing scope (missions and retrackers to be used).

Ifremer “Datarmor” platform

Physically, the platform is based on the Datarmor platform operated by Ifremer IT department (refer to the facility section in the management proposal). The current storage capability is about 12 PB (20 PB by 2020) and the available capacity largely exceeds the need for CCI products.

The reprocessing framework for CCI will therefore make use of direct access to the complete CCI input data archive on disk, physically located within the cluster, and distribution of the processing over multiple nodes of the cluster. The task of managing and distributing the processing jobs is alleviated by the use of batch tools implemented by CERSAT (and used for all its reprocessing works) such as Gogolist.

Prun is a tool which aims to ease the execution and monitoring of embarrassingly parallel processings. It is a wrapper which submits jobarrays to batch schedulers (torque/maui, oar, pbspro...) and manages the output logs and progress reports, with an easy monitoring.

Jobarray submission to batch schedulers is generally easy, but monitoring the job progress and accessing to the error logs of a few tasks among thousands is the same as finding a needle in a haystack... This wrapper was created to avoid spending more time in manual “logfile-mining” (grep, tail...) than the processing time itself. Some other needs it addresses:

- listing management avoids too big jobarrays (which crashes some batch schedulers)

- list the tasks in errors to ease their reprocessing
- allowing several execution modes : sequential, streaming (pipe)
- allowing multiple batch scheduler as backend
- having tasks status reporting and real-time monitoring
- having job history, meta-data & log files organization

This tool is easily handled by users and has been used successfully in other projects involving multiple remote partners such as OceanFlux. Online help is available but direct support par CERSAT engineering team is also provided.

The general scenario for the reprocessing of the altimeter or SAR data with a dedicated processor provided by a project partner will be as follows:

Objective:
A partner wants to reprocess or analysis a complete time series of data, which requires large processing resources and accessing a huge volume of data.

Scenario:
The user has a processor or analysis script ready to be used. It may be in python (preferably) or any other language available on the *Datarmor* platform.

If a language or a dependency is not installed on *Datarmor*, it can be added by the facility operation team. If specific system requirements are mandatory (for example, a specific linux distribution and version), a virtual server or a docker image can be installed but we encourage users to rather implement and test their processors on the default server operating system of *Datarmor*.

The user connects to the platform by SSH to a virtual machine on the cluster. It uploads its processor. The Ifremer system team takes it from here and test it on sample data, comparing the result with the partner to ensure they are identical. Integration and testing issues are solved by direct interaction between the Ifremer team and the partner providing the processor.

If the same task (processor) is to be run over a large amount of files, the processing can be easily distributed over different nodes of *Datarmor* using *gogolist*, a user friendly batch manager. The operator just needs to provide the list of files he wants to process, the processor command line, possibly the minimal resource requirements (ex: memory): the batch manager takes in charge the execution of the processor over each file and reports to the user on processing progress, errors. Unprocessed files and logs are clearly reported back to the user at the overall processing completion.

The results can be analysed remotely by the project partners by connecting through SSH to *Datarmor*.

There are two major activities at Ifremer which deal with data services, namely Coriolis/SISMER for in situ and CERSAT for satellite. Their respective operational components are currently being merged. They provide the evidence of Ifremer operational capacity to host the processing part of the CCI Seat State Service. The software for the CCI data processing system will also be based on the existing multi-mission (re)processing framework available from Ifremer for satellite data, the majority of which has been developed within the scope of operational or demonstration services, such as Medspiration, GlobWave, OSI SAF or CMEMS, to implement a real-time operational capacity under strong timeliness and robustness requirements by oceanographic and meteorological agencies. Ifremer is also an expertise and processing center for SMOS mission (CATDS, PI-MEP, SMOS-Storm,) and upcoming French-Chinese wave observation mission CFOSAT.

The operational center is operated by a support technical team (CLS), whereas the Ifremer engineers monitor the data processing, ensuring quality control, interfaces with the different entities involved in these operational activities or with other collaborating data centers, as well as maintenance or improvement of the operating system.

To perform its archiving and processing activities, as well as to support its research team in their activities, CERSAT has implemented a set of resources that will also support the project.

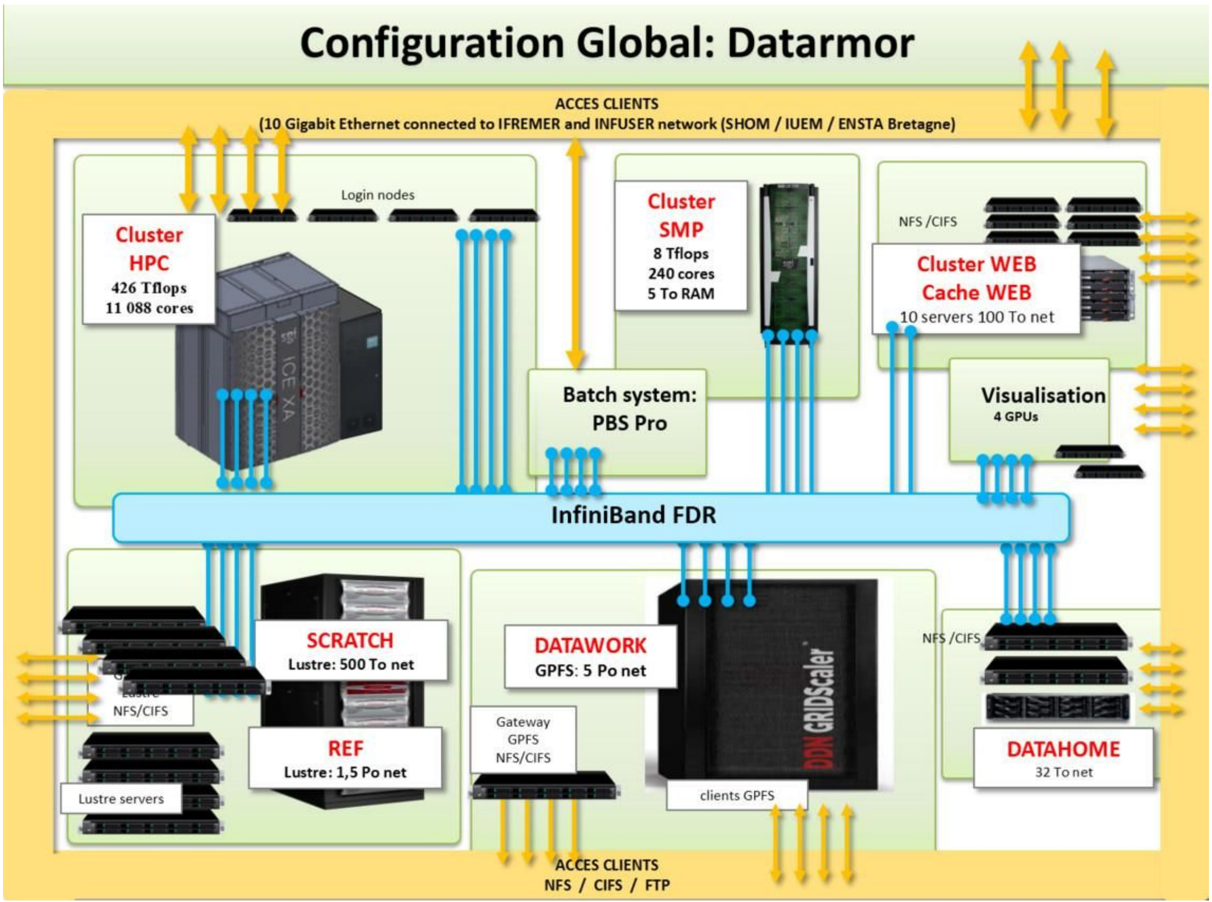
To cope with the reprocessing requirements of the CCI Sea State, the service will be operated on the new Ifremer supercomputer, *Datarmor*. *Datarmor* extends the capabilities of the previous cluster for:

- CPU intensive or memory demanding applications and processing
- massive data storage

It consists of:

- 11088 cores - 426 Tflops (128 GB memory and 28 cores per node) for HPC applications
- 240 cores, 5 TB RAM for non MPI processing
- 10 servers for web services

- large data storage capacity
  - 500 TB Lustre for HPC
  - 1.5 PB Luster storage for reference data
  - 5 PB GPFS storage for project and work data
  - 100 TB for services and web applications
  - 32 TB per home directory



1.5 PB storage is currently reserved for CERSAT processing in this infrastructure which is deemed to be sufficient for the CCI Sea State requirements.

ESA SARvatore

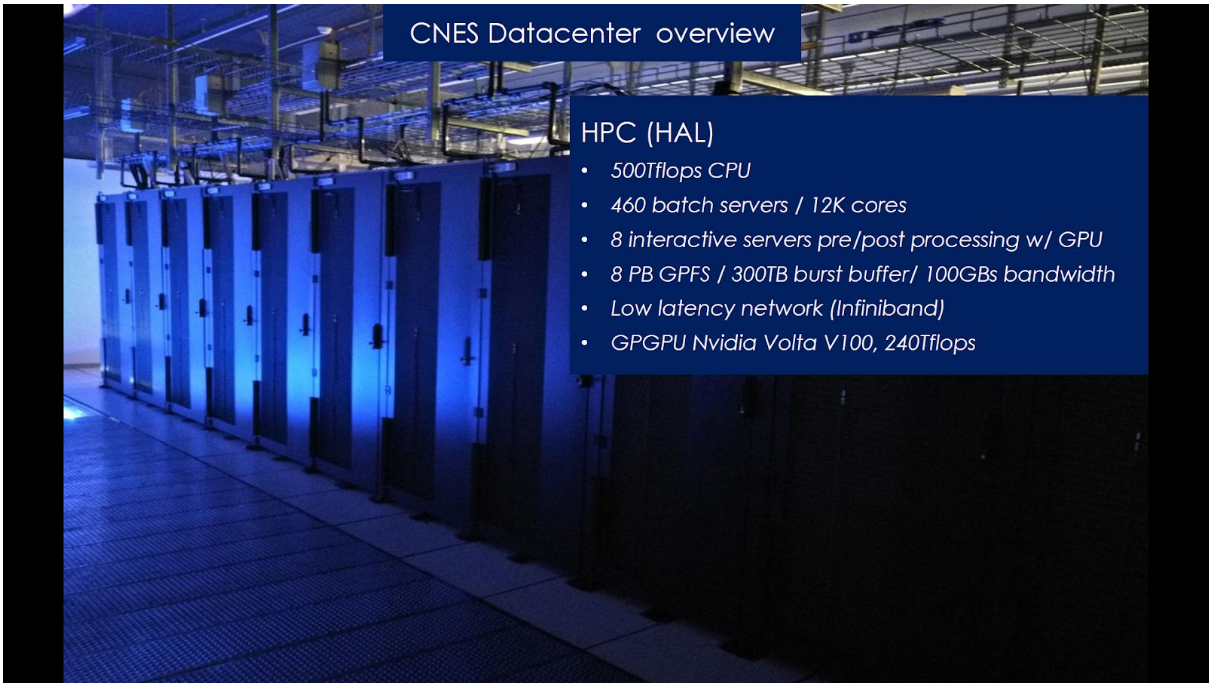
The Grid-Processing-On-Demand Sentinel-3 & CryoSat SAR/SARin Processing service is a web platform that provides the capability to process online and on demand Sentinel-3 SAR and CryoSat SAR and SARin data, from L1a (FBR) data products until Level-2 geophysical data products, with different algorithms then Ground Segment products. The service, coined **SARvatore** (**SAR** **V**ersatile **A**ltimetric **T**oolkit for **O**cean **R**esearch & **E**xploitation), can process data over any surface.

The service is open, free of charge and accessible online from everywhere. To access the service, users need EO-SSO (Earth Observation Single Sign-On) credentials. Registration can be done at https://earth.esa.int/web/guest/general-registration). Afterwards, an e-mail to the G-POD team (eo-gpod@esa.int) shall be sent requesting the activation of the CryoSat-2/Sentinel-3 service for the EO-SSO user account.

In particular it hosts the highly voluminous L1A and L1B data for Sentinel-3 SRAL. In addition, the platform provides engineering support to help the integration and deployment of the processors the CCI Sea State may need to run on the platform.

CNES HAL

The CNES HPC center is already used for some altimetry reprocessing campaigns for AVISO project. It is also a possible target or complement to our project for addressing the Jason mission series and possibly others implemented on this platform (like ERS or Envisat for FCDR project).



3. Overview of System Verification

In version 1 of this SVR we present an overview of the system control and verification issues, as applicable to the heritage GlobWave processing system used to produce the version 1.1 Sea State CCI products, hereafter referred to the CCIv0 system. Detailed verification tests and results for the future CCI processing system will be elaborated in subsequent versions of this report.

Since each task of the system is designed to be run independently, a top-level system component is needed to link together these tasks (workflow management) and to provide an overview of the system operations. This is done by the system controller which is in charge of sequencing the various tasks of the CCIv0 system and logging and monitoring the system operations.

The system control is performed through a framework developed and implemented in the frame of the Medspiration project (and subsequent operational projects at Ifremer/CERSAT), ensuring the robustness and reliability of our solution. This generic framework consists of the following elements:

- A scheduler which is in charge of running the tasks (actually the associated process(es)) when required and sequencing them correctly.

- Specialised process runners which control the execution of a single process. The tasks are run on a distributed environment which means that the system has to run each one in its dedicated environment or machine. This is done by the process runner which selects and spawns the processes depending on the available resources and the load of available servers.

- A log server which maintains a log of all events occurring within the system and allows queries of recent (for the monitoring of the system operation) or and historical records (to extract operation statistics).

- A monitoring graphical user interface which allows a system operator, through a graphical user interface, to have an overview of the current state of the system and of the potential errors which have occurred.

3.1 Scheduler

The scheduler is the core entity of the system. It is permanently active and initiates all the processing operations within the system. Its main functions are:

- To run the various processing tasks on time (for periodic processes) or when incoming data are available (polling data spools). Before running a process, it is able to perform some pre-condition testing (if required) to assess whether a process can be run effectively.

- Registering its log information to the log server (when triggering a process for instance).

The scheduler is designed to recover after a system failure, detecting in its initialization phase all the processes which were aborted or not completed, and running them again.

For reliability and robustness reasons, two different controllers are operated simultaneously on two different hosts, minimising the risk of a failure.

3.2 Process Runners

A process runner is used by the scheduler to execute a given process. Its main functions are:

- To determine the target host to run the process and to launch the process.

- To format a command file and to adapt the inputs to the correct syntax expected by the process. This function is required because of the heterogeneity of the processes implemented from different sources, which may require this additional encapsulation layer. This implies that there is a specific process runner associated with each process and that these process runners are designed as template processes which have to be possibly completed with specific additional code in some cases.

- To trace all the system execution context (start time, host, process identifier, etc) related to the process in an associated task file.

- To log into the log server the relevant captured events regarding the process.

3.3 System Monitoring

System monitoring is facilitated by a graphical user interface that provides an overview of the current operations and of the state of CCIv0 system to the system operators. It consists of a graphical user interface featuring:

- A synoptic view of all active processes, presenting their status and, if existing, access to the process log. This view uses the task files generated by the process runners to provide the information.

- A view of all system alarms (subset of error logs) which may require action or investigation by the system operators. Each alarm has to be acknowledged by the operator.

- A view of the complete registered logs (to investigate the workflow within the system).

3.4 Log Servers

The log server is a database which contains and registers all incoming log entries from the CCIv0 system controller (direct connection to the database by the scheduler or the task runners). The content is then accessible for query by the monitoring application or to extract processing statistics.