My Sustainable Forest

Earth observation services for silviculture

D3.1 INTEGRATION AND VALIDATION PLAN

OPERATIONAL SUSTAINABLE FORESTRY WITH SATELLITE-BASED REMOTE SENSING

Project no.	776045
Project title	Operational sustainable forestry with satellite-based remote sensing
Project acronym	MySustainableForest
Start date of project	1 November 2017
Duration of project	36 months
Deliverable	D11 D3.1
Due date of deliverable	30/05/2018
Actual submission date	06/09/2018
Organisation name of lead contractor for this deliverable	GMV EC REA
Dissemination level	Public

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776045

Code:	D11 D3.1
Version:	V1
Date:	06/09/2018
Internal code:	GMV 21922/18 V1/18

TECHNICAL REFERENCES

Project Acronym	MySustainableForest
Project Title	Operational sustainable forestry with satellite-based remote sensing
Project Coordinator	Julia Yagüe GMV mjyague@gmv.com
Project Duration	1 Nov 2011 – 30 Oct 2021 (36 months)
Grant Agreement	No. 776045

Deliverable Number	D11 D3.1
Dissemination Level	PU
Workpackage	WP3 Service Development and Engineering
Task	T3.3 Platform and Processing Chain Preparatory Activities (primarily) T3.1 Standards and Interoperability (Early Service Readiness/ Service Preparation) T3.4 Remote Sensing Algorithms, Models Tailoring and Data Fusion (Early Service Readiness/ Service Preparation) T3.5 Workflow Integration Engineering (Early Service Readiness/ Service Preparation) T3.6 Service Chain Testing (Early Service Readiness/ Service Preparation)
Lead beneficiary	GMV
Contributing beneficiary (ies)	RAIZ CFRI UFE FORESNA FOAL CNPF MADERA+ FORA EFI GUK
Due date of deliverable	30/05/2018
Actual submission date	06/09/2018

DOCUMENT STATUS SHEET

Version	Date	Pages	Changes
V1	06/09/2018	19	First version of the document

TABLE OF CONTENTS

1. INTRODUCTION 6
1.1. PURPOSE 6
1.2. SCOPE 6
2. APPLICABLE AND REFERENCE DOCUMENTS 7
2.1. APPLICABLE DOCUMENTS 7
2.2. REFERENCE DOCUMENTS 7
2.3. ACRONYMS AND DEFINITIONS 7
3. SYSTEM OVERVIEW 9
3.1. SYSTEM COMPONENTS 9
3.1.1. GEOBROWSER 10
3.1.2. EOM BACKEND APPLICATION 10
3.1.3. EOM WEB FRONTEND 10
3.1.4. DATA MANAGER 10
3.1.5. SERVICES AND PRODUCTS CATALOGUE 11
3.1.6. DATA PROCESSORS 11
3.1.7. PROCESSING ORCHESTRATOR 11
4. INTEGRATION AND VALIDATION STRATEGY 12
4.1. INTEGRATION PROCESS 12
4.1.1. ROLES AND TEAMS 12
4.1.2. INTEGRATION AND VALIDATION WORKFLOW 12
4.1.3. INTEGRATION AND VALIDATION INFRASTRUCTURE 13
4.2. VALIDATION PROCESS 13
5. INTEGRATION TEST CASES 14
5.1. PLATFORM LEVEL INTEGRATION 14
5.1.1. MSF_EOM_INT_TEST_001_START_STOP 14
5.1.2. MSF_EOM_INT_TEST_002_DATAMANAGEMENT 15
5.1.3. MSF_EOM_INT_TEST_003_GEOSERVER_EOMGIS_CATALOGUE 15
5.2. BUILDING BLOCK LEVEL 16
5.2.1. MSF_EOM_INT_TEST_004_SINGLE_ALGORITHM_INTEGRATION 16
5.3. SERVICE LEVEL 16
5.3.1. MSF_EOM_INT_TEST_005_PIPELINE_INTEGRATION 16
6. VALIDATION TEST CASES 18

LIST OF TABLES AND FIGURES

Table 1-1. WP3 deliverables and coherence of contents 6
Table 2-1. Applicable Documents 7
Table 2-2. Reference Documents 7
Table 2-3. Acronyms 7
Table 2-4. Definitions 8

Figure 3-1. Earth Observation Merlin (EOM) Platform System Components 9
Figure 3-2: EOM Platform, Geobrowser 10
Figure 4-1: Integration Workflow 13

1. INTRODUCTION

Work performed under **WP3 Service Development and Engineering** strives:

1. To define new remote sensing algorithms or refine existing ones.
2. To integrate all prototyped components and needed technological infrastructure to build the platform – technological infrastructure- which will allow an automated service.
3. To check that each component meet its specification.
4. To prepare reference datasets for testing and training purposes.
5. To verify that the integrated system/service works as requested by End-Users.

Five deliverables shall be issued under these objectives as listed in Table 1-1

Table 1-1. WP3 deliverables and coherence of contents

WP1 Deliverable			Contents
D11	D3.1	Integration and Validation Plan	■ Defines the **procedures for testing** the integration of the different **service components** ■ Defines the **procedures for verifying** that the **integration of the service components complies with the Service Requirements.**
D17	D3.2	Remote Sensing Algorithms and Models Trade-off	■ Documents the new or refined remote sensing algorithms and models in order to fulfil the End-Users needs.
D18	D3.3	Integration and Validation Report (v1)	■ Reports the results of the Integration and Validation tests upon the **Early Service**
D31	D3.4	Integration and Validation Report (v2)	■ Reports the results of the Integration and Validation tests upon the **Enhanced Service**
D38	D3.5	Integration and Validation Report (v3)	■ Reports the results of the Integration and Validation tests upon the **Advanced Service**

1.1. PURPOSE

This document corresponds to **Deliverable D11 D3.1 Integration and Validation Plan** of MySustainableForest H2020 Project.

The Integration and Validation Plan defines the procedures for testing the integration of the different service components and for verifying that they comply with the Service Requirements. This document describes the approach and the steps for integration and validation of the technological infrastructure that supports the MSF project and services, as well as the high-level schedule of the activities associated with it.

1.2. SCOPE

This document is structured according to:

- Section 1, (present chapter) defines the purpose of the document within the WP objectives and the overall purpose of WP deliverables.
- Section 2, includes the list of applicable documents and additional references to be taken into account during the project life cycle. It also contains the definitions and acronyms used in this document.
- Section 3, recalls the EOM system components through a brief overview, (also refer to [RD3])
- Section 4, details the steps (method) that shall be taken for the integration and validation of the EOM platform.
- Section 5, lists the integration test cases to be followed concerning the platform, the building blocks (processing algorithms) and services.
- Section 6, lists the validation test cases to be followed. .

2. APPLICABLE AND REFERENCE DOCUMENTS

2.1. APPLICABLE DOCUMENTS

The following documents, of the exact issue shown, form part of this document to the extent specified herein. Applicable documents are those referenced in the Contract or approved by the Approval Authority.

Table 2-1. Applicable Documents

Ref.	Title	Code	Version	Date
[AD.1]	Grant Agreement Nº 776045—MySustainableForest	Ares(2017)5215238	1.0	25/10/2017
[AD.2]	D01_D1.1_Project Plan	D01_D1.1	1.0	30/11/2017

2.2. REFERENCE DOCUMENTS

The following documents, although not part of this document, amplify or clarify its contents. Reference documents are those not applicable and referenced within this document. They are referenced in this document in the form [RD.X]:

Table 2-2. Reference Documents

Ref.	Title	Code	Version	Date
[RD.1.]	D03_D2.1_End Use User Cases	GMV 20463/18 V1/18	V1	
[RD.2.]	D09_D2.2_Service Requirements and Validation of Use Cases	GMV 21920/18 V1/18	V1	04/09/2018
[RD.3.]	D10_D2.3_Architectural Design	GMV 21921/18 V1/18	V1	06/09/2018

2.3. ACRONYMS AND DEFINITIONS

The following acronyms have been used across this document:

Table 2-3. Acronyms

Acronym	Full term
AD	Applicable document
AOI	Area of Interest
API	Application programming interface
EO	Earth Observation
EOM	Earth Observation Merlin Platform
GDB	Geo Data Base
GIS	Geographic Information System
MSF	MySustainableForest Project
OGC	Open geospatial Consortium
Px	Product Number
RD	Reference Document
Sx	Service number

Acronym	Full term
WP	Work Package

Table 2-4. Definitions

Concept/Term	Definition
Service	In the context of MySustainableForest project, a "service" is an assembled system of products that support Sustainable Forest Management (SFM) and good forest practices. The project focusses upon six SFM services : 1. Forest Site Characterization 2. Wood Characterization 3. Biomass and CO2 stocking 4. Forest Condition 5. Ecosystem vulnerabilities 6. Socioeconomic Functions and Conditions
Product	In the context of MySustainableForest project, a "product" is an independent and complete component of a service; products are complete in themselves and provide a full set of information. However, single products, however valuable, have a limited usage and scope within SFM practices. Products are defined by the input data, the algorithms that process those data and the output type required by the end user E.g.: Service 1 "Forest site Characterisation" counts with 12 products :
Building Block	In the context of MySustainableForest project, "building blocks" are either data items needed to achieve a product or processing components of products. Because most processing components implement a mathematical model, some of them are simply referred as "models"::

PRODUCT ID №	PRODUCT NAME
S1 P1	Forest mask
S1 P2	Stand delineation
S1 P3	Forest infrastructures
S1 P4	Main forest types
S1 P5	Stand height
S1 P6.1 S1 P6.2	Forest age year of reference Forest age biannual updates
S1 P7.1 S1 P7.2	Burnt scars reference Burnt scars 6- months updates
S1 P8.1 S1 P8.2	Clear cuts reference Clear cuts bi-annual update
S1 P9	Elevation
S1 P10	Slope
S1 P11	Aspect
S1 P12	Site Index

- Auxiliary data
- Derived products
- EO input data
- LIDAR-based models
- Non EO input data
- Satellite-based models
- Socio-economic models
- Wood quality models

3. SYSTEM OVERVIEW

The My Sustainable Forest relies on GMV EOM platform for the generation and the provision of data products to user. EOM (EO Merlin) is a platform of Earth Observation data exploitation that include components for data processing, visualization and cataloguing, between others. The platform will be tailored and configured to host the MSF services and to provide the functionality that is required by the project. Although other instances of the platform are used internally in GMVs, the platform for MSF will follow the integration and validation flow as if it was a new development.

The EOM platform is designed to provide a set of functionalities, both user-centric and data-centric:

- **Data Management**. The Data Management comprise the means to retrieve, store and manage the data used in the platform. Several components are part of the data management since data can brought to the platform in various ways and different kind of data exists (e.g. files, tabular data in a database etc.)
- **User Management**. EOM platform is multi-tenant that means that provides isolation of data and users groups. Hence several projects can leverage the EOM platform as it was their own. Users and groups, along with their privilege, can be defined
- **Data Visualization** While virtually any kind of data can be stored and processed in the EOM, the platform offers tools for geospatial data visualization: geo-located data and geo-referenced images can be overlayed on top of maps and presented to the user through specific geospatial information tools. Some of the tools provided by the platform also allows users for data exploration and filtering certain kind of data.
- **Data Processing** One of the core objective of the EOM platform is to offer users a scalable data processing capability: users can either process their own data using available processing blocks on the platform or integrate their own processing blocks using a specific interface.

3.1. SYSTEM COMPONENTS

The high-level architecture is depicted in the following diagram. Note that the architecture considers a fully-featured instance of the EOM platform. Certain features may be disable for a specific project or user group

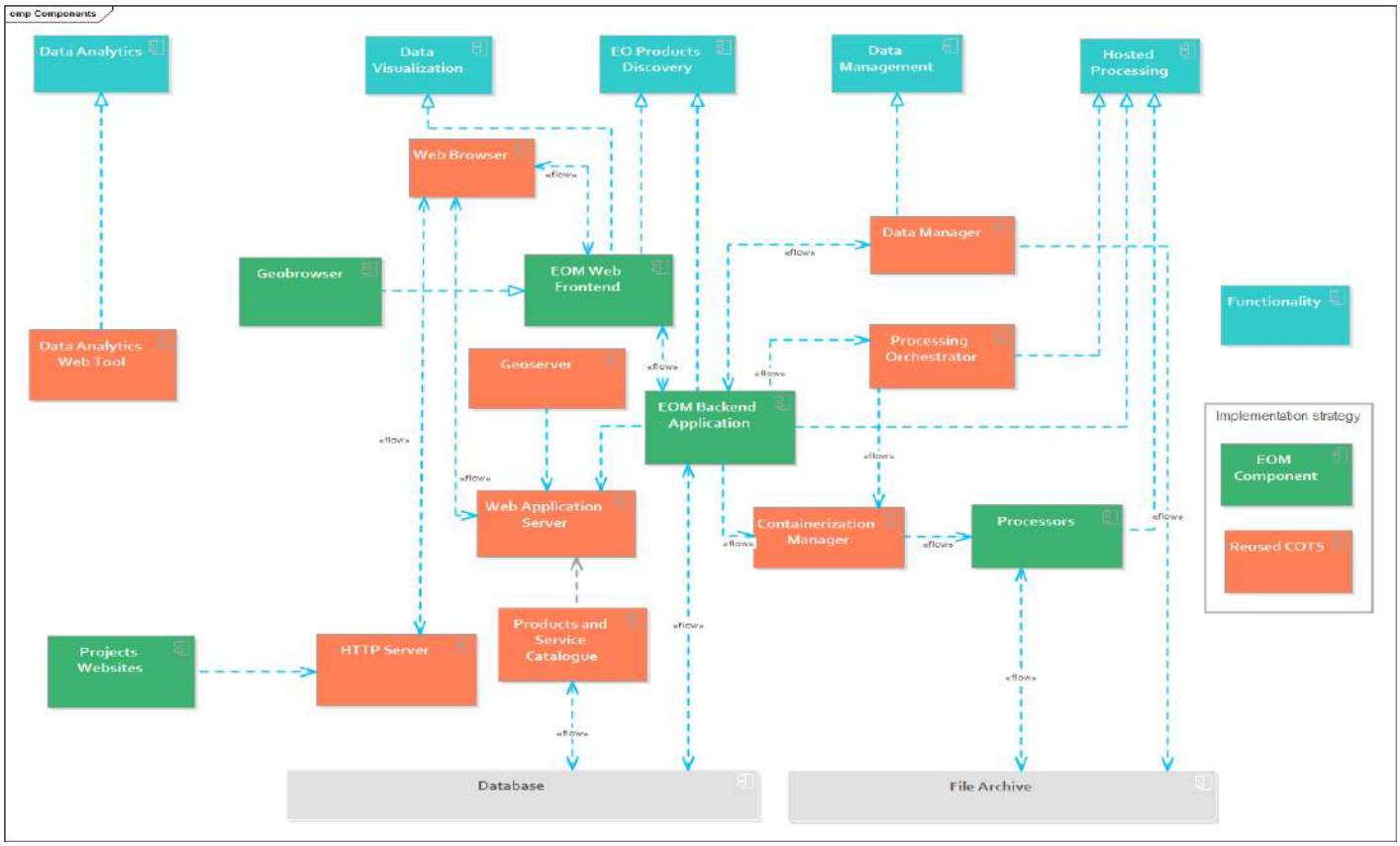

Figure 3-1. Earth Observation Merlin (EOM) Platform System Components

3.1.1. GEOBROWSER

The Geobrowser allows for geospatial data to be visualized on a web browser using a GIS viewer with custom views tailored for each project.

A sample view of the Geobrowser using dummy data is reported in document D2.2 "Architectural Design" [RD3]. Note that the Geobrowser will be tailored and properly configured once that data is integrated into the platform. A sample is included in **¡Error! No se encuentra el origen de la referencia.** below.

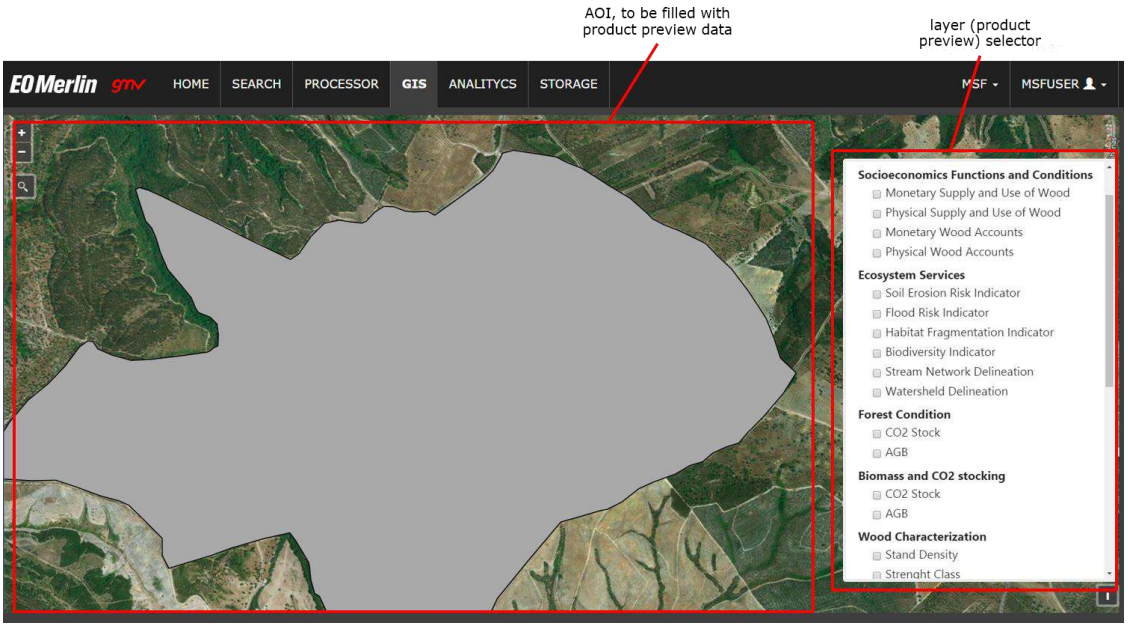

Figure 3-2: EOM Platform, Geobrowser

3.1.2. EOM BACKEND APPLICATION

The EOM backend application is the core of the EOM platform as it is responsible for generating the user interface and control the overall user experience. The backend is composed of several parts, mainly integrating the user management component is responsible for user authentication, authorization and isolation, since different data can be accessed exclusively by certain users.

3.1.3. EOM WEB FRONTEND

The EOM Web frontend is the entry point to the EOM platform for the user. It integrates several sections corresponding to the different functionalities and components that the platform offers to the user:

- The project homepage
- The geospatial catalogue frontend
- The processing dashboard, through which the users who process data can build processing chains and execute them ·
- The data management frontend, through which the user can exchange files
- The Geobrowser
- The Data analytics frontend

3.1.4. DATA MANAGER

The data manager interacts with the user to allow data retrieval and injection into the platform. On one side is provides a web interface similar to Google Drive or Dropbox for the user. On the other side, the data access component is responsible for storing and retrieving the data items in the underlying infrastructure (e.g. filesystem or even remotely through the network).

3.1.5. SERVICES AND PRODUCTS CATALOGUE

The catalogue store the definition of the data products and services and allows users or other components to search them and to retrieve the information associated to them. The catalogue is itself a set of components: the EOM platform offers its own internal catalogue for datasets and processing blocks. EOM can be easily customized to integrate **Geonetwork** and offer users and external system a OGC-compliant catalogue.

3.1.6. DATA PROCESSORS

The Data Processors are blocks that transform data. In the EOM platform, data processors (also referred as *processing blocks*) are encapsulated as Docker images and executed in a specific sequence. Such sequence is also called *pipeline*. Pipelines are handled by the **Processing Orchestrator**.

The processing blocks are pulled from a Docker repository. Once installed in the processing nodes, they are launched in background and operations within the Dockers are performed upon request by the processing orchestrator.

The component in charge of deploying Dockers in the different processing nodes is **Kubernetes**. Kubernetes allows to organize the containers execution on different nodes basing on the available resources, needed redundancy levels, replica policy. It allows to scale the processing environment by starting additional instance of the containers and to route the service requests to specific Dockers according to load balancing or other policies.

3.1.7. PROCESSING ORCHESTRATOR

Typical data transformation pipelines are made of different independent and element processing operations chained together in a specific sequence and with strict dependencies between the stages.

In order to bring a standardized approach to dependency handling and the scheduling of different operations, EOM integrates a data-agnostic, state of the art processing orchestrator that is responsible of executing the intermediate processing steps to generate the output data products or any other kind of output that the EOM platform is used for. The default processing orchestrator in EOM is **Apache Airflow** but processors can be chained together using simpler methods (e.g. scripts) when chains are small and fast enough to be executed in a single node efficiently. Since several independent nodes are used for processing, Airflow pipelines will rely on Kubernetes to execute the independent processing steps in the processing cluster managed by it. Kubernetes will then forward the execution request to the appropriate nodes and Docker images.

4. INTEGRATION AND VALIDATION STRATEGY

The integration of the platform in the MSF project is done in several stages:

- The specific instance of the platform for MSF is integrated by generating a common configuration and enabling the components that will be part of it.
- Once the empty platform is integrated, the data is integrated, that means configuring the data buckets and connectors so that data can be uploaded, downloaded and remotely accessed towards the external data repositories
- Once data integration is done the processors are integrated into the platform. This is done in two steps: first the containers are created from the algorithms and the Docker containers are deployed in the platform, configuring the appropriate pipelines.
- Finally, end-to-end tests are done to assure that each end data product can be generated using updated data that is manually or automatically stored in the platform.

4.1. INTEGRATION PROCESS

The integration process spans over most project lifetime according to the workflow depicted in Figure 4-1 below.

4.1.1. ROLES AND TEAMS

Several teams are involved in the integration process:

- The *platform integration team* that is in charge of creating, configuring and maintaining the instance of the EOM platform for MSF
- The *service development team* is composed of technical people from the partners developing the service algorithms in their own premises and with minimum technical constraints imposed by the platform.
- The *service integration team* will be mostly composed by GMV technical personnel providing support for integrating the algorithms into the platform and for testing the outcome of such integration. It is to be noted that part of the integration (e.g. Containerization) could be done by the service development teams as part of the algorithms packaging and delivery process

4.1.2. INTEGRATION AND VALIDATION WORKFLOW

The integration workflow is divided into two big blocks, as depicted in Figure 4-1 below, corresponding to the integration of the (i) platform architecture and the (ii) integration of the [algorithms that enable the automatic processing of] services:

- The platform integration team integrated the platform components in order to allow the services to be integrated. The outcome of this activity is a working platform instance with all the functionality deployed, activated and tested.
- The platform is configured through the complex concatenation of building blocks that are executed on their inputs to produce the different output products [RD2]. This concatenation is also called pipeline. The pipeline configuration is part of the early stage of the service integration, even without the building blocks being released by the partners.
- The service development works on the algorithms and creates the deliverable releases of the building blocks.
- Once a version of a certain algorithm (building block) is released, it is integrated in the platform and integration tests for single building blocks are executed.
- Once all the building blocks of a certain pipeline are available, the full pipeline is execute with input to produce the output data product.
- When all the pipelines corresponding to the output products of a service are integrated and individually tested, the services can be considered integrated.
- At this point, the service validation takes place. Service validation essentially consists in assessing whether data products generated for specific services on a specific area of interest are valid. See section "Validation Process" for more details.

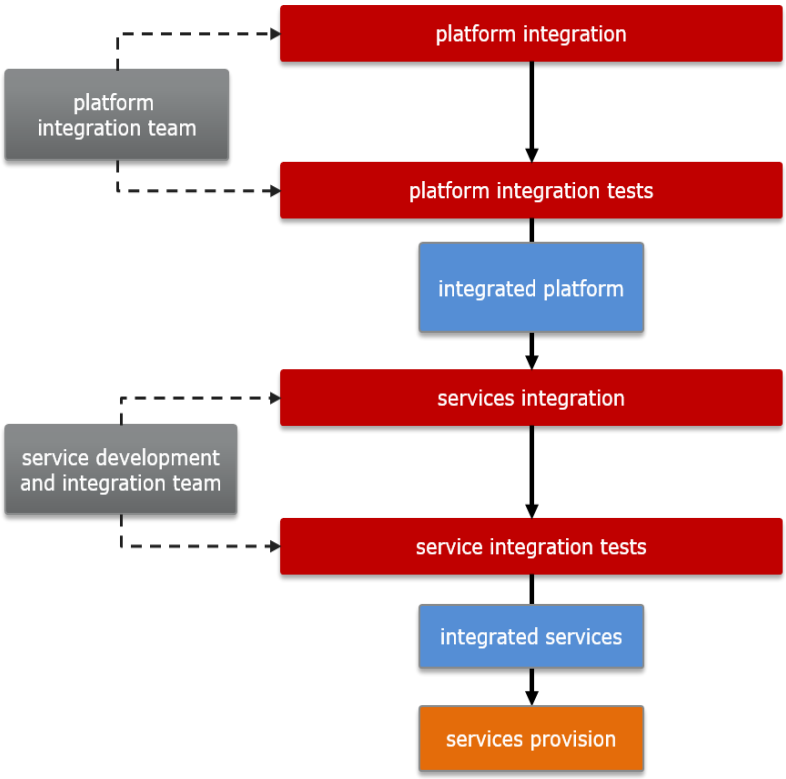

Figure 4-1: Integration Workflow

4.1.3. INTEGRATION AND VALIDATION INFRASTRUCTURE

The infrastructure for Integration and Validation is composed by several environments:

- The service development environment, in which the algorithms are developed in arbitrary programming languages, frameworks etc. The service development environment is the collection of environments belonging to partners that develop processing services.
- The integration environment is aimed at integrating the EOM platform and the services on it. It not only serves as an environment for initial platform integration but is used to integrate and validate successive releases of algorithms. Once the platform and the services are integrated and validated, they are migrated to the production environment, in which they are accessed by the end users.
- The production environment is composed by a fully integrated and verified instance of the platform with the services deployed and validated on top of it. It is only modified when the appropriate integration and validation activities have been performed in the integration platform.

4.2. VALIDATION PROCESS

The validation process will be mostly carried on by the Service Development team with the support of the Service Integration team once the output have been produced by the platform. It will be in great part a domain-specific activity, in which output data will be checked against reference output according to the specific service evaluation metrics identified in the corresponding project activities.

5. INTEGRATION TEST CASES

Integration test cases are designed targeting different levels:

- At **platform** level, in which high-level platform functionalities are verified by checking that the corresponding components are correctly deployed.
- At **building block** level, in which it is checked that the mechanism to process data and integrate custom building block is working
- At **service** level, in which it is checked that complete end-to-end processing chains including the execution of a complex combination of building blocks on different pieces of data produce the output products, and ultimately that the latter are available to the users using the appropriate delivery mechanism.

Below a collection of high-level integration test cases are reported, to be further detailed, split or complemented by others if considered convenient. Note that in some cases the input data and the parameters associated with each of them mostly depend on the specific design of the building blocks to be used.

5.1. PLATFORM LEVEL INTEGRATION

5.1.1. MSF_EOM_INT_TEST_001_START_STOP

5.1.1.1. Test Objective

The objective of this test is to verify that the platform components are correctly installed and can be accessed through the web interfaces

5.1.1.2. Prerequisites

- The platform components have been installed: EOM Portal, Catalogue, EOMGIS, Geoserver, Data Management
- The platform instance has been configured with a single project manager user who has access to all the platform functionality
- The platform instance has been configured to link to the project website
- The Geoserver has been configured with an single GIS layer containing the different AOIs
- The Catalogue has been configured with an empty catalogue
- The EOMGIS has been configured to target the GIS layer in the Geoserver
- The Data Management has been configured with a single shared workspace

5.1.1.3. Test Steps

T	Description	Expected Results
1	Launch all the EOM components	No errors are raised
2	Connect to the platform homepage and login with project manager credentials	The MSF EOM homepage is displayed and the tool tab displays all the tools deployed in the instance
3	Click one by one on the tabs and check that for each tool its homepage is displayed	Each tool homepage is displayed
4	Log out from the platform	The log out is successful
4	Stop all the EOM components	All the EOM components can be stopped successfully

5.1.2. MSF_EOM_INT_TEST_002_DATAMANAGEMENT

5.1.2.1. Test Objective

The objective of this test is to verify that the Data Management component is working and allows different workspaces to be created and connected to remote data repositories such as ftp servers

5.1.2.2. Prerequisites

- MSF_EOM_INT_TEST_001_START_STOP

5.1.2.3. Test Steps

T	Description	Expected Results
1	Login with project administration in the platform	No errors are raised
2	Create representative users belonging to the different categories	No errors are raised
3	In the Data Management component create the workspaces corresponding to all the different input and output products to be stored in the platform and test read/write operations on those workspaces	No errors are raised
4	In the Data Management component create the workspaces corresponding to the remote data to be accessed using ftp/www connectors	No errors are raised
5	For each workspace with remote connection, check that data items are listed from remote directories	data is listed from remote directories

5.1.3. MSF_EOM_INT_TEST_003_GEOSERVER_EOMGIS_CATALOGUE

5.1.3.1. Test Objective

The objective of this test is to verify that the Geoserver in the EOM can be configured to serve the different layers corresponding to the raster and vector output products to be visualized through the GIS browser of the platform, and to verify that each layer can be visible to users specifically linked to certain AOIs.

5.1.3.2. Prerequisites

- MSF_EOM_INT_TEST_001_START_STOP
- MSF_EOM_INT_TEST_002_DATA_MANAGEMENT

5.1.3.3. Test Steps

T	Description	Expected Results
1	Login with project administration in the platform	No errors are raised
2	In the Geoserver configuration interface, add the layers corresponding to all the products that will have a GIS visualization in the platform and put dummy data for such layers	The GIS layers can be previewed in the Geoserver interface
3	In the Catalogue, add the records corresponding to the data products to be visualized in the GIS interface, for each service and for each AOI	The added data products and services can be visualized
4	Create one user for each AOIs (subproject) to be accessed	The users can be created
5	For each user, login and open the GIS interface	The layers corresponding to the products for the specific AOIs are displayed and selectable
6	For each user, login and open the GIS interface	The layers corresponding to the products for the specific AOIs are displayed and selectable

5.2. BUILDING BLOCK LEVEL

5.2.1. MSF_EOM_INT_TEST_004_SINGLE_ALGORITHM_INTEGRATION

5.2.1.1. Test Objective

The objective of this test is to verify that a single building block with an algorithm can be integrated and executed in the platform.

5.2.1.2. Prerequisites

- MSF_EOM_INT_TEST_001_START_STOP

5.2.1.3. Test Steps

T	Description	Expected Results
1	Login with project administration in the platform	No errors are raised
2	In the processing interface, create a new pipeline with a single dummy building block	The pipeline can be created successfully
3	Upload dummy data in a dummy data workspace	Data is successfully uploaded
4	Create a dataset that includes the dummy data workspace previously created. Review the dataset content	The dataset is successfully created
5	Manually associate the created dataset with the dummy building block and execute it	The building block is successfully executed, a job has been created and registered in the processing interface and the output data is stored in the Job output directory

5.3. SERVICE LEVEL

5.3.1. MSF_EOM_INT_TEST_005_PIPELINE_INTEGRATION

5.3.1.1. Test Objective

The objective of this test is to verify that a full pipeline of various algorithm can be integrated and executed, generating output products that are stored in the appropriate workspaces and GIS views.

5.3.1.2. Prerequisites

- MSF_EOM_INT_TEST_004_SINGLE_ALGORITHM_INTEGRATION

5.3.1.3. Test Steps

#	Description	Expected Results
1	Login with project administration in the platform	No errors are raised
2	In the processing interface, create a new pipeline using a tailored script for the orchestrator and marking the pipeline to be executed with the orchestrator	The pipeline is created successfully
3	In the Data Management interface, create the data workspaces for the input and output products for the pipeline	The data workspaces are created successfully

#	Description	Expected Results
4	Upload dummy data in the input workspaces	Data is successfully uploaded
5	Create a dataset that includes the input data workspaces previously created. Review the dataset content	The dataset is successfully created
6	Create a dataset that includes the output data workspace previously created	The dataset is successfully created
7	In the GIS server, configure a GIS layer as per pipeline specification.	The dataset is successfully created
8	In the Catalogue, register a product corresponding to the output product and layer	The product is successfully created
9	Manually associate the created input and output datasets with the pipeline and execute it. Review the execution log, the output workspace and the GIS layer	The pipeline is successfully executed, all the building blocks are executed in the correct order, the job has been created and registered in the processing interface, the output data is stored in the output directory and the GIS layer is updated.

6. VALIDATION TEST CASES

The validation test cases are linked with the use cases and scenario validation. Refer to document D2.2 "Service Requirements and Validation of Use Cases" [RD2]

END OF DOCUMENT

My Sustainable Forest

Earth observation services for silviculture

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 776045

www.mysustainableforest.com