Journal of Geomatics Vol. 13, No. 2, October 2019

Development of Oceansat-2 OCM Data Cube over Indian Subcontinent

Tushar Shukla*, Sampa Roy and Debajyoti Dhar
Space Applications Centre, Ahmedabad
*Email: tushar@sac.isro.gov.in

(Received: Jan 18, 2019; in final form: Nov 18, 2019)

Abstract: The number of Earth observation(EO) data users and developers are growing and a number of challenges need to be solved to fill the gap of acquisition and use of ever-increasing satellite data acquired by ISRO. The majority of EO data still remains underutilized mainly because of the challenges of big data namely, volume, velocity, veracity and variety. However, the full information potential of EO data can be utilized by directly providing Analysis-Ready-Data(ARD) to the user community. The ARD has all pre-applied corrections for radiometry and geometry. EO Data Cube (DC) is a new paradigm aiming to realize the full potential of satellite data by eliminating the barriers caused by these big data challenges and providing access to large Spatio-temporal data in a user and developer-friendly environment, thereby fulfilling both visualization and analysis needs. Systematic and regular provision of Oceansat-2 OCM Analysis Ready Data (ARD) will significantly reduce the post-processing burden on ISRO's Oceansat series data users and application scientists. Nevertheless, ARD is not commonly produced as a part of standard data processing chain of Oceansat-2 mission (operational at IMGEOS/NRSC, Hyderbad) and therefore getting uniform and consistent ARD remains a challenging task. This paper presents an approach to enable rapid data access and pre-processing to generate ARD using interoperable services chains. The approach has been tested and validated by generating OCM-2 ARD while building the Oceansat-2 OCM Data Cube.

Keywords: DataCube, Oceansat-2, Ocean Color Monitor(OCM), ARD, Pre-processing, ISRO.

1. Introduction

Due to pressures from climate change, demographic, and economic growth, natural resource consumption and exploitation are more than ever (Rockstrom, et al., 2009). To better preserve the quality of the environment and improve the management of natural resources and land planning, it is useful to monitor these changes through time (Wulder et al. 2008). One of the main characteristics of remote sensing is the ability to provide a synoptic view of a given spatial extent. With the archives from ISRO's Ocean colour monitoring satellite sensors, the evolution of this coverage can be monitored all the way back to 1999 (with the launch of IRS-P4). Now with the introduction of new satellite sensors (e.g. Oceansat-2 and upcoming Oceansat-3) facilitate inter-decade comparison and analysis of EO data. Remotely sensed Earth Observations (EO) data are increasingly available from a number of freely and openly accessible repositories. These data are highly valuable because of their unique and globally consistent information that they include (Lewis, et al., 2016). Indeed, global observations together with scientific expertise and appropriate tools provide substantial benefit supporting economic development, decision-making, and policy implementation for all countries. However, the full information potential of EO data has not been yet realized. They remain still underutilized and stored in electronic silos of data. This is due to several reasons:

(1) increasing volumes of data generated by EO satellites;
(2) lack of expertise, infrastructure, or internet bandwidth to efficiently and effectively access, process, and utilize EO data;
(3) the particular type of highly structured data that EO data represent introducing challenges when trying to integrate or analyze them;
(4) and the substantial effort and cost required to store and process data limits the efficient use of these data.

The EO data can be considered as Big Data, data that are too large, fast-lived, heterogeneous, or complex to get understood and exploited (Baumann, et al., 2016a). Consequently, we need new approaches to fully benefit from EO data and support decision-makers with the knowledge they require by systematically analyzing all available observations and convert them into meaningful geophysical variables. To address these Big Data challenges, it is necessary to move away from traditional local processing (e.g. desktop computer) and data distribution methods (e.g. scene-based file download) and lower the barriers caused by data size and related complexities in data preparation, handling, storage and analysis. This paradigm shift is currently represented by EO Data Cubes (Baumann, et al., 2016b), an approach that is receiving increasing attention as a new solution to store, organize, manage, and analyze EO data in a way that was not possible before. Data Cubes (DC) are aiming to realize the full potential of EO data repositories by addressing Volume, Velocity, and Variety challenges, providing access to large Spatio-temporal data in an analysis-ready form.

2. Related Research and Operation

Currently, there are various operational DC like the Australian Geoscience Data Cube (AGDC). These different initiatives are covering different spatial scales (e.g. AGDC, EODC (Earth observation Data cube by ESA [European Space Agency]); storing different data (e.g. only Landsat 8 for the EODC while the AGDC stores Landsat 5, 7, 8, MODIS, and Sentinel 2 data; only processed products for the ESDC); using different infrastructure (e.g. high performance computer for the AGDC, and cloud used by many others); using different

software implementations (e.g. Open Data Cube for the AGDC; RasDaMan by ESA). Figure 1 illustrates a spatial Hadoop framework for storing and serving the petabytes of EO data. Rasdaman used by ESA follows a similar framework for storing their datacube. The diversity of approaches asks also for a clear definition of an EO Data Cube. A recent publication of The Datacube Manifesto by CEOS defines a Data Cube as a massive multi-dimensional array, also called raster data or gridded data; massive entails that we talk about sizes significantly beyond the main memory resources of the server hardware. Data values, all of the same data type, sit at grid points as defined by the d axes of the d-dimensional datacube. Coordinates along these axes allow addressing data values unambiguously. A d-dimensional grid is characterized by the fact that each inner grid point has exactly two neighbours along each direction; border grid points have just one. The main objective of this initiative is to provide a data architecture solution to lower the technical barriers for users to exploit EO data to its full potential and consequently solving the problem of accessibility and use while increasing the impact of EO data. The primary problems for users are data access, data preparation, and efficient analyses to support user applications. The two first issues are essential challenges to tackle while building a DC. Indeed, these steps concern the generation of Analysis Ready Data (ARD). CEOS defines ARD as satellite data that have been processed to a minimum set of requirements and organized into a form that allows immediate analysis without additional user effort. Figure 2 shows the ARD production steps from RAW satellite data. It is envisioned that systematic and regular provision of ARD will significantly reduce the burden on EO data users. To be considered as ARD, data should satisfy the following requirements:

(1) metadata description; (2) radiometric calibration; (3) geometric calibration. ARD data from various ISRO missions such as Oceansat series and Resources at series can be ordered by placing a request at UOPS (User online processing system) maintained by NRSC, Hyderabad. However, getting uniform and consistent ARD remains a challenging task due to various environmental challenges and acquisition-related problems. As such, data ordering and delivery can take long time (e.g. several hours or days); and the full process from ordering to getting the data has not been automated yet. This clearly limits the accessibility and ingestion processes while building and updating a DC and consequently ask to find alternative ways to generate ARD products. Recognizing these issues, the aim of this paper is to present an approach to enable rapid data access and pre-processing to generate Analysis Ready Data. The approach has been tested and validated by significantly facilitating the generation of ARD using Oceansat-2 OCM medium-resolution imagery allowing to build the first version of the OCM-2 Data Cube.

3. Building OCM-2 Datacube: Techniques and Infrastructure Implementation

The Data Cube is a system designed to:
1) Catalogue large amounts of Earth Observation data
2) Provide a Python-based API for high-performance querying and data access
3) Give scientists and other users easy ability to perform Exploratory Data Analysis
4) Allow scalable continent-scale processing of the stored data
5) Track the provenance of all the contained data to allow for quality control and updates.

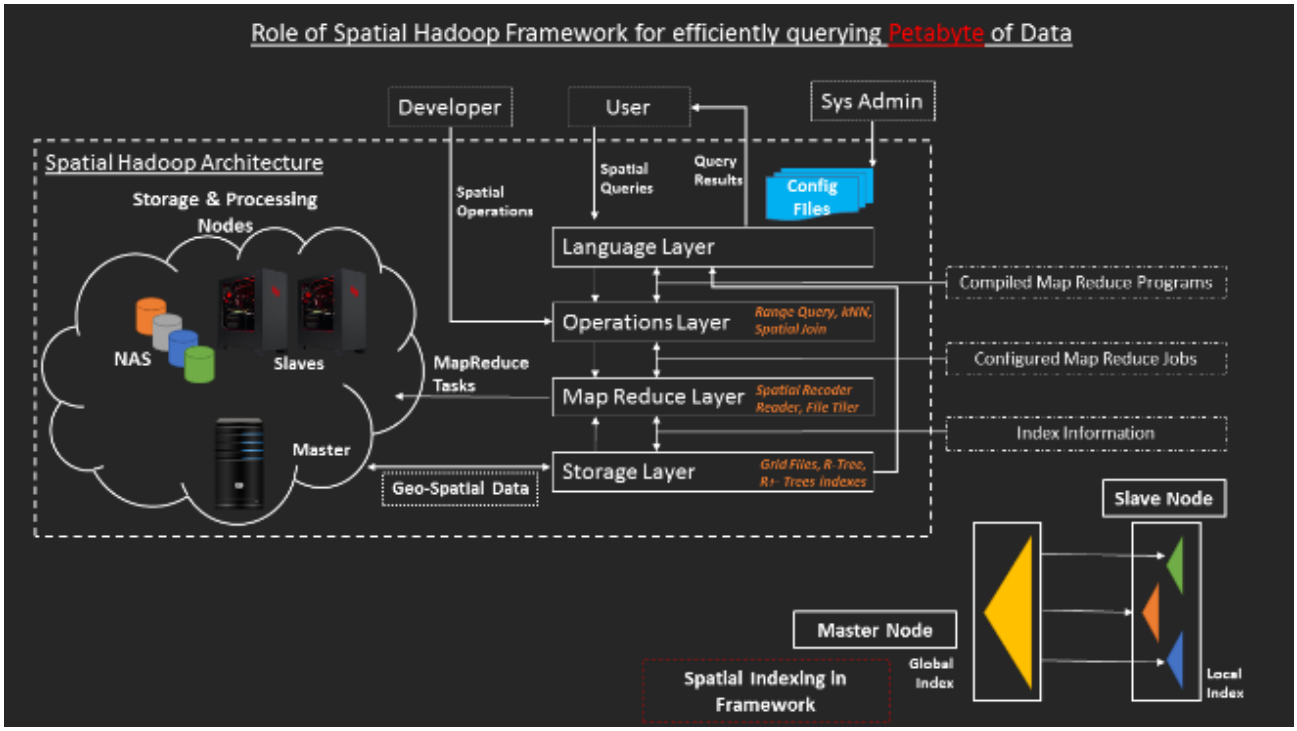

Figure 1: Spatial Hadoop Architecture for DataCube Infrastructure

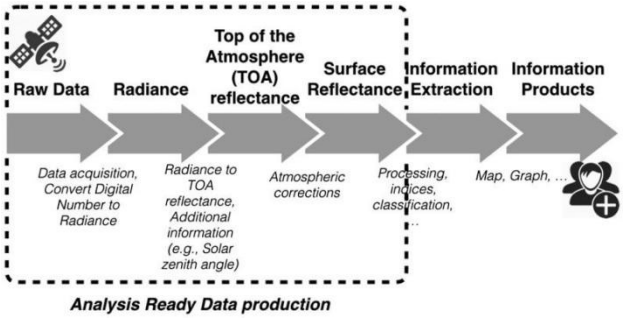

Figure 2: ARD production steps

A fundamental aspect while building a DC is having ARD products ingested, stored in the database, and readily available. Considering that ARD products are not commonly generated by data providers and the fact that current delivery mechanisms are not efficient, this requires finding a procedure to routinely generate ARD ensuring that all observations stored in a Data Cube are consistent and comparable. Ideally, this procedure must be automated as much as possible (e.g. discover, download, and pre-processing), should be able to discover and access data from different repositories, should handle different sensors (e.g. Oceansat-2 OCM, Resources at-2 LISS-3, LISS-4 and AWIFS), and should be interoperable (e.g. to enhance reusability).

To satisfy these requirements, the ARD Product Generation and Ingestion (APGI) framework has been developed and used. Figure 3 illustrates this automatic processing workflow for directly preparing the raw product for ingestion into datacube. APGI is a framework that helps to automate EO data discovery and (pre-)processing using interoperable service chains for transforming observations into information products suitable for monitoring environmental changes (Giuliani, et al., 2017). This framework is developed using a combination of large storage capacities, high-performance computers, and interoperable standards to develop a scalable, consistent, flexible, and efficient analysis system that can be used on various domains through decades of data for monitoring purposes. APGI take the discovers the RAW product and Generated ARD and finally Ingests specified data to Datacube. While building the DC, the APGI framework has helped to automatically generate ARD products by overcoming the obstacles presented manual product generation and ingestion. Figure 4 highlights the key components of OCM-2 datacube framework for accessing ingested data via API and OGC compliant services.

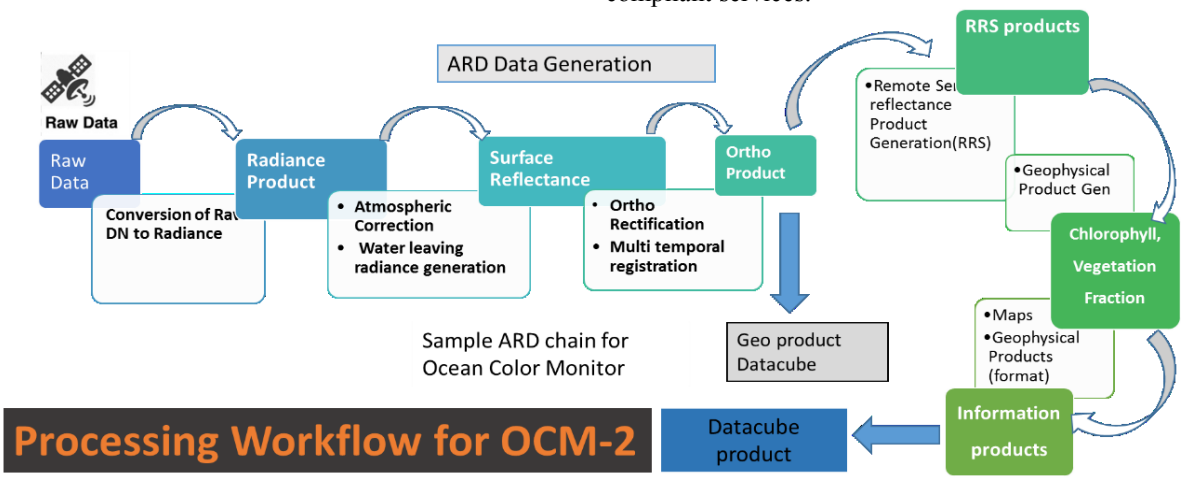

Figure 3: Processing workflow for OCM-2 RAW to ARD to Datacube Chain

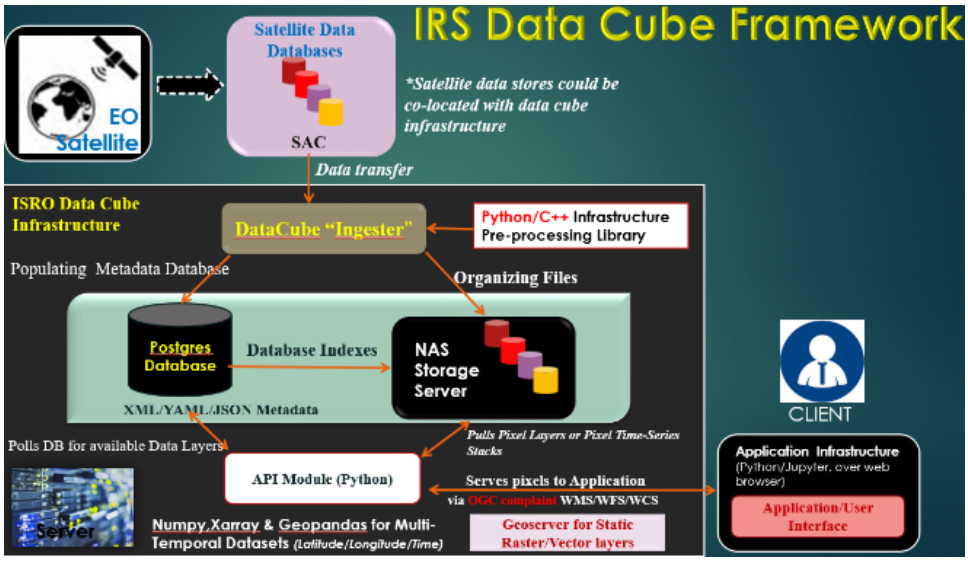

Figure 4: IRS Data cube framework for OCM2

2.1 Indexing Data

When you load data into the Data Cube, all you are doing is recording the existence of and detailed metadata about the data into the index. None of the data itself is copied, moved or transformed. This is, therefore, a relatively safe and fast process. There are a few pre-requisites for Indexing data:

1) A working Data Cube setup.
2) Some Analysis Ready Data to load.
3) A Product definition added to your Data Cube for each type of dataset.
4) Dataset metadata documents for each individual dataset.

2.2 Oceansat OCM-2 data volume, coverage and Computing performance

The available OCM-2 data had the total cumulative size of more than 30 TeraBytes (TB) (2-day repetivity and total year duration of 2011-2018) and as such high computing performance became one the major requirement of generating the datacube structure. Figure 5 illustrates the coverage of OCM-2 Path 9 and 10 coverage over India, Srilanka and parts of Tibet and Pakistan. High computing and processing performance was achieved by developing efficient software written in python and c++ for geophysical parameter generation, multi-temporal image registration, indexing and ingestion; to efficiently utilize multi-processing environment. Both data and task-level parallelism techniques were employed to process data within a meaningful time duration. The entire activity was divided into smaller goals for building this huge Datacube: development of scripts for

(a) large data handling and reducing redundancy;
(b) efficient storage and categorization of radiance,
(c) reflectance and geophysical data for rapid access;
(d) reference generation and multi-temporal image registration; and finally
(e) development of geo-spatial web user interface.

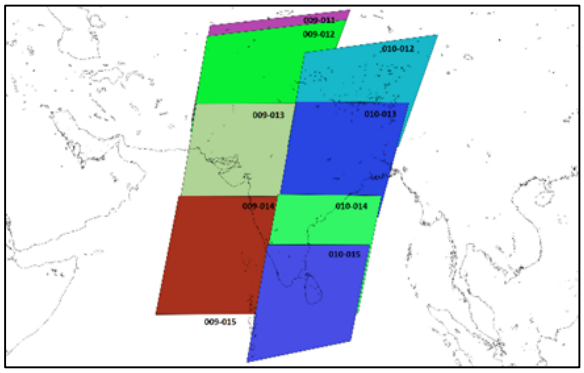

Figure 5: OCM-2 Path Row coverage over India

3.2.1 Working behind Data loading

Types of Data Loading

There are two major use-cases for loading data from the Datacube Ad hoc access, and Large scale processing. These are described below.

1. Ad hoc access
   - A small spatial region and time segment are chosen by the user.
   - Data is expected to fit into RAM.
2. Large scale processing (GridWorkflow)
   - Continental-scale processing
   - Used to compute new products or to perform statistics on existing data
   - Often unconstrained spatially
   - Often unconstrained along the time dimension
   - Data is accessed using a regular grid in small enough chunks

The specific access pattern is algorithm/compute environment-dependent and is supplied by the user and requires manual tuning.

For large scale processing, we turn to Dask (Dask Development Team, 2016) library which offers lazy load processing, this is explained in the following section.

3.2.2 Lazy load with Dask

In computer science, context lazy means roughly not computed until needed. Rather then loading all the data immediately load_data() function can instead construct a array (Hoyer, et al., 2016). The dataset that the user can use in the same way as a fully loaded data set, except that pixel data will be fetched from disk/network on-demand as needed. The on-demand loading functionality is provided by third-party libraries xarray and dask (used internally by array). Datacube code constructs a process for loading data on demand, this process is executed as needed by xarray + dask library when real data is required to be loaded for the first time.

3.2.3 Internal interfaces

The primary internal interface for loading data from storage is BandDataSource class, unfortunately, this rather generic name is taken by the specific implementation based on the raster library. BandDataSource is responsible for describing data stored for a given band, one can query:

- The Shape (in pixels) and data type
- Geospatial information: CRS + Affine transform

and also provides access to pixel data via 2 methods

1. **read**(): access a section of source data in native projection but possibly in different resolution
2. **reproject**(): access a section of source data, re-projecting to an arbitrary projection/resolution

This interface follows very closely the interface provided by the raster library. Conflating the reading and transformation of pixel data into one function is motivated by the need for efficient data access. Some file formats support multi-resolution storage for example, so it is more efficient to read data at the appropriate scale rather than reading highest resolution version followed by downsampling. Similarly, re-projection can be more memory efficient if source data is loaded in smaller chunks interleaved with raster warping execution compared to a conceptually simpler but less efficient load all then warp all approach.

3.3 Improved SIFT-based data product registration

Achieving sub-pixel accuracy is a must for valid time-series data and composite data product generation. An improved Scale Invariant Feature Transformation technique was developed to solve this challenge. For all the years 2011-2018, seasonal references are generated and geometrically registered for within the year image

registration. Further to handle the huge amount of associated data processing, extensively parallel C++ software were written for the utilizing full potential of multi-processor environment using OpenMP, SSE, and AVX (Shukla et. al, 2018).

3.4 Geo-Spatial Web-Interface

Developing a multi-temporal data analysis portal which will help users in the visualization and analysis of pre-processed data. It utilizes the strength of the underlying On-Line Processing of ARD temporal data-stacks for same geospatial regions.

The platform provides freedom to develop and integrate pluggable applications for various algorithms which in turn can help users to process data online and get results instead of downloading input data and setting up environment for applications to run for the same this, in turn, saves lot of user's time. The developed Web User interface uses Geoserver for serving Static layers such the pre-computed monthly composite layers to quickly serve data using WMS, and Datacube API calls for accessing multi-temporal ARD. Some major highlights of Web Interface are:

- Online available Analysis Ready Data (includes Multi-Temporal Registration correction)
- On-the-fly post-processing of ARD data (ingested in datacube framework) to generate custom mosaic for different Bio-geophysical products such as
    - Vegetation fraction
    - Land Surface Water,
    - Chlorophyll-a concentration
    - Aerosol Optical Depth
    - Remote sensing reflectance (6 bands)
- Online Application for Change Detection between selected dates (includes PCA, SSIM etc.)
- Tool for ROI based on-the-fly pixel drilling query over available geophysical products. (Figure 6)
- Online Time-series Trend analysis algorithms such ARIMA.

Figure 7 illustrates the web interface for data cube system allowing easy access to multi-temporal data and time series of geophysical products.

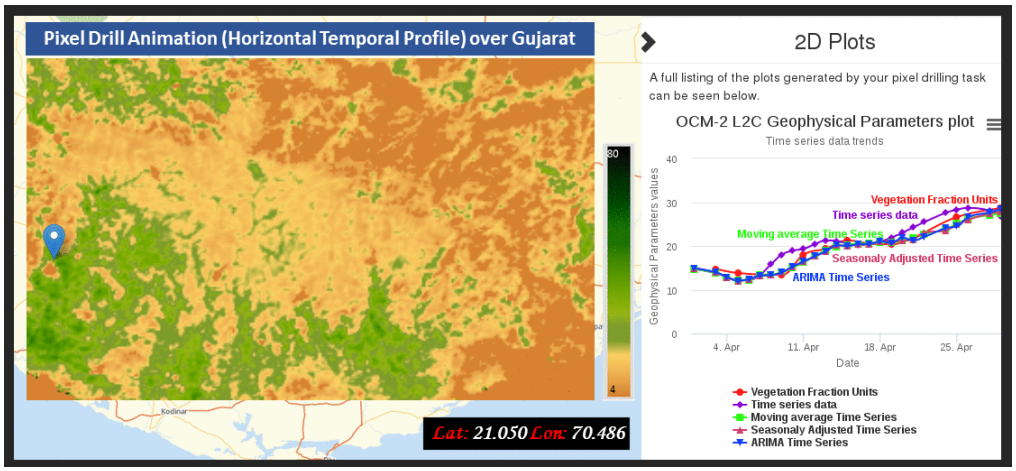

Figure 6: Pixel drilling and Trend Analysis Application Integrated with Web User Interface

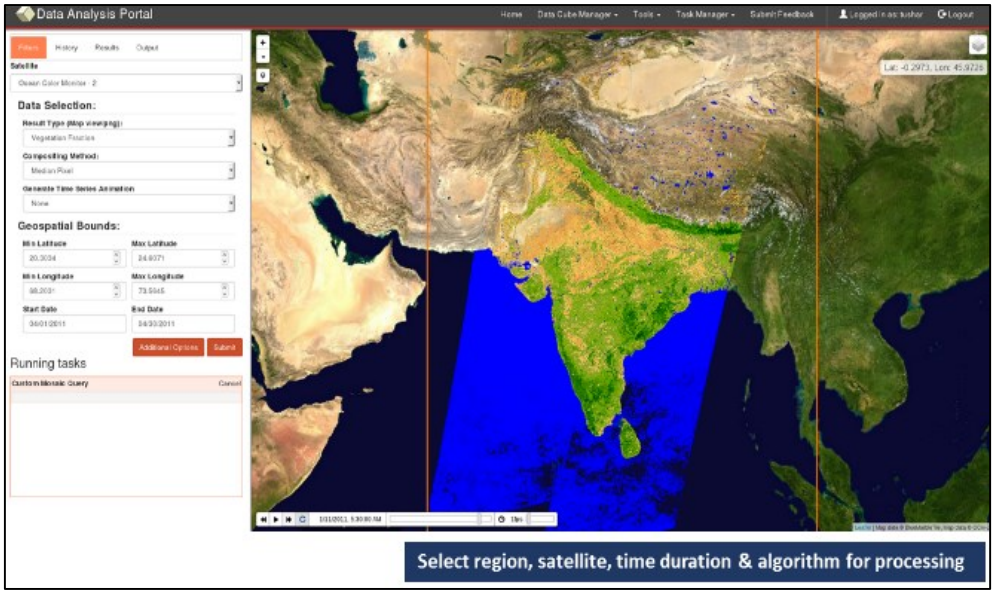

Figure 7: Web User Interface with a static monthly composite layer of Vegetation fraction and Land surface water product

4. Conclusions

Data Cubes are revolutionizing the way users can work with EO data. It is a disruptive technology that is significantly transforming the way that users interact with EO Data. It has the potential to routinely transform Earth Observations into useful and actionable information for users. To reduce the processing burden on users, generating Analysis Ready Data is a fundamental requirement. ARD products minimize the time and scientific knowledge required to access and prepare satellite data having consistent and spatially aligned calibrated surface reflectance observations. The proposed approach makes use of the APGI framework to build interoperable data processing chains for generating ARD products. This methodology has been tested in building the OCM-2 Data Cube, a country scale DC for monitoring the environment in space and time, and has allowed to efficiently download, pre-process, and ingest thousands of Oceansat scenes in a couple of days. The Datacube infrastructure allows for the integration of EO data from multiple satellites and as such, our future work focusses primarily on addition and assimilation of more and more data to the cube. Currently, the Resources at-2 LISS-3 ARD is being generated and invested in the existing infrastructure.

References

Baumann, P., A.P. Rossi, O. Clements, A. Dumitru, B. Evans, P. Hogan, … J. Wagemann (2016a). Fostering cross-disciplinary earth science through datacube analytics (p. 32).

Baumann, P., P. Mazzetti, J. Ungar, R. Barbera, D. Barboni, A. Beccati, … S. Wagner (2016b). Big data analytics for earth sciences: The earth server approach. International Journal of Digital Earth, 9(1), 3–29.

Dask Development Team (2016). Library for dynamic task scheduling. URL https://dask.org

Giuliani, G., H. Dao, A. De Bono, B. Chatenoux, K. Allenbach, P. De Laborie and P. Peduzzi (2017). Live monitoring of earth surface (LiMES): A framework for monitoring environmental changes from earth observations. Remote Sensing of Environment.

Hoyer, S. and J. Hamman (2016), Journal of Open Res. Software, xarray: {N-D} labelled arrays and datasets in Python.

Lewis, A., L. Lymburner, M.B.J. Purss, B. Brooke, B. Evans and S. Oliver (2016). Rapid, highresolution detection of environmental change over continental scales from satellite data – The Earth Observation Data Cube. International Journal of Digital Earth, 9(1), 106–111.

Rockstrom, J., W. Steffen, K. Noone, A. Persson, F.S. Chapin, E. Lambin and J. Foley (2009). Planetary boundaries: Exploring the safe operating space for humanity. Ecology and Society, 14(2).

Shukla, T., S. Roy and D. Dhar (2018), Improved SIFT-based Geometric Accuracy Improvement of Oceansat-2 OCM Imagery for Time-series Datacube, ICRIEECE – 2018.

Wulder, M. A., J.C. White, S.N. Goward, J.G. Masek, J.R. Irons, M. Herold, … C.E. Woodcock (2008). Landsat continuity: Issues and opportunities for land cover monitoring. Remote Sensing of Environment, 112(3), 955–969.