



SAPS

3. Saps

Description

SAPS (SEB Automated Processing Service) is a service to estimate Evapotranspiration (ET) and other environmental data that can be applied, for example, on water management and the analysis of the evolution of forest masses and crops. SAPS allows the integration of Energy Balance algorithms (e.g. Surface Energy Balance Algorithm for Land (SEBAL) and Simplified Surface Energy Balance (SSEB)) to compute the estimations that are of special interest for researchers in Agriculture Engineering and Environment. These algorithms can be used to increase the knowledge on the impact of human and environmental actions on vegetation, leading to better forest management and analysis of risks.

Architecture

Figure 11 shows the architecture of SAPS. This architecture is automatically deployed, configured and managed by EC3. All the SAPS components run on a K8s cluster, so the location of each component depends on the K8s scheduler. The only component that needs to run in the front machine of the cluster is the Dashboard, so it can be exposed using the public IP of the front to the users.

As shown in figure 11, the user interacts with the system through the Dashboard, a web-based GUI that serves as a front-end to the Submission Dispatcher component. Through the Dashboard, the user, after successfully logging in, can specify the region, the period that he/she wants to process, as well as the particular Energy Balance algorithm that should be used. The execution consists of a three-stage workflow: input download,

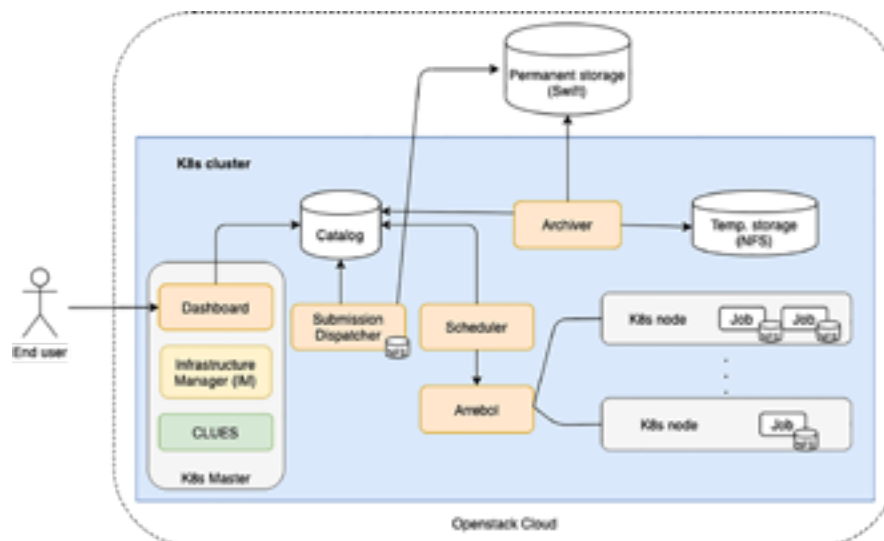


Figure 11 - Architecture of SAPS deployed on a Kubernetes (K8s) cluster by EC3

input preprocessing, and algorithm execution. With this data, the Dashboard creates the processing requests and submits them sequentially to the Submission Dispatcher. Each request generated corresponds to the processing of a single scene. The Submission Dispatcher creates a task associated with the request in the Service Catalog database (PostgreSQL). This element works as a communication channel between all SAPS components. Tasks have a state associated with them that is used to indicate which component should act next in the processing of the task.



The Scheduler component is in charge of orchestrating the created tasks through various states until they finish. It uses Arrebol to create and launch the tasks on the K8s cluster as Kubernetes Jobs. A Job downloads the appropriate Docker image from Docker Hub and starts its execution. Input and output files are stored on a Temporary Storage NFS that is accessible to all Jobs running at the cluster. Arrebol monitors all active Jobs to find out the status of the executions, and updates the state of each task in the Service Catalog, accordingly. The Archiver component collects the data and metadata generated by tasks whose processing has either successfully finished or failed. The associated data and metadata are copied from the NFS Temporary Storage, using an FTP service, to the Permanent Storage, which uses the Openstack Swift distributed storage system, where they are made securely and reliably available to the users.

Through the Dashboard, the user can also have access to the output generated by completed requests. The interface to access the output data uses a world map. A heat-map, segmented based on the standard tiles used by the Landsat family of satellites, is superimposed to the world map. The heat-map gives an idea of the number of scenes for each Landsat tile that have already been processed.

EOSC Services

In the context of EOSC-SYNERGY, SAPS is being integrated with several services offered by EOSC. This integration will facilitate European scientists to exploit the evapotranspiration estimation services from remote sensing imagery. Currently, the service relies on the next EOSC Services:

- **EC3 and IM tools:** both are services used by SAPS to deploy and configure a Kubernetes cluster automatically with SAPS running on it. Also, EC3 is used to manage the elasticity of the K8s cluster automatically. These tools facilitate the deployment and management of SAPS service.
- **EOSC computing resources:** through EC3 and IM, the SAPS service is deployed on top of a virtual elastic K8s cluster, that may rely on EOSC federated cloud computing resources or in on-premises solutions like Openstack.
- **EGI Check-in:** through EC3 portal. To deploy a cluster with SAPS, we use the EC3 portal of EOSC-SYNERGY, which is already integrated with EGI Check-in. So, to access a SAPS cluster, you should identify yourself with EGI Check-in. We will also consider integrating EGI Check-in directly on the SAPS dashboard in the next year of the project, for a fixed production endpoint of SAPS.

3. Saps

Service Endpoint

The SAPS dashboard is designed to facilitate the deployment and management of Landsat analysis tasks. Figure 12 shows the appearance of it for (a) submission of a new processing request and (b) access to the output data.

To access the SAPS Dashboard, a user is requested, as shown in figure 13. Internally, this is managed by local authorisation tokens. This solution is limited to the application, and we plan to study the viability of integrating EGI Check-in.

EOSC Synergy does not provide an endpoint of the SAPS service. Instead, you can deploy your own instance easily through the EC3 EOSC Synergy portal (<https://servproject.i3m.upv.es/ec3-synergy/index.php>), selecting as LRMS 'Kubernetes' and as Software package 'SAPS', or you can directly use the EC3 recipe and YAML files from the <https://github.com/amcaar/saps-docker> GitHub repository to deploy your own instance and properly configure the access to Openstack Swift storage solution. However, we also plan to offer a production instance of SAPS ready for non-advanced users, that will be available in the next months below the SAPS VO (saps-vo.i3m.upv.es).



(a) Submission of new processing requests.



(b) Access to output data.

Figure 12 - Snapshot of the SAPS interface

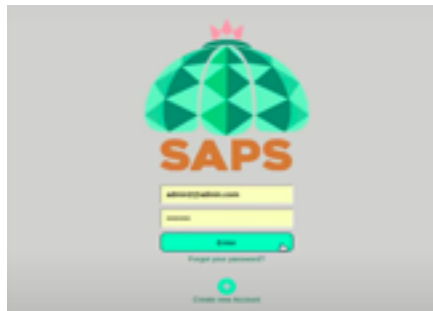


Figure 13 - Snapshot of the SAPS interface



Demonstration Video

We have prepared a demonstration video where we not only show SAPS in action, but its integration with some of the EOSC services (EC3 and IM) and the developments we have performed during the first year of the project (mainly the integration with Kubernetes).

The demo is mainly divided in three parts. The first part of the video shows the deployment of the SAPS application on top of an elastic Kubernetes cluster by terms of EC3. The video shows the command needed to deploy the cluster by using EC3 CLI and how to connect to it. Once inside the cluster, in the video we show how the SAPS microservices are deployed in Kubernetes and wait for an initial working node to run. This action is automatically done by CLUES, the elasticity manager of the cluster. On the second part of the video, we access SAPS Dashboard and show a bit the graphical interface it offers to create and monitor the status of the tasks. We also explain the required parameters SAPS asks the user to create a new landsat workflow analysis. Finally, the third part of the video shows an example execution created in the SAPS dashboard, and how the elastic Kubernetes cluster adapts automatically its size to cope with the 62 tasks that compose the workflow. The last part of the video shows the graphical interface that SAPS offers to access the output of the previously executed workflows, that is based on a world heat map.



Click the image to view the video

- **SAPS (SEB Automated Processing Service)** is a service to estimate Evapotranspiration (ET) and other environmental data that can be applied on water management and the analysis of the evolution of forest masses and crops.
- **SAPS** is one of the **Thematic Services** of the **EOSC Synergy** project.
- **Objective:** provide wider access to knowledge on the impact of human and environmental actions on vegetations, leading better forest management and analysis of risks.