Climate Change Initiative Extension (CCI+) Phase 1 New Essential Climate Variables (NEW ECVS) High Resolution Land Cover ECV (HR_LandCover_cci)

System Verification Report

(SVR)

Prepared by:

Università degli Studi di Trento Fondazione Bruno Kessler Université Catholique de Louvain Università degli Studi di Pavia Università degli Studi di Genova Politecnico di Milano Université de Versailles Saint Quentin CREAF e-GEOS s.p.a. **Planetek Italia** GeoVille Université catholique de Louvain FONDAZIONE BRUNO KESSLER UNIVERSITÀ DEGLI STUDI DI GENOVA UNIVERSITÀ DEGLI STUDI DI TRENTO planetek Geoville CREAF e-geos POLITECNICO ASI / TELESPAZIO COMPANY italia LSCE MILANO 1863

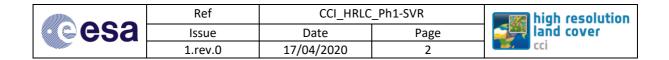
	Ref	CCI_HRLC	_Ph1-SVR	migh resolution
esa	Issue	Date	Page	and cover
	1.rev.0	17/04/2020	1	cci

Changelog

Is	sue	Changes	Date
0.	1	First version.	01/03/2020
1.	0	Final version including all the test on the system	17/04/2020

Detailed Change Record

Issue	RID	Description of discrepancy	Sections	Change
N/A	N/A	N/A	N/A	N/A



Contents

1	Intro	oduction	3
	1.1	Executive summary	3
	1.2	Purpose and scope	3
	1.3	Applicable documents	3
	1.4	Reference documents	4
	1.5	Acronyms and abbreviations	4
2	Chal	lenges for System Verification	6
	2.1	Main challenges from SSD	6
	2.1.1	Challenge 1: Orchestration of resources and IaaS	6
	2.1.2	Challenge 2: Flexibility in the pipeline development	6
	2.1.3	Challenge 3: Delivery of results	6
	2.2	Verification environment	6
3	Syste	em Verification	7
	3.1	Design of the verification scenarios	7
	3.2	Design of the verification tests	8
	3.3	Verification tests	9
	3.3.1	TEST-1.1: OpenEO Process Discovery API	9
	3.3.2	TEST-1.2: OpenEO Process Creation API	9
	3.3.3	TEST-1.3: Activation of multiple resources 1	1
	3.3.4	TEST-1.4.1: Pipeline execution monitoring using Service API 1	2
	3.3.5	TEST-1.4.2: Pipeline execution monitoring using Max-ICS logs 1	2
	3.3.6	5 TEST-2.1: Pipeline Design	3
	3.3.7	TEST-2.2: Pipeline Run	8
	3.3.8	3 TEST-3-1 Metadata Display 1	9
	3.3.9	TEST-3-2 Products Display 2	1

	Ref	CCI_HRLC	_Ph1-SVR	migh resolution
Cesa	Issue	Date	Page	land cover
	1.rev.0	17/04/2020	3	cci

1 Introduction

1.1 Executive summary

Within the European Space Agency (ESA), the Climate Change Initiative (CCI) is a global monitoring program which aims to provide long-term satellite-based products to serve the climate modelling and climate user community. Permafrost has been selected as one of the Essential Climate Variables (ECVs) which are elaborated during Phase 1 of CCI+ (2018-2021).

Following the activities of user requirements updating according to Climate User Community and other users' consultations, the Consortium has defined the related HRLC products requirements accounting for technical constraints such as main data sources available, spatial and temporal coverage, software and tools for quality control. This High Resolution Land Cover (HRLC) System Specification Document defines the system architecture and the description of the first version of the system that will generate the CCI+ HRLC products over the areas of interests.

This document outlines the system verification procedures and results for the system with respect to its capabilities to meet the challenges described in paragraph 2.1.

A test scenario is defined for the system verification. Two key aspects of the processing system are tested: 1. Capability to scale automatically 2. The capability to easily design pipelines. The verification procedure makes use of known processing steps as the final processors (steps) for the generation of HRLC products are still in the integration phase. The goal of the verification is to ensure that the system is capable to on-board generic processing steps.

The scaling capability is then guaranteed also by the fact that the system is deployed on a Cloud and thus can access to a nearly infinite resource pool.

1.2 Purpose and scope

The HRLC System Verification Report defines the criteria to meet the challenges defined in the SSD and verifies the availability of them in the system.

Input to this document are the Tender Specification [AD2] and the other Applicable Document and, in particular, to meet the challenges defined in System Specification Document (SSD)

The document is organized with the following contents:

- The description of the challenges of the system (2.1) and of the test environment (2.2)
- The description of a verification scenario (3.1)
- The design of the test for the verification (3.2)
- The description and execution of the test cases (3.3)

1.3 Applicable documents

Ref. Title, Issue/Rev, Date, ID

- [AD1] CCI HR Technical Proposal, v1.1, 16/03/2018
- [AD2] CCI Extension (CCI+) Phase 1 New ECVs Statement of Work, v1.3, 22/08/2017, ESA-CCI-PRGM-EOPS-SW-17-0032
- [AD3] Data Standards Requirements for CCI Data Producers, v2.1, 02/08/2019, CCI-PRGM-EOPS-TN-13-0009
- [AD4] User Requirements Document, v1.1, 12/04/2019, CCI_HRLC_Ph1-D1.1_URD
- [AD5] Product Specification Document, v1.0, CCI_HRLC_Ph1-PSD
- [AD6] Data Access Requirement Documentm v1.0, CCI_HRLC_Ph1-DARD
- [AD7] System Requirement Document v2.0, CCI_HRLC_Ph1-SRD

	Ref	CCI_HRLC	_Ph1-SVR	migh resolution
eesa	Issue	Date	Page	land cover
	1.rev.0	17/04/2020	4	cci

[AD8] System Specification Document v1.0, CCI_HRLC-Ph1-SSD

1.4 Reference documents

- Ref. Title, Issue/Rev, Date, ID
- [RD1] The Global Climate Observing System: Implementation Needs, 01/10/2016, GCOS-200

1.5 Acronyms and abbreviations

- API Application Programming Interface
- AOI Area Of Interest
- ARD Analysis Ready Data
- AWS Amazon Web Services
- CCI Climate Change Initiative
- CRC Climate Research Community
- CMUG Climate Modelling User Group
- DIAS Data and Information Access Services
- ECV Essential Climate Variables
- ESM Earth System Models
- EVI Enhanced Vegetation Index
- FTP File Transfer Protocol
- GCOS Global Climate Observing System
- GDPR General Data Protection Regulation
- GIS Geographical Information System
- HR High Resolution
- laaS Infrastructure as a Service
- L1C Level-1C
- L2A Level-2A
- LAI Leaf Area Index
- LaSRC Landsat Surface Reflectance Code
- LC Land Cover
- LCC Land Cover Change
- LCCS Land Cover Coverage Classification System
- LCML Land Cover Meta Language
- LCZ Local Climate Zone
- LEDAPS Landsat Ecosystem Disturbance Adaptive Processing System
- LSCE Laboratoire des Sciences du Climat et de l'Environnement
- MR Medium Resolution
- NDVI Normalized Difference Vegetation Index
- OGC Open Geospatial Consortium
- OWS OGC Web Services
- PFT Plant Functional Type

Ref	CCI_HRLC	_Ph1-SVR	migh resolution
Issue	Date	Page	land cover
1.rev.0	17/04/2020	5	cci

- RS Remote Sensing
- SAR Synthetic Aperture Radar
- SFT Surface Functional Type
- SRD Software Requirements Document
- SSD Software Specification Document
- SVR Software Verification Report
- TOA Top Of Atmosphere
- URD User Requirements Document
- VM Virtual meeting
- WCS Web Coverage Service
- WFS Web Feature Service
- WMS Web Map Service
- WP Work Package

	Ref	CCI_HRLC	_Ph1-SVR	migh resolution
esa	Issue	Date	Page	land cover
	1.rev.0	17/04/2020	6	cci

2 Challenges for System Verification

2.1 Main challenges from SSD

The platform prototype scope is to bring the results of the research activities to a pre-operational level by scaling up the processing capacity in order to allow the production of massive land cover mosaics following GCOS requirements. In practice, having to deal with several TB of data (hundreds) means that the concept of pre-operational is not applicable and that the system must provide a huge capability to scale the processing. For this reason, some constrains and requirements coming from the SRD are dealing with the capability of the IaaS platform chosen for the execution of the production.

2.1.1 Challenge 1: Orchestration of resources and IaaS

The orchestration platform, described in the following paragraphs, is based on the concept of big-data processing and reactive pipeline execution (Reactive Manifesto, <u>https://www.reactivemanifesto.org/en</u>) and is based on Max-ICS technology (Max-ICS by Earthlab Louxembourg, <u>http://www-max-ics.earthlab.lu/</u>) which is part of e-GEOS CLEOS platform (currently used internally but soon to be released as a service).

The challenge is to perform a test of scalable processing under known conditions this means that a known processor will be used to do some basic data processing to verify the scalability. The verification is done by

• Verifying the activation of multiple resources as requested by the designed test on a single node execution

2.1.2 Challenge 2: Flexibility in the pipeline development

Another important point is the capability of the platform to manage easily changes in the pipeline with minimal manual intervention. In particular, the prototype platform will put the basis for future enhancement by allowing easy link between the research activity and the production activity.

The overall process starts from the research activity, using Conda environment (<u>https://conda.io/</u>). This activity provides as output the code (and its updates) on the internal GitLab. Then, each processor generates a processor as described in SRD that is managed by the orchestrator to run a pipeline.

The challenge is to design and implement a pipeline under known conditions linking processing nodes. The verification is done by:

- Verifying the capability to design a pipeline of nodes
- Verifying the capability to run the pipeline and obtain the final results

2.1.3 Challenge 3: Delivery of results

In addition, a general OGC server is added to the processing platform that is used for the access to the large amount of data. The server allows to search for data collections using OGC CSW interface and to visualize/download data (e.g. HRLC products) using OGC WMS/WCS services.

The challenge is to implement a software component able to present different type of geospatial data and in particular of the type raster and vector, the verification is done by:

- Verifying the capability to show Metadata type of data (Vector encoding) as OGC Service and through a Web Interface
- Verifying the capability to show raster data of different type, specifically Sentinel 2 data (True Color Image) and a final product classification

2.2 Verification environment

The environment is fully configured on a Cloud and is designed to exploit the capability to be elastic of a cloud. In the following table we give some technical details of the software and configuration info where applicable.

Ref	CCI_HRLC	_Ph1-SVR	migh resolution
Issue	Date	Page	land cover
1.rev.0	17/04/2020	7	cci

The configuration is made in region eu-central-1 (frankfurt) as Sentinel data are only hosted on this region. The test environment is equivalent to the final environment for what concerns the Orchestrator scope as it can scale using all resources of the cloud. For the delivery system it is downgraded (to avoid wasting of resources) even if is made of on-demand resources that can be scaled anytime. The reference for the instances is given at https://www.amazon.com/ec2/instance-types/?nc1=h_ls.

Component	Software	Configuration	
Orchestrator	Max-ICS, <u>http://www-max-ics.earthlab.lu/</u>	Orchestrator: 3 t3.2xlarge (8 vCPU 32. GB) instances on AWS Worker: r4.2xlarge	
laaS	AWS S3, https://aws.amazon.com/s3/?nc1=h_ls	Bucket: S3://ccihrlc	
1003	AWS EC2, https://aws.amazon.com/ec2/?nc1=h_ls	Region: eu-central-1	
Code Repository	GITLAB, <u>https://about.gitlab.com/</u>	Not Applicable	
OGC Server	Gesoerver, <u>http://geoserver.org/</u>	Geoserver and Postgres Database: t3a.2xlarge 8 vCPU 32.0 GB	

Table 1: Environment configuration

3 System Verification

The objective of this paragraph is to describe a known scenario in terms of processors (with well-known and tested input and output) and to use the scenario to design test cases for the system according to the verification scenario.

3.1 Design of the verification scenarios

In version 1 of this SVR we present the verification test as applicable to the system described in the SSD. Detailed verification tests and results including the processor integration will be included in the following versions of the document.

The scenario for the verification of the system is described in the following points:

- Creation of the pipeline and execution of a processing of Sentinel 1 SLC data to generate Amplitude products.
- The process is discovered using OpenEO Discovery API
- The execution is triggered using OpenEO Execution API.
- Products to be delivered demonstrating the capability to deliver different products both raster and vector:
 - Sentinel 2 catalogue
 - Sentinel 2 L2A True Color Image
 - o Raster Classification

Cesa	Ref	CCI_HRLC	_Ph1-SVR	migh resolution
	Issue	Date	Page	land cover
	1.rev.0	17/04/2020	8	cci

3.2 Design of the verification tests

Scenarios for the verification are taken from the SSD and in particular from the Enterprise Viewpoint as the interest is to demonstrate the capability of the system to meet user needs and the technical challenges. The following user are then introduced:

- 1. **User of final products role** it is the user that access the delivery services to visualise the products and access to them with available interfaces, the use cases are
- 2. User of processing service role it is the user that access the available interfaces (API/CLI) to execute a processing service, the use case is
- 3. **Expert user role** it is the user that access to the available interfaces (API/UI) to deploy a new processing service and create a pipeline using those available

The following table links the challenges with the users and define the verification test. The numbering of the tests is explained as follow:

- The first number TEST-1-1 identifies the challenge number as defined in paragraph 2.1 so TEST-1-1 is related to the Challenge 1
- The second number TEST-1-1 identifies the incremental number of test on the challenge so TEST-1-1 is the first test of the challenge

Code	User	Test Name	Verification Objective	
TEST-1-1	User of processing service	OpenEO Process Discovery API	Verify the availability of OpenEO API for Process Discovery	
TEST-1-2	User of processing service	OpenEO Process Execution API	Verify the availability of OpenEO API for Process Execution	
TEST-1-3	User of processing service	Activation of multiple resources	Verify the activation of multiple resources as requested by the designed test on a single node execution	
TEST-1-4	User of processing service	Pipeline execution monitoring	Verify the monitoring of execution of a pipeline from the interface	
TEST-2-1	Expert user role	Pipeline Design	Verifying the capability to design a pipeline of nodes	
TEST-2-2	Expert user role	Pipeline Run	Verifying the capability to run the pipeline and obtain the final results	
TEST-3-1	User of final product	Metadata Display	Verifying the capability to show Metadata type of data (Vector encoding) as OGC Service and through a Web Interface	
TEST-3-2	User of final product	Products Display	Verifying the capability to show raster data of different type, specifically Sentinel 2 data (True Color Image) and a final product classification	

Table 2: Organisation of test with respect to users and challenges

	Ref	CCI_HRLC	migh resolution	
esa	Issue	Date	Page	land cover
	1.rev.0	17/04/2020	9	cci

3.3 Verification tests

3.3.1 TEST-1.1: OpenEO Process Discovery API

TEST-1	.1: OpenEO Process Discovery API
Initial o	conditions:
	The OpenEO API is up and running A Process is registered in the system
Test ex	ecution procedure:
1.	The operator send the OpenEO request according to the specification defined for discovery of jobs using GET with no: Error! Hyperlink reference not valid. where the url is masked internally
2.	The server responds with a JSON document
Test re	sult:
The red	quests is executed correctly as shown in the following JSON Snippet
{	
	"processes": [{
	"id": "apply",
	"summary": "Perform SAR geocoding on images",
	"description": "",
	"categories": [
	"Sentinel 1"
],
	"parameters": [{
	"name": "data",
	"description": "image",
	"schema": {
	"type": "object",
	"subtype": "image"
	}
	},

3.3.2 TEST-1.2: OpenEO Process Creation API

TEST-1.1: OpenEO Process Creation API

Initial conditions:

- 1. The OpenEO API is up and running
- 2. A Process is registered in the system

```
Ref
                                               CCI_HRLC_Ph1-SVR
                                                                                    high resolution
  esa
                                                                                    land cover
                         Issue
                                            Date
                                                               Page
                                                                                    cci
                        1.rev.0
                                         17/04/2020
                                                                10
Test execution procedure:
    1. The operator send the OpenEO request according to the specification defined for creation
        of batch job: <u>https://localhost/api/0.4/jobs</u> Error! Hyperlink reference not valid. where the
        url is masked internally
{
        "title": "NDVI based on Sentinel 2",
        "description": "Perform SAR geocoding on images ",
        "process_graph": {
                "dc": {
                        "process_id": "load_collection",
                        "arguments": {
                                "id": "Sentinel-1",
                                "spatial_extent": {
                                        "west": 16.1,
                                        "east": 16.6,
                                        "north": 48.6,
                                        "south": 47.2
                                },
                                "temporal_extent": [
                                        "2018-01-01",
                                        "2018-02-01"
                                ]
                        }
                },
                "sargeocoding": {
                        "process_id": "sargeocoding",
                        "description": "",
                        "arguments": {
                                "data": {
                                        "from_node": "dc"
                                },
                                "polarization": [
                                        "All"
                                ]
                        }
                },
                "result": true
```

	Ref	CCI_HRLC	migh resolution	
Cesa	Issue	Date	Page	land cover
U UU	1.rev.0	17/04/2020	11	cci

}

}

- 2. The server responds with a 200 status code sending back a 201 status code with the following information:
 - Location URL of the created resource: ex. <u>https://openeo.org/api/v0.4/jobs/123</u>
 - OpenEO-Identifier: ex. 123

Test result:

The request is executed correctly. The process then can be executed sending a POST request to the created resource: <u>https://openeo.org/api/v0.4/jobs/123/results</u>

3.3.3 TEST-1.3: Activation of multiple resources

TEST-1.	3: Activation of multiple resources
Initial c	onditions:
1.	The OpenEO API is up and running
2.	AWS API is up and running
Test ex	ecution procedure:
1.	The user sends a POST request to the API to activate multiple resources
	POST • Orodbas/ST/I/4c2/request_instances Params Authorization Headers (5) Body (6) Pre-request Script Tests Settings
2.	The user checks that the response status is "200 OK" and the response body contains the IDs of activated resources
2.	activated resources
Test res The req	activated resources Body Cookles Headers (9) Test Results Status: 200 OK Time: 17 Pretty Raw Preview Visualize JSON
Test res The req	activated resources Body Cookies Headers (9) Test Results Status: 200 CK Time: 17 Prety Raw Preview Visualize JON T The Preview Visualize JON The
Test res The req	activated resources Body Cookes Headers (9) Test Results Status: 200 CK Time: 17 Pretty Raw Preview Visualize JON I IIII IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII

	Ref	CCI_HRLC	migh resolution	
esa	Issue	Date	Page	land cover
	1.rev.0	17/04/2020	12	cci

3.3.4 TEST-1.4.1: Pipeline execution monitoring using Service API

TEST-1.	TEST-1.4.1: Pipeline execution monitoring using Service API						
Initial c	onditions:						
1.	The OpenEO API is up and running						
2.	The pipeline is executing at least one job						
Test ex	ecution procedure:						
1.	The user sends a GET request to the API endpoint /jobs with the running job ID						
2.	The user checks the "status" field in the JSON response						
Test res	sult:						
The req	uests is executed correctly as shown in the following JSON						
	<pre>{ "product_instance_id": "test-product-instance-id", "node_id": "test-node-id", "service_id": "test-service-id", "items": ["test-item-1", "test-item-2"], "status": "started", "delivered_links": [] }</pre>						

3.3.5 TEST-1.4.2: Pipeline execution monitoring using Max-ICS logs

TEST-1.	TEST-1.4.2: Pipeline execution monitoring using Max-ICS logs										
Initial c	onditions:										
1.	The Service API	The Service API is up and running									
2.	The pipeline is e	The pipeline is executing at least one job									
3.	The user is logge	The user is logged in Max-ICS and he is a contributor of the pipeline									
Test exe	ecution procedure	2:									
1.	The user access	to the pip	eline panel								
	🗞 CCI Pipeline						🖺 😻	<u>۵</u>			
	General	•	Draw your pipeline	*	Documentation	1	Define internet access	×			
	Title*	CCI Pipeline									
	Description										
				le le							
	Resources allocated	0.2 core(s) CPU	256 MB 0 gpu(s) Memory GPU	1 node(s) Autoscale							
2	The second states a			•	I						
2.	The user clicks a	ccess to t	ne pipeline mon	itoring p	anei						
3.	The user clicks o	n "Logs"	tab								
4.	The user waits that all the nodes of the pipeline log their behavior for the running job ID										

	Ref	CCI_HRLC	migh resolution	
esa	Issue	Date	Page	land cover
	1.rev.0	17/04/2020	13	cci

t result:													
request is exe	ecuted correctly	as shown in	n the f	followi	ing scr	eensh	ot:						
	≡ ← ☎ 🗌 / My p	pelines / Monitoring	/ Logs							ß	#	ወ	Ş
	🚳 CCI Pipeline												
Dashboard	Instances monitoring Logs	Deployment errors	Tasks	Commit hist	tory Stora	ge							
our treatments	Application All applications	Word wrap OFF											
Arrow Als Arrow Ar	10410 14:55:44,94205 32286 f 10410 14:55:44,94220 20286 f 10410 14:55:44,94220 20286 f 10410 14:55:44,94274 32286 f 10410 14:55:44,94264 32286 f 10410 14:55:44,94264 32286 f 10410 14:55:44,94264 32286 f 10410 14:55:44,942683 32286 f 10410 14:55:44,942683 32286 f 10410 14:55:44,942683 32286 f 10410 14:55:44,94283 32286 f 10411 14:55:44,94353 52286 f 10411 14:55:44,94535 5286 f 10411 14:55:44,9455 5286 f 10411 14:55:44,9455 5286 f 10411 14:55:44,9455 5286 f 10411 14:55:44,9455 5286 f 10	etcher.cpp:175] Downloadin etcher.cpp:1349] Fetching U etcher.cpp:1618] Fetched 'n etcher.cpp:1623] Successful kec.cpp:126] Version: 1.8. kec.cpp:126] Executor regi kecutor.cpp:190] Received kecutor.cpp:190] Received kecutor.cpp:1710] Starting	ng resource f URI 'https://nexus Illy fetched a .1 istered on ag SUBSCRIBED e ed executor o LAUNCH event task maxics_ command at 322	<pre>from 'https:// //nexus.earthlab.lu, ss.earthlab.lu, all URIs into event on autoprov-26 tt s_cci-pipeline 2298</pre>	<pre>//nexus.earthla .ab.lu/nexus/se i/nexus/service > '/var/lib/mes >-1192-4a6e-a6a 00200409t200139 :_new-node.inst</pre>	<pre>b.lu/nexus/se rvice/local/r /local/reposi os/slaves/261 1-334066afed8 -9.prov.earth</pre>	ervice/local repositories itories/max- 18ad1b-1192- 80-515 hlab.lu	/repositories :/max-ics-rele ics-release/c 4a6e-a6a1-334	/max-ics-rele ase/content/c ontent/cci-pi 066afed80-515	ase/conte ci-pipeli peline/ne	nt/cci-pi ne/new-no w-node/20	peline/ne de/202004 200416T14	w-nod 16T14 5523Z

3.3.6 TEST-2.1: Pipeline Design

TEST-2	2.1: Pipeline Design	
Initial	conditions:	
1. 2.	User logged in to Max-ICS User created an empty workspace to design the pipeline on Max-ICS UI	
Test ex	xecution procedure:	
1.	The user creates a node of type API by clicking "Add node"	
	CCI Test pipeline General Image: Comparison of the second secon	
	Complete view Add node Add flow	
2.	The user selects API as the type of the node	

	Ref	CCI_HR	_C_Ph1-SVR	niah r	esolutio		
esa	Issue Date		Page	and c	high resolution high resolution		
1.rev.0 17/04		17/04/2020	14	CCI			
Add new applica	ation				×		
	Туре			Options			
1. Select type	and sub type						
Type*	1	!≡!	8	9			
	Poller	Connector	Classical treatment	Deep learning treatment			
	UI	API	Database				
	01		Database				
Source*	git	4					
	Continuous deployment	Docker	Github				
Template	Repository will be created fror	n scratch Q					
					→		
3. The user s	ets the node parar	neters (CPU, Mer	norv. Autoscalin	g)			
			,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	67			

	Ref	CCI_HRLC	migh resolution	
esa	Issue	Date	Page	land cover
	1.rev.0	17/04/2020	15	cci

Node UID	Will be set on save			4
Туре	API			
Title*	New node	*		
Autoscaling	ON	Minimum* 1	Maximum* 년	•
Number of instance(s)*	1 - 1	1		
GPU Support	OFF			
Security Class*	Public Cloud	Earthlab Cloud	M On-Premises	
Public provider*	a Amazon Web Services	۱ Mundi	Google Cloud Platform	
Tenant*	Earthlab Customer			
CPU cores*	0,1 -	2		
Memory*	128Mo 128 256 512 1024 204	8 4096 6144 8192	10240 12288 14336 16384	34816Mp 18432 24576 32768 34816
	122 122 122 122	o wudo olee ditiz	102907 12200 14330 16384	ditere dottae ofters server

	Ref CCI_HRLC_Ph1-SVR		igh re	Figh resolution	
esa	Issue	Date	Page	land co	
	1.rev.0	17/04/2020	16		
Add new applicati	op				×
Add new applicati	011				
	Туре			Options	
1. Select type a	and sub type				
1. Select type a	ind sub type				
Type*	.	! ⊜]	F	9	
	Poller	Connector	Classical treatment	Deep learning treatment	
	UI		Database		
Source*	git	٠			
	Continuous deployment	Docker	Github		
Template	Repository will be created from	scratch Q			
					>
6. The user s	sets the node para	matars (CPUL Mar	nory Autoscaling)		
0. The user .			nory, Autoscaning,		
7. The user s	saves the node by o	clicking the save b	utton		
	waataa a "flaw" ah	is at by aliaking "A	dd flaw"		
8. The user of	creates a "flow" ob	Ject by clicking A	ad now		
🗞 CC	I Test pipeline				
	General	4	Draw your pipe	line 🖌	
×	Complete	view 🔼			
	d node 🛛 🧭 Add flow				
9. The user l	inks the nodes tog	ether by using a "	flow" object		

IssueDatePage1.rev.017/04/202017	Cesa	Ref	migh resolution		
1.rev.0 17/04/2020 17		Issue	Date	Page	
		1.rev.0	17/04/2020	17	cci

	🗞 CCI Pipeline	
	General	~
	🗙 💠 Comple	te view
	Add node 🛛 🤌 Add flow	🖉 Edit node 🛞 Delete selected
	APi Queue	
	New node N	ew node - 2
Test ı	esult:	
1	0 1 1 1	
2	0 1 1 1	ted for "New hode -2" Service Registry alongside with its output queue
4	-	a Service Registry alongside with its input queue
6	., .	
AF	New node	
	Node UID	new-node
	Service port	38426
	Public access	Not activated yet
	Internal access Git https access	http://rp.service.earthlab.lu:38426 https://max-ics.earthlab.lu/gitlab/cci-pipeline/new-node.git
	Debugging marathon identifier	maxics_cci-pipeline_new-node.710a014b-8fc3-46ff-8b67-3ae46656afc2

	Ref	CCI_HRLC_Ph1-SVR			Figh high resolution	
esa	Issue	D	ate	Page	high resolution	
	1.rev.0	17/0	4/2020	18	cci	
🔊 New node -	- 2					
		Node UID	new-node2			
		Service port	36168			
	Pu	Iblic access	Not activated ye	t		
	Internal access			earthlab.lu:36168		
	Git h	ttps access	https://max-ics.e	arthlab.lu/gitlab/cci-pipeline/r	new-node2.git	
	Debugging marathe	on identifier	maxics_cci-pipe	line_new-node2.549757f3-	-1615-424f-ac0d-6cbec2976c63	

3.3.7 TEST-2.2: Pipeline Run

TEST-2	2.2: Pipeline Run					
Initial o	conditions:					
1. 2. 3. 4.	 The user has pushed the code to each node of the pipeline to handle requests The nodes code includes logs of the process status to the stdout 					
Test ex	kecution procedure:					
2. 3.	The user sends a request to the pipeline API node with one of the following u Curl Postman Python requests Swagger Test UI	ıtility:				
Servers https://co	ci-pipeline.max-ics.lu/pipeline 🗸					
defau		~				
Post the	/invoke_pipeline Request to invoke the pipeline e JSON of the request from the service platform					
Parame	eters	Try it out				
No para						
A json o	st body regined appli	cation/json V				

	Ref CCI_HRLC_Ph1-SVR			migh resolution
esa	Issue	Date Page		and cover
	1.rev.0	17/04/2020	19	cci

2. The	user acces	s the Ma	x-ICS logs par	nel			
🗞 CCI P	ipeline						
nstances m	onitoring	Logs	Deployment	errors	Tasks	Commit history	Storage
Application	New node	•	Word wrap	OFF			
I0416 15:1 I0416 15:1	4:27.758380 4:27.760663 4:27.761962 4:27.762163 4:27.762302 4:27.763018 4:27.763069 4:27.763087 4:27.763099 4:27.763116 4:27.763150 4:27.765398	1279 exe 1282 exe 1284 exe	c.cpp:162] Vers c.cpp:236] Exec cutor.cpp:190] cutor.cpp:194] cutor.cpp:1968] cutor.cpp:668] cutor.cpp:668] cutor.cpp:684] cutor.cpp:684] cutor.cpp:710] cutor.cpp:726]	sion: 1.8. cutor regi Received Subscribe Received Overwriti Overwriti Overwriti Overwriti Starting Forked co	1 stered on SUBSCRIBED d executor LAUNCH eve ing environ ing environ ing environ ing environ task maxic mmmand at 1	on autoprov-2020040 nt ment variable 'LANG' ment variable 'LC_AL ment variable 'PATH' ment variable 'LC_AL ment variable 'LANG' s_cci-pipeline_new-n	4a6e-a6a1-334066afi 7t220740-4.prov.ea L' L' ode.instance-e727bl
est result: he logs sho	ow that the	request	has been har	ndled			
nts":["ln", hts":["ln", sioner/cond New node] : 5:22:11] "GI 5:22:11] "GI 5:22:11] "GI 5:22:13] "GI 5:22:13] "GI 5:22:13] "GI New node] : 5:22:29] "GI 5:22:29] "GI	"-s","/sys/f "-s","/sys/f tainers/7fcc :ffff:10.99. ET /pipeline ET /pipeline ET /pipeline ET /pipeline ET /pipeline :ffff:10.99. ET /pipeline ET /pipeline	s/cgroup/ s/cgroup/ bf7a-7748 60.14 /ui/ HTTP /ui/swagg /ui/swagg /ui/swagg /openapi. /ui/favic 60.14 /ui/swagg /ui/swagg	cpu,cpuacct"," -4c1d-8650-471 (2020-04-16 1 /1.1" 200 1557 er-ui.css HTTP er-ui-standalo er-ui-bundle.j json HTTP/1.1" on-32x32.png H (2020-04-16 1 er-ui-standalo er-ui.css HTTP	/var/lib/ /var/lib/ 45c5f0f7b L5:22:11] 0.004643 /1.1" 200 ne-preset s HTTP/1. 200 852 TTP/1.1" L5:22:28] ne-preset /1.1" 200	mesos/prov /backends/ "GET /pipe 142144 0. .js HTTP/1 1" 200 974 0.000539 200 913 0. "GET /pipe .js HTTP/1 142144 0.	.1" 200 307340 0.110 342 0.127025 000743 eline/ui/ HTTP/1.1" 3 .1" 200 307340 0.076 076529	7fccbf7a-7748-4c1d 312527-5704-4bb5-a 308 481 0.001598 3798 200 1557 0.000518
5:22:29] "GI 5:22:29] "GI	ET /pipeline ET /pipeline	/ui/swagg /openapi.	er-ui-bundle.j json HTTP/1.1" on-32x32.png H	s HTTP/1. 200 852	1" 200 974 0.000502	342 0.157146	

3.3.8 TEST-3-1 Metadata Display

TEST-3.1: Metadata Display

Initial conditions:

1. The user has entered in the Web Interface with a user/password

Test execution procedure:

1. The user selects the map or create a new map

	Ref	CCI_HRLC	migh resolution	
esa	Issue	Date	Page	land cover
	1.rev.0	17/04/2020	20	cci

- 2. The user uses the top right tool to add a WMS Land Cover layer from the catalogue by selecting the WMS "Sentinel 2 DataCube"
- 3. The user load on the map a WMS of a sentinel scene (True Color Image) of choice and a second image for the same Tile
- 4. The user can visualize the images on the map

Test execution procedure:

1. The user uses the Catalog function to query the Sentinel 2 catalogue

×	-	and the second	A 8 3	BERRY	🗁 Catalogo			\rightarrow
IN COLHRLC	A.	100	1-1-20	Couk	Servizio			
Filtra layers	· 500	Randa	M. C. R.	CHINE I	Sentinel 2 Data Cube		• • •	+
	12	AN PART		5-5-5	testo da cercare.			
💋 Default	- 12	Mit is	ATT.	Rato	Cerca			
1C changes 3-2	. 79	150		ME	1	S2L2A_29TOH2020010200R10mTCI		i
	- 50	A SAMA	3 La		Preview Not Avaliable	S2_29TGH52L2A_29TGH2020010200R10mTCl S2L2A_29TGH2020010200R10mTCl		
🛙 💋 LC changes 2-1	< the		Se m	& +		+ Aggiungi alla mappa		
💋 Coffee Plantations M.	< 10E	RX		6-	1000000	S2L2A_29TQH2020010500R10mTCl		
	. Chr	C M	1 Martin	$\mathcal{C} \times \mathbb{R}$	Preview Not	St.2gTQH:StL2A,2gTQH2020010500R10mTCI		
Land Cover epoch 1		22 mil	Hereby		Avaliable	S2L2A_2gTOH2020010500R10mTCl + Agglungi alla mappa		
		12-19	ES 1	3D		S2L2A_29TOH2020010700R10mTCl		Ξ.
	6	C. May	2473			1 2 3 4 5 . 107 · -		
		ANS PES	- 18 A	100		Risultati 1-4 di 428		

2. The user can select multiple dates from a Sentinel 2 tile and add them to the map to check for original data. In this case, in the following screenshot, an example of True Color images is shown



	Ref	CCI_HRLC	migh resolution	
C esa	Issue	Date	Page	land cover
	1.rev.0	17/04/2020	21	cci



3.3.9 TEST-3-2 Products Display

TEST-3.2: Product Display

Initial conditions:

2. The user has entered in the Web Interface with a user/password

Test execution procedure:

- 1. The user selects the map or create a new map
- 2. The user uses the top right tool to add a WMS Land Cover layer from the catalogue by selecting the WMS "CCI HRLC Test"
- 3. The user load on the map a WMS Land Cover layer
- 4. The user load on the map a WMS Change Map layer

Test result:

1. The following screenshot show an example of land-cover classification using LCCS based legend

	Ref	CCI_HRLC	migh resolution	
Cesa	Issue	Date	Page	land cover
	1.rev.0	17/04/2020	22	cci



2. The following screenshot show an example of transition map used to visualise different transition of interest

