

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

DHuS Administration Manual



Role/Title	Name	Signature	Date
Authors	Copernicus IVV Team		05/03/2019
Verified/Approved	Copernicus Maintenance Manager		05/03/2019

Change register

Version /Rev.	Date	Description
1.0	05-02-2015	First issue of the document
1.1	17-03-2016	<p>The following deletion has been made to be in line with the Software distribution of versions after 0.9.0-2:</p> <ul style="list-style-type: none"> • GWT GUI deletion • Statistic panel description deletion <p>User manual remanded to the online versions</p>
1.2	27-07-2016	<p>The following information has been added in the document to be in line with the Software distribution of versions after 0.10.3-4:</p> <ul style="list-style-type: none"> • configuration parameters description • removed example of software update from 0.4.3-1 version • synchronization best practice in case of proxy rules added in the synch without copy instructions. • synch with copy description and instructions • timeout parameter instruction • the data hub software functions description and image changed • add missing info on the synchronization instructions • new dhus.xml updated
1.3	09/02/2017	Review and Update of the whole document
1.4	21/03/2017	Shapefile configuration description
1.5	13/09/2017	Update according to DHuS 0.13.4-13 Software Release Note
1.6	09/10/2017	Update according to DHuS 0.14.1-3 Software Release Notes: introduction of the scalability 2.0 deploy mode and deletion of the scalability 1.0
1.6	11/10/2016	<p>The following information has been added in the document:</p> <ul style="list-style-type: none"> • Description to configure the sorting criteria for datatake sensing start and tile id in AJS GUI; • Description to disable the new map toolbar in AJS GUI.
1.7	12/03/2018	<p>The following information has been added in the document in Appendix 7:</p> <ul style="list-style-type: none"> • Map polygon selection/pan management; • Map layers attributions configurability; • Configuration of more than three map layers;

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

		Map layer selector restyling.
1.8	11/04/2018	Update according to DHuS 0.13.4-21-1 Software Release Note
1.9	26/04/2018	Overall document structure has been updated. Content updated according to DHuS 0.14.1-10 Software Release Notes.
2.0	10/08/2018	Content updated according to DHuS 0.14.4-3 Software Release Notes. The following information has been added and updated in the document: <ul style="list-style-type: none">• Eviction section• DataStore section• Sentinel-1 PDGS LTA component interface• dhus.xml, start.sh, temporary_files_ehcache, AJS UI appconfig.json and styles.json configuration files• Synchronizer section• Connection information section
2.1	17/09/2018	Content updated according to DHuS 0.14.4-4 Software Release Notes. <ul style="list-style-type: none">• Linked eviction for GMP DataStore section added• Linked eviction section updated• DataStore Automatic Size Management section updated• Customizable Eviction Actions updated• GMPDataStore section updated• Eviction Management section updated
2.2	20/11/2018	Content updated according to DHuS 0.14.7-2 Software Release Notes. The following information has been added and updated in the document: <ul style="list-style-type: none">• Customizable Eviction section• Sentinel-2 Archive Interface and related management sections• Storage configuration section• SAFE Eviction management section• dhus.xml and AJS UI appconfig.json files
2.3	05/02/2019	Content updated according to DHuS 0.14.8-5 Software Release Notes. The following information has been added and updated in the document: <ul style="list-style-type: none">• User Management, the management of users via API has been added• Approach related to netCDF cache location configuration• Cancel Eviction action via API has been added• DataStore CRUD operations updated

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

		<ul style="list-style-type: none">Enlarging thumbnail function added int the appconfig.json parameters
2.3.1	05/03/2019	New release naming convention used for DHuS 0.14.8-5, now named DHuS 2.0.0.

Table of Contents

1.	Introduction	10
1.1	Scope.....	10
1.2	Purpose.....	10
1.3	Document applicability	10
1.4	Reference documents	11
1.5	Acronyms	12
1.6	Definitions and Glossary	12
2.	System overview	12
2.1	DHuS Funcionalities	13
2.2	DHuS Interfaces	14
2.2.1	Web-based Graphical User Interface (AJS).....	14
2.2.2	Open Web Component (OWC)	14
2.2.3	Open Data Protocol (OData)	14
2.2.4	OpenSearch	15
2.3	DHuS system databases.....	15
3.	DHuS deployment modes.....	15
3.1	Single Instance mode.....	15
3.2	Front-End/Back-End Mode	16
3.3	Scalability 2.0 Mode	16
4.	Installation pre-requisites.....	18
4.1	Infrastructure Requirements.....	18
4.1.1	Single instance and FE/BE Mode	18
4.1.2	Scalability 2.0	18
4.2	Network Requirements	18
4.3	Software Requirements	19
5.	Administration Functionalities	19
5.1	Login	19
5.2	Product Ingestion	19
5.2.1	Ad hoc product upload	19
5.2.2	Upload via File Scanner	20
5.3	Product deletion	20
5.4	Synchronizer	21
5.5	User Synchronizer.....	25
5.6	User management	25
5.7	Collection management.....	26
5.8	System management	26
5.9	Product Eviction.....	26
5.9.1	Customizable Evictions	26
5.9.2	DataStore Automatic Size Management	28
5.10	Auxiliary Data Files	29
5.11	Sentinel-1 PDGS LTA component Interface.....	29
5.11.1	Overview	29
5.11.2	Quotas	30
5.11.3	GMP	31
5.11.4	GMPDataStore	31
5.12	Sentinel-2 Archive Interface	32
5.12.1	Overview	32
5.12.2	RemoteDHuSDataStore	32
5.12.3	SAFE-Eviction.....	33
5.12.4	Pre-requisites.....	33
6.	Installation and Configuration Procedures	33
6.1	Software configuration	33

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

6.1.1	Storage configuration	34
6.1.2	DataStore properties and configuration	35
6.1.3	NetCDF Temporary Cache mechanism	37
6.1.4	OWC configuration	38
6.2	Single Instance installation (HFS).....	39
6.3	Scalability 2.0 Mode installation	40
7.	Administrator Procedures	40
7.1	Login	40
7.1.1	How to login as administrator	40
7.2	Product Ingestion	42
7.2.1	How to upload ad hoc product	42
7.2.2	How to upload via File Scanner	44
7.3	Product deletion	47
7.3.1	How to delete a product	47
7.4	Synchronizer	49
7.4.1	How to create a new Synchronizer	49
7.4.2	How to update a Synchronizer	52
7.4.3	How to delete a Synchronizer	53
7.5	User Synchronizer.....	53
7.5.1	How to create a User Synchronizer.....	53
7.5.2	How to start/stop a User Synchronizer.....	54
7.5.3	How to update a User Synchronizer.....	54
7.5.4	How to delete a User Synchronizer.....	54
7.6	Management Panels.....	54
7.7	User Management.....	56
7.7.1	How to manage users via GUI.....	56
7.7.2	How to manage users via API	60
7.8	Collection Management	62
7.8.1	How to create a new collection	62
7.8.2	How to delete a collection.....	63
7.9	System Management.....	63
7.9.1	How to set mail configuration	65
7.9.2	How to set Support information	66
7.9.3	How to change root password.....	66
7.9.4	How to restore database	67
7.10	Eviction Management.....	68
7.10.1	How to manage a Customizable Eviction via AJS GUI	68
7.10.2	How to manage a Customizable Eviction via OData API	70
7.10.3	How to manage a SAFE Eviction via OData API	72
7.10.4	How to manage an Automatic On-Insert Eviction	73
7.11	Main Menu Settings	74
7.12	Auxiliary Data Files	75
7.12.1	How to configure ADF retrieval chain.....	75
7.13	User details information	76
7.14	Connections information	76
7.14.1	How to identify Active users.....	76
7.15	Network information	77
7.16	DataStore Management	77
7.16.1	JSON ODATA representation for HFSDATAStore	77
7.16.2	JSON ODATA representation for Openstack DataStore.....	77
7.16.3	DataStore CRUD operations	78
7.17	OpenStack storage command line samples.....	79
7.18	NetCDF Temporary Cache setting	80
7.18.1	How to disable the Cache	80
7.18.2	How to configure the Cache.....	80

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

7.19	Sentinel-1 PDGS LTA component Interface setting.....	83
7.19.1	JSON ODATA representation for GMPDataStore	83
7.19.2	How to manage GMPDataStore	84
7.20	Sentinel-2 Archive Interface setting	84
7.20.1	JSON ODATA representation for RemoteDHus DataStore	84
7.20.2	How to manage RemoteDHuSDDataStore	85
8.	DHuS configuration file parameters	85
8.1	dhus.xml	85
8.1.1	crons.....	86
8.1.2	messaging	87
8.1.3	network.....	88
8.1.4	products	89
8.1.5	search	90
8.1.6	server.....	91
8.1.7	system	91
8.1.8	dataStores	93
8.1.9	scanners.....	96
8.1.10	synchronizers.....	96
8.1.11	evictions	97
8.2	server.xml	98
8.3	start.sh	100
8.4	log4j2.xml	104
8.5	suggerster.dic	107
8.6	opensearch-description-file.xml.....	107
8.7	temporary_files_ehcahe.xml	107
9.	AJS GUI configuration parameters	109
9.1	appconfig.json	110
9.1.1	Login badge.....	110
9.1.2	Missions	110
9.1.3	First node inspection	113
9.1.4	Shapefile	113
9.1.5	Terms and conditions link.....	114
9.1.6	List pagination	114
9.1.7	Deletion cause	115
9.1.8	Map polygon selection/pan management.....	115
9.1.9	Map layers attributions configurability	116
9.1.10	Configuration of more than three map layers	117
9.1.11	Map layer selector restyling	117
9.1.12	Sorting criteria for datatake sensing start and tile id	118
9.1.13	LTA management.....	118
9.1.14	Enlarging thumbnail	119
9.2	styles.json	119
10.	OWC configuration parameters.....	121
10.1	appconfig.json.....	122
10.1.1	Flex layout templates	123
10.1.2	WMS - Map servers configuration in the Client	124
10.1.3	Metadata plugin definition	124
10.1.4	ADF feature	125
10.2	styles.json.....	126
10.3	menu.json.....	126
10.4	theme.json.....	126
11.	Configuration converter script	127
Appendix A.	Example of dhus.xml file coming from distribution	128

Table of Figures

Figure 1 DHuS High Level Architecture	13
Figure 2 DHuS functionalities	13
Figure 3 DHuS Scalability 2.0 deploy	17
Figure 4 Metadata Synchronization.....	22
Figure 5 Products Synchronization.....	22
Figure 6 DHuS-LTA interfaces	30
Figure 7 DHuS Login (AJS).....	41
Figure 8 DHuS Login Panel (AJS).....	41
Figure 9 User Information display (AJS).....	41
Figure 10 Login panel (OWC)	42
Figure 11 Products Upload (AJS)	42
Figure 12 Upload Panel (AJS).....	43
Figure 13 Upload Panel (OWC).....	43
Figure 14 File Scanner Panel (GUI).....	45
Figure 15 File Scanner Panel (OWC).....	46
Figure 16 Deletion via OData API	47
Figure 17 Deletion cause panel	48
Figure 18 OData Synchronizer Access.....	50
Figure 19 OData Synchronizer Panel (AJS).....	50
Figure 20 OData Synchronizer Panel (OWC).....	50
Figure 21 Create Synchronizer Panel (AJS)	51
Figure 22 Create Synchronizer Panel (OWC)	51
Figure 23 Updating a synchronizer	52
Figure 24 Management area (AJS)	55
Figure 25 Management Subpanel (AJS)	55
Figure 26 Management Subpanel (OWC)	56
Figure 27 DHuS User Management Panel (AJS)	57
Figure 28 User Creation Form (AJS)	57
Figure 29 User Creation form (OWC)	58
Figure 30 Update, delete users, modify user password on AJS	59
Figure 31 Delete users (OWC).....	59
Figure 32 Update, edit user password and delete users AJS	60
Figure 33 Collection Management Panel (AJS).....	62
Figure 34 System management sections (AJS)	64
Figure 35 System management sections (OWC)	65
Figure 36 Mail configuration management Panel (AJS)	65
Figure 37 Mail configuration management Panel (OWC)	66
Figure 38 Support configuration management panel (AJS)	66
Figure 39 Root configuration management panel (AJS).....	67
Figure 40 Root configuration management panel (OWC)	67
Figure 41 Restore Database panel (AJS)	67
Figure 42 Restore Database panel (OWC)	68
Figure 43 Eviction Management panel (GUI)	69
Figure 44 Eviction update via GUI	70
Figure 45 Eviction delete via GUI	70
Figure 46 Main Menu Settings panel (OWC)	75
Figure 47 Suggester configuration.....	107
Figure 48 Shapefile description display	114
Figure 49 Map layer attribution display	117
Figure 50 Map layer panel	118
Figure 51 Instrument label as defined in the value field of the styles.json file.....	121

Table Index

Table 1 DHuS version & Documents reference Mapping Matrix.....	11
Table 2 Reference Documents	12
Table 3 Singe Instance & FE/BE Mode Infrastructure Requirements	18
Table 4 Network Requirements	18
Table 5 Synchronizer type vs Remote incoming configuration	23
Table 6 Storage configuration on dhus.xml	35
Table 7 Common DataStore entities properties	36
Table 8 HFSDataStore entity properties	36
Table 9 OpenStackDataStore entity properties	36
Table 10 GMPDataStore entity properties	37
Table 11 Remote DHuSDDataStore entity properties.....	37

1. Introduction

1.1 Scope

This document applies to the Data Hub Software ("DHuS") and is maintained within "Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions" contract (n.: 4000116830/16/I-BG) by Serco.

1.2 Purpose

The purpose of this document is to describe all essential information to make full use of the DHuS software. The target audience of this document therefore are System administrators that will install the DHuS software and manage the DHuS.

This manual includes a description of the DHuS system functions and alternate modes of operation, and step-by-step procedures for system access and use.

In particular, it describes (and dedicates a specific section to):

- How to install the DHuS software;
- How to administer, manage and operate the installed DHuS instance.

The User-oriented manual of the Data Hub is available here <https://scihub.copernicus.eu/userguide/>.

1.3 Document applicability

Here below we provide a cross check between the SW version and the version of this document. In addition, the new features and performance improvements of each release can be found in the corresponding Software Release Note.

DHuS version	DHuS Administration Manual version	SRN reference
0.12.5-6	v1.2	Release Notes v.0.12.5-6-osf
0.12.5-11 to 0.12.5-13	v1.4	COPE-SERCO-TN-17-0079
0.13.2 to 0.13.4-13	v1.5	COPE-SERCO-TN-17-0106
0.14.0 to 0.14.1-3	v1.6	COPE-SERCO-TN-17-0150
0.14.1-7	v1.7	COPE-SERCO-TN-18-0249
0.13.4-21-1	v1.8	COPE-SERCO-TN-18-0269
0.14.1-10	v1.9	COPE-SERCO-TN-18-0274
0.14.4-4	v2.1	COPE-SERCO-TN-18-0322
0.14.7-2	v2.2	COPE-SERCO-TN-18-0357

0.14.8-5	v2.3	COPE-SERCO-TN-19-0391
2.0.0 (old release naming convention 0.14.8-5)	v2.3.1	COPE-SERCO-TN-19-0401

Table 1 DHuS version & Documents reference Mapping Matrix

1.4 Reference documents

Reference	Document name, Reference, issue number, revision number
RD 1	GitHub open source framework https://github.com/SentinelDataHub/DataHubSystem
RD 2	OData System Design Document, GAEI-P286-SDD-003-01-00, v1.2, 2016-04-25
RD 3	Open Search Description Technical Note, GAEI-P286-TCN-013, v1.0, 2017-06-12
RD 4	OWC System Design Document, SPA-COPE-DHUS-ADD-001, v1.0
RD 5	DHuS Architectural Design Document, GAEI-P286-ADD-001, v2.1 2014-06-03
RD 6	DHuS OData and Open Search Interface Control Document, COPE-SERCO-IF-17-0018, v2.0
RD 7	Cache management GAEI-P286-TCN-014, v1.0
RD 8	OWC Developer guide: http://sentineldatahub.github.io/DataHubSystem/owc.html
RD 9	OWC User Manual: http://sentineldatahub.github.io/DataHubSystem/OWC/user_manual.html
RD 10	Solr documentation: http://lucene.apache.org/solr/
RD 11	COPE-SERCO-TN-17-0140 DHuS external DB Installation and Configuration Manual v1.2
RD 12	COPE-SERCO-TN-17-0143, GMP Installation and Configuration Manual, v1.0
RD 13	COPE-SERCO-TN-17-0161, GMP System Design Document, v1.0

RD 14	Quartz Enterprise Job Scheduler: http://www.quartz-scheduler.org/documentation/quartz-1.x/tutorials/TutorialLesson06
RD 15	DHuS User Guide: https://scihub.copernicus.eu/userguide/7ODataAPI

Table 2 Reference Documents

1.5 Acronyms

Acronym	Description
API	Application Programming Interface
BE	Back-End
DHuS	Data Hub Service
FE	Front-End
GUI	Graphical User Interface
OWC	Open Web Component
SRN	Software Release Note

1.6 Definitions and Glossary

Acronym	Description
API	Application program interface, is a set of routines, protocols, and tools for building software applications.
Back-End	DHuS instance that has products in its storage (i.e. that has ingested or remote-synchronized products)
Checksum	A checksum is a digit representing the sum of the correct digits in a piece of stored or transmitted digital data, against which later comparisons can be made to detect errors in the data.
DHuS Node	A single DHuS instance, which may be networked with one or more other DHuS instances. Any DHuS node is potentially a source of data products, accessible from any other node in the network.
Front-End	DHuS instance that synchronizes product metadata from a Back-End
Procedures	A set of written instructions to be followed by an operator to perform his required tasks.

2. System overview

The Data Hub Software (DHuS) is an open source software developed by a Serco/Gael consortium to the purpose of supporting the ESA Copernicus data access.

The DHuS provides a web interface (UI) to allow interactive data discovery and download, and a powerful Application Programming Interface (API) that allows users to access the data via computer programs/scripts thereby automating/integrating the download within their workflow.

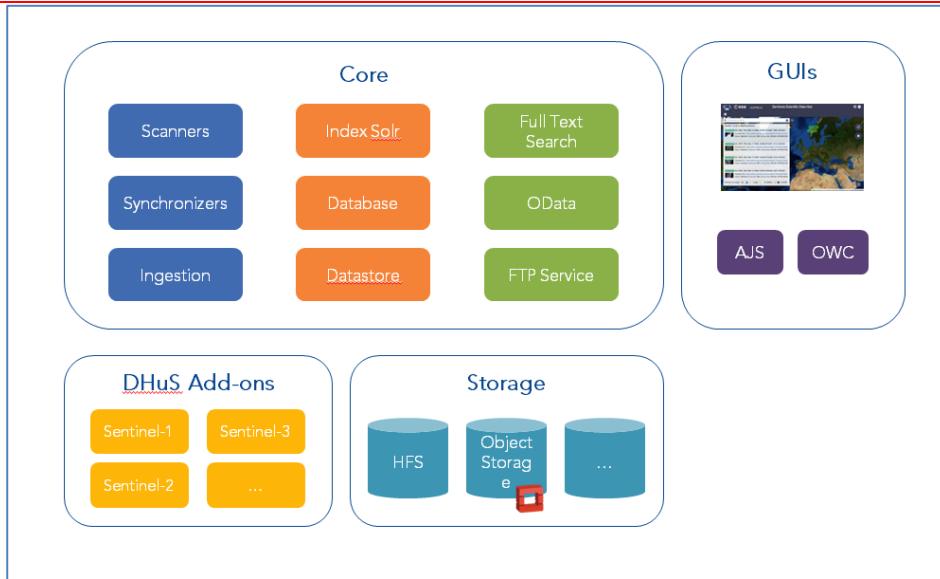


Figure 1 DHuS High Level Architecture

2.1 DHuS Funcionalities

The major functionalities of the Data Hub Software are schematically represented in figure below.

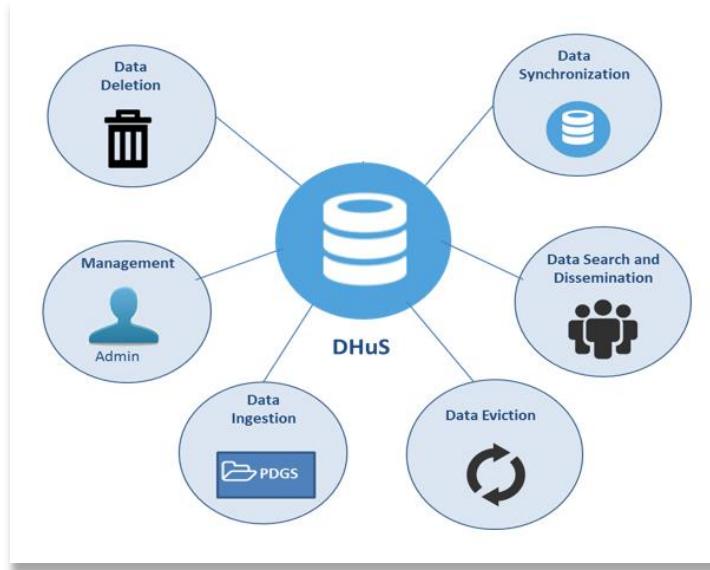


Figure 2 DHuS functionalities

Data Ingestion (DHuS→external data provider, e.g. PDGS): this function is activated by DHuS, which retrieves Sentinels' products from an external data source.

Data Search (Users->DHuS): this function is activated by external users who perform searches via four possible interfaces (described in Section 2.2):

- Web-based Graphical User Interface (**AJS**),
- Open Web Component (**OWC**) (see [RD 4]),

- Open Data Protocol (**OData**) interface (see [RD 6]),
- **OpenSearch** interface (see [RD 3]).

Data Dissemination (DHuS→Users): The dissemination function manages the product dissemination, retrieval and access by the user. It also allows disseminating/retrieving sub-components of products. This is considered of particular relevance for the voluminous data products managed in the Sentinels mission.

Data Synchronization (DHuS 1→ DHuS 2): this interaction, based on the OData protocol, allows a DHuS Instance (DHuS Front-End) to retrieve products and their metadata from another DHuS instance (DHuS Back-End).

Data deletion (Administrators->DHuS): this functionality permits to the administrators to delete products from the archive and eventually to include a record in the DHuS Database which can be read by OData protocol in order to let operators and user to identify the deleted products and the deletion cause.

Management (Administrators->DHuS and Users): this function is in charge of managing the user accounts for access to the DHuS. It enables the definition of the roles and permissions of the users and prevents uncontrolled accesses. Moreover, it controls the data eviction and the system configuration.

Data Eviction (Administrators->DHuS): The DHuS rolling archive is managed through control of the overall disk space available for EO products, the management of a general eviction policy and the creation of specific eviction rules governing the period during which products and collections remain available for dissemination prior to eviction.

2.2 DHuS Interfaces

2.2.1 Web-based Graphical User Interface (AJS)

The “default” DHuS graphical interface is the **AJS UI**: it is in charge of providing the user with an interface for the discovery, and downloading of products and for the visualization of the relevant metadata. AJS is accessible by the default at: [http://\[DHUS IP ADDRESS\]](http://[DHUS IP ADDRESS]).

2.2.2 Open Web Component (OWC)

The Open Web Component (see [RD 4]) UI is based on Google Polymer library.

This UI offers the same functionalities of the AJS one, with additional features devoted mainly to increase the application configurability.

OWC is accessible by default at [http://\[DHUS IP ADDRESS\]/new/](http://[DHUS IP ADDRESS]/new/).

2.2.3 Open Data Protocol (OData)

OData (see [RD 6]) is a standardized protocol for creating and consuming data APIs. OData builds on core protocols like HTTP and commonly accepted methodologies like REST thus facilitating access via a large set of client tools as simple as common Web browsers, download-managers or command line tools such as cURL or wget. The result is a uniform way to expose full-featured data APIs.

2.2.4 OpenSearch

OpenSearch (see [RD 3] and [RD 6]) is a collection of technologies that allow publishing of search results in a format suitable for syndication and aggregation. It is a way for websites and search engines to publish search results in a standard and accessible format.

2.3 DHuS system databases

DHuS application supports both embedded and externalized databases (since version 0.14.1+).

An embedded database means that the database is integrated as an inseparable part of an application software. The database engine runs inside the same JVM while the application is running.

An externalized database means that the database is separate from the application software. It is installed as service on a machine different from the ones hosting DHuS services.

The following databases are available in DHuS:

- Relational database
- Non-relational database

HSQL DB and Solr are the databases supported for the embedded case.

PostgreSQL DB and Solr are the databases supported for the externalized case.

3. DHuS deployment modes

DHuS can be installed and configured to run in three different modes. The purpose of having different modes is to cope with increasing user load and amount of data to be managed. The modes, in order of performances (and requirements) are:

- *Single instance mode (Section 3.1);*
- *Front-End/Back-End Mode (Section 3.2);*
- *Scalability 2.0 mode (Section 3.3);*

3.1 Single Instance mode

The deploy in Single Instance mode foresees the set-up of one Data Hub instance exposing its catalogue, i.e. the list of Sentinel' products available in the instance, to the users.

The following functionalities are managed by this instance:

- data harvesting (ingestion);
- management of user requests;
- products publication to end users.

Resources allocation for the single instance depends on the user load and the ingestion rate. Values provided for single instance apply (refer to Section 4.1.1).

3.2 Front-End/Back-End Mode

Deploy in Front-End/Back-End mode foresees the set-up of two (or more) Data Hub instances connected via OData synchronizers (sections 5.4 and 7.4). Depending on the kind of OData synchronizer active among the Data Hub instances, this deployment can be set up in two different ways:

- the data harvesting functionality (ingestion) is managed by one or more DHuS instances, called "*Back-Ends*";
- the management of user requests and products publication to end users is managed by **one** DHuS instance called "*Front-End*".

Resources allocation for the FE and BEs instances depends on the user load and the ingestion rate. Values provided for single instance apply (refer to Section 4.1.1).

3.3 Scalability 2.0 Mode

The objective of the configuration in externalized mode is to have several DHuS instances acting as one to share the user load and the products information: the deployment in externalized mode is completely transparent to the user.

Here in after we list the necessary actors for the deployment of DHuS in scalability 2.0, however resources allocation for all the actors depends on the user load and the product harvesting (ingestion/synchronization) rate.

- **Proxy:** A proxy is needed for load balancing among the nodes. It must be configured to redirect incoming traffic to the DHuS nodes based on a roundrobin load balancing algorithm. Please refer to the proxy documentation for instructions on how to implement this.
- **Relational Database:** it is installed as service on a machine different from the ones hosting DHuS services. It is the only product and user information (e.g. profile changes) data source and it is shared among all DHuS nodes. At now, only PostgreSQL software is supported. Additional PostgreSQL instances can be used for redundancy (please refer to software documentation for instruction on how implement this).
- **Non-Relational Database:** it is installed as service on a machine different from the ones hosting DHuS services. It is the only product indexes data source. So far, only Solr software is supported. Additional Solr instances can be used for redundancy (please refer to [RD 10] for instruction on how implement this).
- **Non-Relational Database orchestrator:** it is installed as service on a machine different from the ones hosting DHuS services. It is in charge of configuration files management for the non-relational database and redundancy management in case of additional Solr instances. So far, only Zookeeper software is supported.
- **Master:** The DHuS master is the one in charge of the ingestion/synchronization of products. Please note that the DHuS master is functionally equivalent to the DHuS nodes.

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

- Nodes:** The DHuS nodes are DHuS instances towards which the user traffic is redirected from proxy. It is mandatory that master and nodes share the same DataStores to allow access to ingested/synchronized products (please refer to 6.1.2 for details about the DataStore configuration). The product deletion and eviction can be executed on all nodes.

Tomcat session and EhCache replication mechanism shall be configured properly in the **dhus.xml** file of all DHuS instances, as described in [RD 11].

Here follows a representation of the deployment in scalability mode:

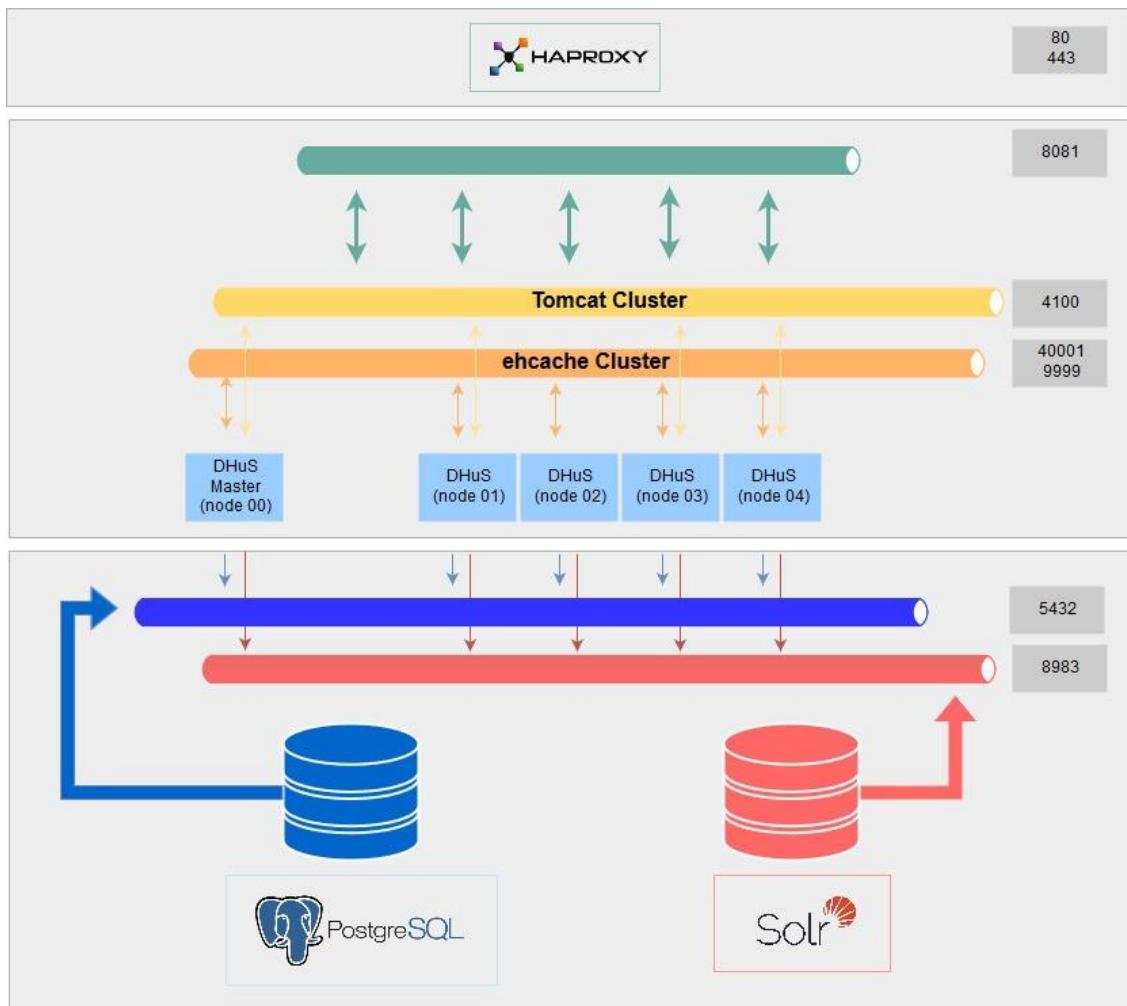


Figure 3 DHuS Scalability 2.0 deploy

4. Installation pre-requisites

4.1 Infrastructure Requirements

4.1.1 Single instance and FE/BE Mode

The technical specifications of each machine hosting the DHuS software (both in Single Instance and FE/BE Mode) are provided in the following table:

	MINIMUM	MEDIUM	HIGH
CPU Core Number	4	24	32
RAM	8 GB	32 GB	48 GB
LOCAL DISK		> 1 Gb	

Table 3 Singe Instance & FE/BE Mode Infrastructure Requirements

4.1.2 Scalability 2.0

The requirements for the Scalability 2.0 mode are described in the [RD 11] document, that provides a detailed step-by-step DHuS scalability 2.0 installation guide.

4.2 Network Requirements

DHuS is accessed primarily via HTTP and FTP interface. The Installation procedure of the DHuS Software must be performed using a non-privileged user (not root); application installed in this way cannot start services listening on ports numbers smaller than 1024. By default, the HTTP interface is reachable on 8081 port that must be opened for inbound requests. The DHuS FTP service is reachable, by default, on 2121. The DHuS requires for some of his functions a mailing service based on an external SMTP server. Following table describes the default DHuS network ports configuration:

SERVICES	INBOUND	OUTBOUND
HTTP	8081	External map server port External Nominatim service port
HTTPS	443	-
FTP	2121	-
SMTP	-	25

Table 4 Network Requirements

Please note that, since the UI Map is retrieved from an external service (and not internally from DHuS), the machine where DHuS is installed shall be able to access a map server (e.g. OpenStreetMap (<https://www.openstreetmap.org>)). Moreover, DHuS provides also an interface with Nominatim (<http://nominatim.openstreetmap.org/>), consequently the machine where DHuS is installed shall be able to reach the service.

4.3 Software Requirements

DHuS software is fully written in Java and can be considered portable to any platform providing the JRE (Java Runtime Environment). The DHuS supports:

- Java RE version 8 (since 0.13.4.21-1 DHuS is no longer compatible with Java 7 version)

For what regards the operating system, the distribution is specifically tailored for Linux based systems and has been tested on the following distributions:

- Debian >= 7.7
- Red Hat >=6.7
- Centos >= 6.6
- Mac OS > =El Capitan (run but not tested)

It is recommended to use a Linux Operating System working on a multithread environment running in 64bit.

5. Administration Functionalities

5.1 Login

Once the installation package has been successfully installed (as described in Section 4), the DHuS server can be accessed online ([https://\[DHUS IP ADDRESS\]](https://[DHUS IP ADDRESS])) or on local URL (<https://localhost:PORT/>). To access the administrator panels, it is first necessary to login as root, using the default settings.

5.2 Product Ingestion

The DHuS allows the ingestion of Sentinels products using the following methods:

- Ad hoc upload (Section 7.2.1);
- Ingestion via File Scanner (Section 7.2.2);
- Ingestion via OData (Section 7.2.1.2 and 7.2.2.2).

5.2.1 Ad hoc product upload

Ad hoc upload feature is available only to the administrator. DHuS system makes available an incoming space to let the user upload a product via HTTP/HTTPS protocols. Once uploaded, data is processed to be referenced by DHuS clients. This panel gathers all the information necessary to perform the upload (at least the path to the product).

Optional: *Assignment of a product to a collection is manually set by the uploader. A product can be included in any collection.*

Ad hoc upload is not cumulative: only one product can be uploaded at a time.

5.2.2 Upload via File Scanner

Cumulative / parallel uploads can be performed creating a File Scanner (see Section 7.2.2).

Optional: Assignment of a product to a collection is manually set by the uploader. A product can be included in any collection.

File scanner panel including the following:

- scan from pattern and from product type;
- directory recursive scan;
- scan could have parameters to automatically add scanned data into specific(s) collection(s) once downloaded;
- download parameters : retry count, retry delay, periodical scan;
- description of failure behavior;
- supported client protocol file/ftp; capability to remove data once downloaded.

5.3 Product deletion

Product deletion of a product can be triggered via UI or directly via OData API.

In particular, it permits to:

- 1) **Delete a product via UI.** The deletion is accepted after the declaration of a cause description. At the end of the process, the product:
 - is not present in Product OData entity;
 - is present in DeletedProducts OData entity with the description of the delete cause;
 - is not present in OpenSearch;
 - is not present in the GUI.
- 2) **Delete a product via OData - No cause specified.** The deletion is launched via command line. At the end of the process, the product:
 - is not present in Product OData entity;
 - is present in DeletedProducts OData entity;
 - is not present in OpenSearch;
 - is not present in the GUI.
- 3) **Delete a product via OData - Cause specified.** The deletion is launched via command line. At the end of the process, the product:
 - is not present in Product OData entity;
 - is present in DeletedProducts OData entity with the description of the delete cause;

- is not present in OpenSearch;
 - is not present in the GUI.
- 4) **Delete a product via OData - Complete deletion.** The deletion is launched via command line. At the end of the process, the product:
- is not present in Product OData entities;
 - is not present in DeletedProducts OData entities;
 - is not present in OpenSearch;
 - is not present in the GUI;

Please note that in the "Front-End/Back-End" deploy, once activated, the deletion acts on databases and archive according to the instance on which it is triggered:

- If the deletion is launched on a DHuS FE instance connected to a BE via OData metadata synchronizer, the process will only erase data from the Relational database and the Non-Relational database of the FE. The BE will not be affected by any change, in fact the product will not be deleted from the BE databases and archive.
- If the deletion is triggered on a BE instance, the process will erase data from databases and archive. Consequently, if the BE is connected to a FE with an OData metadata synchronizer, to avoid inconsistencies in the FE, it will be necessary to delete the same data also from this instance (otherwise product downloads and product inspection will not work).

5.4 Synchronizer

The DHuS provides end users an OData synchronizer service able to populate a DHuS instance with the data stored on the rolling archive of another DHuS instance.

Two different kind of synchronization can be performed:

- 1) **Metadata synchronization:** Copy of product metadata from a DHuS instance to another. The copy is performed according to predefined synchronization selection criteria

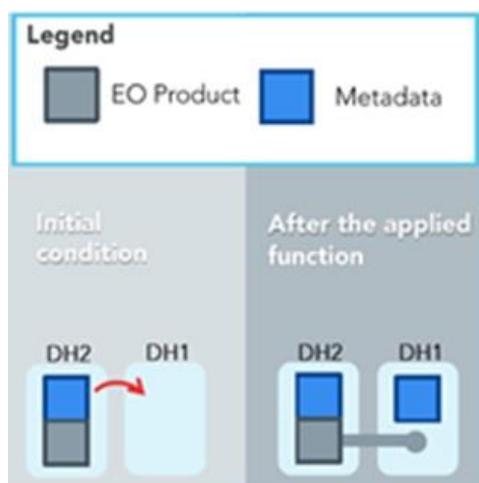


Figure 4 Metadata Synchronization

- 2) **Product/Remote Synchronization:** Copy of product and metadata from a DHuS instance to another. This is different from a product re-ingestion since during the product synchronization the UUID of the product is not re-generated but it is synchronized from the BE.

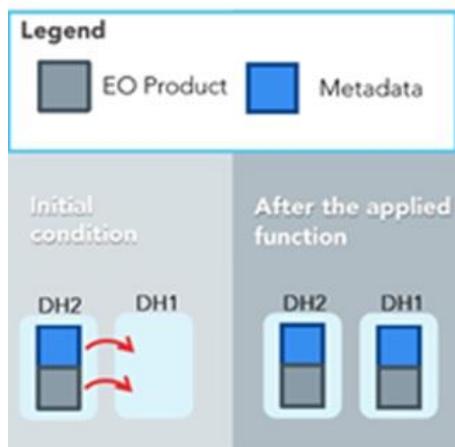


Figure 5 Products Synchronization

In case the rolling archive of the Back End contains some products not present in the Front End, once the synchronization runs, the synchronization mirrors products present in the Back End instance respecting the synchronization specification (according to the chosen configuration) and that are not in the database of the Front End instance (in case of metadata synchronization, only the metadata will be mirrored).

Warning: Please note that it is not possible to do metadata synchronization between two Front-End instances; in fact, the metadata synchronizer needs to access the LocalPath property of products to complete the synchronization of metadata with success. Indeed in this case, quicklook and thumbnail will be not shown in the DHuS instance requesting metadata.

Note: Please note that only backward compatibility is supported for synchronization purposes; in order to have a correct functioning of the Synchronizer feature the Front-End running DHuS version shall be higher or equal to the Back-End running DHuS version.

It is possible to configure multiple synchronizers fetching products from different data sources. In case of more than one synchronizer set, they will be triggered consequentially.

A synchronizer is characterized by the following parameters that shall be configured when the synchronizer is created:

- **Label**= Name of the synchronizer. Must be unique.
- **Service URL**= `https://[Back-End_DHuS_address]/odata/v1`
- **Service Login Username**= User name of a user registered in the Back end. In case of metadata synchronization, the archive manager rights enabled are requested for this user, otherwise the synchronization between Back End and Front End would be incomplete. In case of product synchronization, a normal user (meaning with search and download rights) can be used.

- **Service Login Password**= password of the user in the previous step.
- **Schedule**= how often the synchronizer shall be running. This shall be configured according to the crontab syntax. A schedule is configured according to the following pattern (refer to RD 14):
 - Seconds Minutes Hours Day-of-month Month Day-of-week [Year]
- **Remote incoming**= path of the incoming folder configured for the DHuS installed as the Back End instance. The role of the remote incoming is linked to the kind of synchronizer: metadata synchronizer ("Remote Incoming" field shall be set) or product synchronizer (empty "Remote Incoming" field).

Moreover, the configuration of such depends also on the DataStore used by the BE instance, please find below a summary of the different possible configurations:

	Synchronizer type	Remote incoming	Notes
BE incoming in HFS	Product	N	
	Metadata	Y	Absolute path of the incoming folder configured for the DHuS installed as the Back End instance. The FE will have access to products, quicklook and thumbnails that remain in the BE incoming folder.
BE incoming in openStack	Product	N	
	Metadata	N	The dhus.xml <DataStore> section of the FE instance shall report the description of the openstack DataStore of the BE (where the products are located) in readOnly=true (see 7.16 dhus.xml DataStore section). The FE will have access to products, quicklook and thumbnails that remain in the BE openstack container.

Table 5 Synchronizer type vs Remote incoming configuration

One last possible configuration for the remote incoming is the following: "Copy product" field is set as false (metadata synchronizer) and the remote incoming is not set. In this case, the FE will basically be a catalogue of metadata. This means that it will not have access to products, quicklook and thumbnails.

- **Request**= "start" or "stop"
- **Target Collection**= is a nullable and modifiable property with no default value. If specified, the synchronized products will be filled in the referenced FE Collection. Example: collection_name_FE.
- **Page size**= number of products synchronized at each synchronizer run. The PageSize parameter for the OData product synchronization with copy is the number of downloads happening in parallel, however to avoid idling download threads, the download task queue is overfed, thus you may have between 1 and PageSize * 2 download tasks submitted. It is suggested to set this parameter lower than the total number of threads allowed in the system.

- **Copy product**= this parameter determinates if the synchronization is a metadata synchronization or is a products synchronization:
 - **False**: the synchronizer will synchronize only the metadata of the products (the products are stored only in the Back End incoming folder).
 - **True**: the synchronizer will make a copy of the products in the Front End incoming folder during the synchronization.

Note that if this parameter is set as true, the "Remote Incoming" field shall be set as empty.

- **Filter Parameter**= following the OData filters syntax (see [RD 6]), it filters the products and synchronizes only the ones respecting the filter. Example: `substringof('S1A_', Name)`. For reference see RD 15 .
- **Source Collection**= Back End Collection filter: only the products on the Back End instances, which are belonging to the configured collection, will be synchronized to the Front End instance. Example: `Collections('collection_name')`. If this field is left empty, the synchronizer will synchronize every products without any check on their Back End collection.
- **Geofilter** = Geographical filter to synchronize only products belonging to a specific area from the BE. This filter is a post-filter, which means that it is executed client-side, not server-side.

This Property is a String with a specific syntax: <OPERATOR> <Well-Known-Text Shape>.

- <OPERATOR> can be:
 - *within* (a product's footprint is inside the given shape)
 - *contains* (a product's footprint contains the given shape)
 - *disjoint* (a product's footprint and the shape don't overlap)
 - *intersects* (a product's footprint and the shape overlap)
- <Well-Known-Text Shape> is a topologically valid shape written in WKT (See: https://en.wikipedia.org/wiki/Well-known_text)

Example

To synchronise products whose footprint overlap with Iceland, it could be possible using the following GeoFilter value:

```
intersects POLYGON((-24.61903123097289 63.345943833554685, -13.125342536439836 63.345943833554685, -13.125342536439836 66.61007811487349, -24.61903123097289 66.61007811487349, -24.6190))
```

- **Last Creation Date** = Filter on the `CreationDate` of the products stored in the Back End instances. Products with creation date \geq of the one here indicated will be synchronized. This not-nullable property is used by the OData product synchronizer to fetch pages of products in the right order. Once the synchronizer ends its run, this field is updated with the creation date of the last synchronized product.

Please, note that the field `CreationDate` is not synchronized from BE to FE, so:

- the “Creation Date” of a product in the BE represents the availability date on the BE instance of the DHuS during the ingestion process (its value is set to the current Date when the row is inserted in the Database slightly antecedent to the ingestion date)
- The “Creation Date” of a product in the FE represents the moment in which the product has been published in the FE instance.

Note: Until 0.12.5-6, the OData product synchronizer used the LastIngestionDate instead of LastCreationDate (using the same approach described above).

5.5 User Synchronizer

The DHuS provides an OData User synchronizer service able to populate a DHuS instance with the users stored on the rolling archive of another DHuS instance. The user synchronizer shall be created in the instance where the users are going to be imported (Front-End) and it can be activated by users with “user manager” rights.

The user synchronizer retrieves users (from a Back-End) following the user creation date criteria, meaning that it retrieves users from the oldest to the most recent. Here follows an example of the query performed by the User synchronizer:

```
http://DHuS_IP:8081/odata/v1/Users?$top=100&$orderby=Created
```

Refer to Section 7.5 for the management of User Synchronizers.

5.6 User management

DHuS implements a user management system that prevents uncontrolled accesses and manipulations from unauthorized users. DHuS proposes a user authentication and authorization strategy defined in its internal Database. Users are able to register or sign-in and the administrator are able to configure the user/group permissions from the Web user interface.

By means of the dedicated management panels in the UI (Section 7.7.1), the administrator can:

- create and delete any user;
- lock and unlock any user;
- update a user profile, including password reset;
- authorize the user to access a list of services.

User management is also possible via OData API (Section 7.7.2); the administrator can perform the following action for any users:

- creation and deletion;
- update profile;
- lock and unlock.

5.7 Collection management

DHuS service allows gathering products into collections. The administrator can manage collections, creating or deleting them by means of the dedicated Collection Management Panel (Section 0).

5.8 System management

The System management is used to configure basic information in the system. By means of the dedicated management panels in the UI (Section 7.9), it is possible to:

- configure the SMTP server address, the username, password and e-mail account details to send communications to the users;
- contact the DHuS Support Team sending an e-mail to eosupport@copernicus.esa.int for any support information;
- change the administrator password;
- configure a periodical dump of the database.

5.9 Product Eviction

The Data Eviction Service is responsible for removing data to keep to the Data Store compliant to sizing or data-offer constraints.

DHuS provides several tools to evict products or product metadata, in the form of customizable eviction, automatic DataStore size management and linked eviction.

5.9.1 Customizable Evictions

This feature allows fine data management on a DHuS instance by declaring and customizing evictions with several properties such as OData filters. An eviction can be either:

- **Hard**, which affects both physical products, product metadata, quicklooks and thumbnails;
- **Soft**, which only affects physical products.
- **SAFE**, which is a Soft eviction acting on products in the local archive only if they are present also in a different instance, as a Remote DHuS instance. Every Soft Eviction can be triggered in SAFE mode. Please refer to 5.12 and 7.10.2.7 for further details. Please note that if products are present in more than one DataStore Soft Eviction will act always in SAFE mode: no products will be made offline, even if the eviction is not triggered in SAFE mode.

Multiple evictions can be declared in a DHuS instance and it is possible to schedule different cron schedules for each eviction configured. If multiple evictions are set, they will be put in a queue of evictions.

The following properties are used to describe and configure an eviction:

- **Name**

The name of the eviction, used as a reference in OData. Must be unique.

- **MaxEvictedProducts**

The maximum number of products that can be evicted during a single eviction run.

- **KeepPeriod**

The minimal time after which a product can be evicted. The unit of measure is specified in the parameter "KeepPeriodUnit".

- **KeepPeriodUnit**

The time unit of the KeepPeriod property. It is possible to define the minimal keeping period for a product on sub-daily basis (e.g. hourly, by minutes). Default value is "DAYS".

- **Filter**

OData filter used to determine which products can be evicted by this eviction. This filter works the same way as the ones used on the Products entity set.

- **OrderBy**

An OData ordering clause used to determine in which order products will be evicted by the eviction.

- **TargetCollection**

The collection targeted by this eviction. Only products that are part of this collection can be evicted by this eviction. If the TargetCollection property is set to null, products will be evicted regardless of their collection.

- **SoftEviction**

If set to true, only the physical data of products will be evicted from the storage, not metadata and derived products (quicklooks, thumbnails). Moreover the "localpath" and "online" metadata will be updated. The first one, visible only for administrator, will be shown as "null" and the second one will be set as "false".

- **Status**

The current status of this eviction. The "Status" of Evictions is a read-only property and it can only be changed by the underlying system. Possible status are:

- STARTED if the eviction is running;
- QUEUED if the eviction was triggered but another one is currently running;
- STOPPED if the eviction is not running.

- **Active**

Boolean property to indicate whether the eviction will be triggered on schedule.

- **Schedule**

This property represents the schedule configured using the CRON syntax. This CRON property will affect only the associated eviction.

In the dhus.xml file coming from the distribution, an eviction named "DefaultEviction" is set and it is generated automatically by the DHuS at the start-up. This "DefaultEviction" does not have a declared schedule; it can be configured by declaring an eviction CRON in the dhus.xml configuration file or via AJS UI.

5.9.1.1 Actions

It is possible to perform the following actions on a Customizable eviction:

- **Creation:** allows creating a customizable eviction. It can be performed via AJS GUI and OData API.
- **Update:** allows modifying the eviction properties already set. All the eviction properties, except for Name and Status, can be updated. This action can be performed via AJS GUI and OData API.
- **Stop:** allows stopping the current run of the eviction. The Eviction will act again at the next run if a schedule property is set. Stop can be performed via OData API.
- **Deletion:** action to perform in order to remove an eviction rule set. If an eviction run is on-going at the time of the deletion, the run will end and then the eviction will be deleted. The eviction rule will be removed both from the /Evictions OData entity (refer to [RD 6]), from GUI and from dhus.xml as well.
- **Queued:** allows manually triggering an eviction run. When this action is called, the concerned eviction is started and its Status becomes "STARTED". If another eviction was already running, the concerned eviction is put in a queue and its Status becomes "QUEUED", it will then be started once the previously running eviction is finished.

All the listed actions can be performed within the DHuS at runtime. Please refer to Section 7.10 for further details.

5.9.2 DataStore Automatic Size Management

DataStores can automatically manage its size and perform a soft eviction if necessary, this feature is known as "Automatic On-Insert Eviction":

- Each DataStore may declare a maximum size, and keep track of its size as products are inserted or deleted.
- Each DataStore keeps track of its size during product insertions or evictions (note: A DataStore cannot measure its own size statically). Every time a product is inserted into a DataStore, old products may be evicted if the new product causes the DataStore to exceed its maximum size.
- This type of eviction is "soft", meaning that products are evicted only from the DataStore, the corresponding metadata, quicklook and thumbnail are left untouched.

The automatic size management on a DataStore can be configured using the "DataStores" OData interface or the dhus.xml configuration file; the properties related to this feature are part of the definition of the DataStore, with the following properties:

- `MaximumSize` in bytes, default value -1;
- `CurrentSize` in bytes, default value 0;
- `AutoEvict` a boolean, default value false.

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

The `MaximumSize` property represents the limit beyond which a DataStore will perform an eviction. The `CurrentSize` property keeps track of the size of a DataStore and is incremented each time a product is inserted in it. If this insertion causes `CurrentSize` to exceed `MaximumSize`, the DataStore will attempt to free enough space for the new product. This process only happens if `AutoEvict` is set to `true` and only as long as products are inserted in the DataStore.

5.9.2.1 Linked Eviction

An On-Insert Automatic Eviction can be configured so that it can follow the same eviction rules allowed using a Customizable eviction, i.e. the eviction properties listed in sections 5.9.1 and 7.10.2.1.

In order to do that an existing eviction configuration has to be linked to a DataStore configuration.

Products contained in a specific DataStore can be evicted following the rules defined by the properties of the linked Customizable eviction. The trigger remains the same as the Automatic On-Insert Eviction, i.e. insertion of products in the Datastore and thanks to the link to a Customizable Eviction, products can be evicted on the basis of the `KeepPeriod` property together with size criteria, meaning that products will be evicted only if the size threshold is exceeded and the selected `keepPeriod` is elapsed as well.

5.10 Auxiliary Data Files

Auxiliary Data Files are files of different format that can be used to “re-process” a product, if they are better than the ones originally used. These files are retrieved from a DHuS instance able to ingest ADF.

Two features dedicated to ADF are available in OWC Application:

- **Auxiliary File List**, that shows the list of auxiliary data files used to process a product. At present this feature is available only for Sentinel-1 and Sentinel-3 products.
- **Auxiliary File Search** lists the ADF retrieved from another DHuS instance and allows their download. This action is available only if the feature is enabled (see Section 7.12). At present this feature is available only for Sentinel-3 Level 1 and Level 2 products.

The comparison between the Auxiliary File List and the Auxiliary File Search is useful to understand whether it is appropriate to process a product with different Auxiliary Data Files.

Please refer to the OWC User Manual [RD 9] for further details.

5.11 Sentinel-1 PDGS LTA component Interface

5.11.1 Overview

DHuS implements a Sentinel-1 PDGS LTA component interface that allows the possibility to retrieve back from LTA soft-evicted products, restoring them on DHuS to be available for users.

The product catalogues of the Data Hub services will continue to provide access to the full set of data, searchable via the Graphical User Interfaces as well as the OData and OpenSearch APIs. However, some data will no longer be immediately accessible, and will be flagged as “OFFLINE”.

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

An Offline product is obtained performing a soft-eviction over the DHuS database; these products can be ordered by means of triggering the product download via both “Graphical User Interface” and OData API.

The download request activates the data retrieval chain: the product is downloaded from the LTA and it is restored in the Data Hub to be available for immediate download for all registered users (not just for the user who triggered the download). After a successful retrieval from the LTA, the Data Hub updates the product metadata `Contentlength`, `Checksum` and `CreationDate` properties on OData interface.

Please refer to the following image for the diagram detailing the interactions among the entire DHuS-LTA chain; example for a Service deployed in Scalability 2.0 mode is reported.

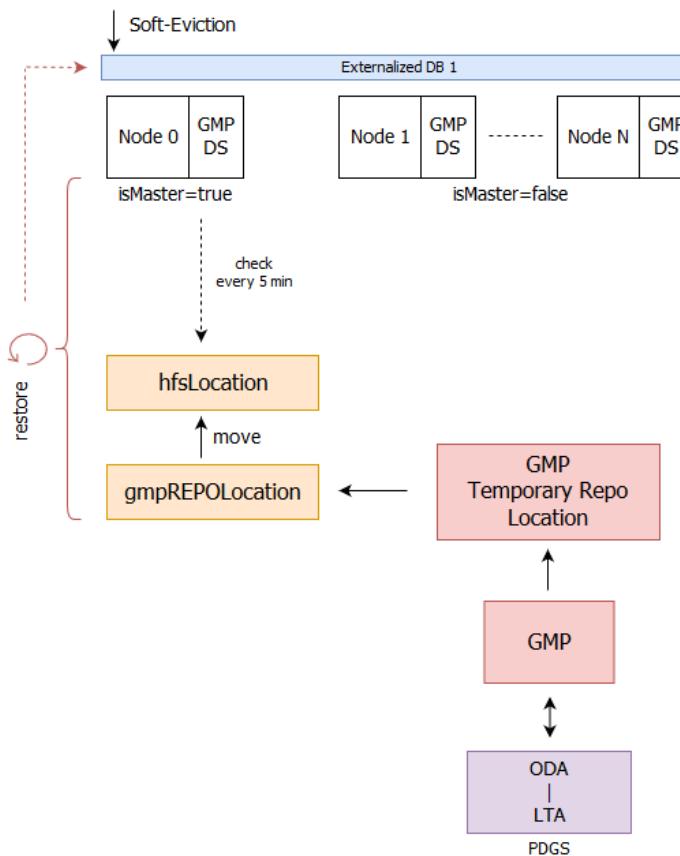


Figure 6 DHuS-LTA interfaces

5.11.2 Quotas

The interface with the Sentinel-1 PDGS can be subjected to quotas. DHuS administrator can configure the Hub so that orders to PDGS for retrieval of offline products are controlled and limited.

Here follows the specific case of quotas that can be configured on DHuS:

- maximum number of parallel requests per instance
- maximum number of orders per time range and per user

5.11.3 GMP

The retrieval of Sentinel-1 products from PDGC components (ODA PACs and LTA) is possible via "Get My Products" (GMP) system. Refer to RD 12 and RD 13 for further details about this software.

DHuS supports an interface with GMP software and it is able to insert correctly entries in the GMP database and verify when the product's download from PDGS is completed. This interface is the GMPDataStore, refer to Section 5.11.4 for details.

5.11.4 GMPDataStore

A GMPDatastore is characterized by the following parameters that shall be configured when it is created:

- **Name** = The reference property. It is unique, cannot be null, and cannot be updated.
- **Priority** = Represents the priority rank of the a DataStore, DataStores with the smallest value are accessed first within the system. It is recommended to configure a GMPDataStore with so that it is accessed last by the system (i.e. a higher value to its priority parameter).
- **ReadOnly** = true|false. Determines whether data can be written on this DataStore. It is recommended to set it as "false" if the DHuS instance is the one eligible for the restore.
- **MaximumSize** = threshold, in bytes, beyond which a DataStore will perform an automatic eviction, if set. Default value is -1.
- **CurrentSize** = keeps track of the size of a DataStore and is incremented each time a product is inserted into that DataStore, if this causes CurrentSize to exceed MaximumSize , the DataStore will attempt to free enough space for that new product. This process only happens if AutoEvict is set to true. Default value is 0. This property is read-only and can be modified only by DHuS or before startup.
- **AutoEviction** = true|false. Property to set the Automatic On-Insert Eviction for the datastore.
- **GMPRepoLocation** = location where GMP software stores (and possibly zips) downloaded products in zip format.
- **HFSLocation** = location of the HFS cache used by the GMPDataStore to restore products. Products retrieved from LTA are moved in this folder to be available for user download. It is an HFSDaStore embedded in the GMPDataStore to store products downloaded by GMP and make them available to users.
- **MaxQueuedRequest** = count of ongoing fetch requests queued until the DHuS has to refuse a new downloads using GMP.
- **MaxQueryPerUser** = maximum number of user orders requestable in the configured time range. Default is 10.
- **Timespan** = Time to live for user ordering quota entries in milliseconds. Default is 1 month.
- **IsMaster** = true|false. Property to configure the GMP master instance. Management of downloads and restore action is done only by the master instance. The GMP master instance verifies if new products are present in the **GMPRepoLocation** folder every 5 minutes.

- **DatabaseUrl** = Definition of GMP database path.
- **User** = User to access GMP database.
- **Password** = Credentials to access GMP database.
- **AgentId** = Agent identifier in the GMP “agent” table to be used by GMP software for the products retrieval.
- **TargetId** = Target identifier in the GMP “target” table to be used by GMP software for the products retrieval.

5.12 Sentinel-2 Archive Interface

5.12.1 Overview

DHuS implements a Sentinel-2 Archive interface that allows the possibility to use another DHuS instance as if it is a read-only archive; it uses the OData interface of a remote DHuS instance in order to retrieve the data of products for downloads and nodes browsing.

The product catalogues of the Data Hub services will continue to provide access to the full set of data, searchable via the Graphical User Interfaces as well as the OData and OpenSearch APIs. However, the involved Sentinel-2 data are evicted in SAFE mode and the users’ requests are forwarded towards the configured Sentinel-2 archive. The entire configuration is transparent to end users. These products can be ordered by means of triggering the product download via both “Graphical User Interface” and OData API.

The user’s request (download or node inspection) activates the data retrieval from the Sentinel-2 archive Hub configured and the results are provided to the user.

5.12.2 RemoteDHuSDataStore

The connection between the DHuS and the Sentinel-2 Archive Hub is possible via the configuration of a RemoteDHuSDataStore in the DHuS instance exposed to the end users.

Remote DHuS DataStores operate like read-only DataStores in the fact that it represents a means of physical products storage. As such, it can open streams for user downloads, including partial downloads, and provide nodes for browsing. A DHuS instance interacts with its Remote DHuS DataStore by using the OData interface of the Remote DHuS instance.

A RemoteDHuSDataStore is characterized by the following parameters that shall be configured when it is created:

- **Name** = The name of the DataStore.
- **ServiceUrl** = The URL of the OData “v1” service of the Remote DHuS instance.
- **Login** = The login to be used to request the Remote DHuS instance.
- **Password** = The password to be used to request the Remote DHuS instance.

- **ReadOnly** = true. Determines whether data can be written on this DataStore. This type of DataStore behaves as read-only regardless of its ReadOnly parameter.

Please note that the Remote DHuS DataStore has to be configured both in BE and FE instance and if FE is deployed in Scalability 2.0 mode the Remote DHuS DataStore shall be created in every deployed nodes.

5.12.3 SAFE-Eviction

The involved Sentinel-2 products are evicted in SAFE mode in order to allow the physical deletion of products from the original storage in a safe way.

A dedicated functionality in the OData Eviction API is present. This operation soft-evicts products from a single DataStore only if those products are available in another DataStore (e.g. the RemoteDHuSDataStore). This kind of eviction can be only queued manually as described in 7.10.2.7.

After a successful SAFE-Eviction, the Data Hub updates the product metadata `LocalPath` property (visible only to Administrator users) on OData interface.

5.12.4 Pre-requisites

In order to activate the RemoteDHuSDataStore function, the following pre-requisites must be met. Consider a DHuS instance called "DHuS A" and its Remote DHuS instance called "DHuS B":

- DHuS A must be in version 0.14.7 or higher to support RemoteDHuSDataStore implementation.
- DHuS B must be in version 0.9 or higher to support downloads and nodes browsing via OData.
- DHuS A and DHuS B must be able to communicate via network (internal or external).
- DHuS A must have a valid login and password in its Remote DHuS DataStore configuration to connect to DHuS B.
- Products on DHuS A and DHuS B must have the same UUIDs.
- Since the requests performed by DHuS A on DHuS B are normal user requests, there should be no download quotas configured on DHuS B.

6. Installation and Configuration Procedures

6.1 Software configuration

DHuS configuration files are:

- **dhus.xml**; located in the `etc` directory
- **server.xml**; located in the `etc` directory
- **start.sh**; located in the installation directory

- **restart.sh**; located in the installation directory
- **stop.sh**; located in the installation directory
- **log4j2.xml**; located in the `etc` directory
- **suggester.dic**; located in the `etc` directory
- **opensearch-description-file.xml**; located in the `etc` directory
- **temporary_files_ehcache.xml**; located in the `etc` directory
- **AJS GUI**
 - **appconfig.json**; located in [DHUSDIR]/etc/conf directory
 - **styles.json**; located in [DHUSDIR]/var/tomcat/webapps/ROOT/conf directory
- **OWC**
 - **appconfig.json**; located in [DHUSDIR]/var/tomcat/webapps/new/conf directory
 - **styles.json**; located in [DHUSDIR]/var/tomcat/webapps/new/conf directory
 - **menu.json**; located in [DHUSDIR]/var/tomcat/webapps/new/conf directory
 - **theme.json**; located in [DHUSDIR]/var/tomcat/webapps/new/conf directory
 - **rules/*.xml**; located in the `etc` directory
 - **task_tables/*.xml**; located in the `etc` directory

The following subsection describes in details how to configure the major DHuS functionalities; for the complete list of configuration parameters, please refer to Sections 8, 9 and 10.

6.1.1 Storage configuration

This section describes how to configure the DHuS data storage, i.e. to use a File System (HFS) or an object storage (OpenStack Swift); this configuration shall be performed within the `dhus.xml` file.

Since DHuS v0.13.x, it is possible to configure DHuS to interface its data storage (physical data location) by using the `[DataStore]/DataStore` configuration: this configuration allows DHuS to interface both data storage on file system and Object storage.

Please note that the `[System]/archive incoming` parameter is not supported anymore since 0.14.7-2 version.

Thus, starting from 0.14.7-2 DHuS version, the DHuS data storage configuration shall be managed according to the following possible scenarios:

- Database coming from previous DHuS version
- Empty Database

Empty Database (no products)		
HFSDaStore configuration can be used in order to store product in a Hierarchical File System, as well as Object Storage configuration.		
Products stored via DataStore	File System (HFS) HFS e configuration can be used in order to store product into Object storage. Refer to Section 7.16.1 for instruction about this configuration.	Object storage (OpenStack Swift) OpenStackDataStore configuration can be used in order to store product into Object storage. Refer to Section 7.16.2 for instruction about this configuration.
Database coming from DHuS previous DHuS version		
Since DHuS 0.14.7-2, OldIncoming configuration is no longer supported. An automatic script, described in Section 11, is provided in order to automatically convert the old incoming configuration in the HFSDaStore configuration in order to store product in a Hierarchical File System.		
Old products stored via OldIncoming	File System (HFS) The script updateConfiguration.sh is in charge to convert the configuration automatically: during the migration from previous DHuS versions to the 0.14.7-2 one, products coming from Oldincoming configuration will be stored in a HFS DataStores. The <DataStore></DataStore> section of the DHuS.xml will be automatically filled.	Object storage (OpenStack Swift) N/A
Products stored via DataStore	File System (HFS) The HFSDaStore configuration already in use must be used in the dhus.xml file.	Object storage (OpenStack Swift) The OpenStackDataStore configuration already in use must be used in the dhus.xml file.

Table 6 Storage configuration on dhus.xml

6.1.2 DataStore properties and configuration

The DataStore Entity is an administration tool inaccessible to regular users and it is available through the OData v4 interface. A DataStore is a layer used by the DHuS to interface its data storage/s.

All the three installation modes (see Section 3) can be configured to use a Hierarchical File System storage or an Object storage. Currently DHuS supports the OpenStack Swift Object storage. Swift is a highly available, distributed, eventually consistent object/blob store (<https://docs.openstack.org/developer/swift/>).

Warning: multiple DataStores can coexist in DHuS; each DataStore is independent from the others. During the ingestion process, products are stored in all DataStores with write access. Please refer to [DataStore]/DataStore in [8.1] for details about how to configure DataStores.

Users with the role SYSTEM_MANAGER can access the DataStore entity and update its entries at this address:

[DHUS_HOST] /odata/v2/DataStores

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

HfsDataStore, OpenStackDataStore, GMPDataStore and RemoteDHuSDataStore inherit from DataStore. Every DataStore, regardless of its subtype, is located in the DataStores entity set.

Table 7 details properties communal to each DataStore:

PROPERTY	TYPE	DESCRIPTION
Name	String	The reference property. It is unique, cannot be null, and cannot be updated.
ReadOnly	Boolean	Determines whether data can be written on this DataStore. Please note that it does not mean that the entity itself is read-only. Note: The RemoteDHuSDataStore behaves as read-only regardless of its ReadOnly parameter.
Priority	Int32	Represents the priority with which DataStores are accessed within the system. DataStores with the smallest priority are accessed first. Please note that GMPDataStore has to be configured with the lowest priority than all the other DataStores present.
MaximumSize	Int32	Represents the threshold beyond which a DataStore will attempt to free enough space for that new product
CurrentSize	Int32	Counter of the current size of a DataStore
AutoEviction	Boolean	Parameter that allows to active an Automatic On-Insert Eviction on the DataStore

Table 7 Common DataStore entities properties

Properties present in the HFSDDataStore are specified here in after:

PROPERTY	TYPE	DESCRIPTION
Path	String	Local storage path.
MaxFileDepth	Int32	Maximum number of sub-folder for each stage.
MaxItems	Int32	Maximum number of file stored in each folder

Table 8 HFSDDataStore entity properties

Properties present in the OpenStackDataStore are specified here in after:

PROPERTY	TYPE	DESCRIPTION
Provider	String	Provider service to use, only "openstack-swift" is currently available.
Identity	String	Identifier for the authentication service.
Credential	String	Credential of the Openstack account to be used. The syntax is the following: <i>project:password</i>
Url	String	URL of Openstack authentication service.
Region	String	Each Region has its own full Openstack deployment, including its own API endpoints and networks. The region is linked to the Openstack account. The default value is <i>regionOne</i> .
Container	String	Name of the container to be used. Please note that the container shall be created manually before configuring this parameter. Please check 7.17 for further details about how to create a container in OpenStack.

Table 9 OpenStackDataStore entity properties

Properties present in the GMPDataStore are specified here in after:

PROPERTY	TYPE	DESCRIPTION
GMPRepoLocation	String	Path to the folder where GMP software store retrieved products in .zip format.

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

HFSLocation	String	Folder where the retrieved products are moved by GMPDataStore for being available for the user.
MaxQueueRequest	Int32	It defines max quota for instance.
MaxQueryPerUser	Int32	Quota for users in the time range selected.
Timespan	Int32	It defines time range in milliseconds for the user quota.
IsMaster	Boolean	true false. It defines the instance in charge of download and restore actions.
DatabaseUrl	String	GMP database path.
User	String	Username to access GMP database.
Password	String	Password to access to GMP database.
AgentId	String	Agent identifier used by GMP software for the product retrieval.
TargetId	String	Processing Archive Centres acting for the product retrieval.

Table 10 GMPDataStore entity properties

Properties present in the RemoteDHuSDataStore are specified here in after:

PROPERTY	TYPE	DESCRIPTION
ServiceUrl	String	The URL of the OData "v1" service of the Remote DHuS instance.
Login	String	The login to be used to request the Remote DHuS instance.
Password	String	The password to be used to request the Remote DHuS instance.

Table 11 Remote DHuSDataStore entity properties

6.1.2.1 Actions

The DataStores entity set supports CRUD (Create, Read, Update, Delete) operations on DataStores and allows to manage the configuration of DataStores within the DHuS at runtime.

Here after the list of the allowed actions:

- **Creation:** allows creating a DataStore.
- **Update:** allows updating DataStore properties.
- **Delete:** action to perform in order to remove a DataStore previously created. It will be removed from /DataStores OData entity (see [RD 6]) and from dhus.xml.
- **List products contained in the Datasore:** allows listing the total stored products in the DataStore.

Please refer to Section 7.16 for details about how to manage DataStores.

6.1.3 NetCDF Temporary Cache mechanism

DHuS implements a temporary cache feature whose objective is to allow the support for the inspection function of NetCDF-based files (Sentinel-3, Sentinel-5P, COSMO-SkyMed). This mechanism can be applied both in Single Instance and Scalability 2.0 modes. Please refer to Section 7.18 for details about its configuration.

6.1.3.1 Design

The first time NetCDF files are accessed within a product during ingestion, they are extracted and stored as temporary .dat cache files so that subsequent accesses can be done faster without having to perform the extraction again. Temporary cache files have been introduced to make sure this extraction is only performed once per process that is once per ingestion.

In addition, in a Front-End scenario, please note the First Node inspection does not open NetCDF files and no .dat file is created when inspecting the first node of a NetCDF product, i.e the cache mechanism is not involved in this case; cached files are created when the NetCDF node itself is inspected. The temporary file is generated the first time a user browses nodes inside of a NetCDF node (allowing faster browsing of child nodes).

The size of temporary cache files is proportional to the related products and usually larger since zipped products are compressed. So a dedicated physical space on the Virtual Machine hosting the DHuS instance should be dedicated to the temporary cache.

As described in the following sections, this cache feature can be:

- disabled configuring a proper option in the `start.sh` file (see Section 7.18.1 and 8.3);
- customized about location and size configuring proper options in the `temporary_files_ehcach.xml` and `start.sh` (see Section 7.18.2.2 and 8.3);
- shared between the DHuS nodes deployed in Scalability 2.0 configuring properly the `temporary_files_ehcach.xml` file (see Section 7.18.2.3).

6.1.3.2 Approach

NetCDF cache can be configured for both ingestion and inspection purposes. Please find below suggestion on how to configure the cached files location in an efficient way.

- **Ingestion (Back-End side)**

The netCDF cache should be configured (on NAS or on local partition) and it should be able to help the ingestion process.

- **Inspection (Front-End side)**

The netCDF cache should be configured only on local partition. If it is not possible to configure the netCDF cache on local partition, the suggestion is to not configure it at all. In this scenario, the response times experienced with a cache configured on a NAS are worst w.r.t. having a disabled cache.

6.1.4 OWC configuration

DHuS implements also a User Interface based on Polymer library: the Open Web Components (OWC) is a client side software following the Single Page Application approach, for space data dissemination.

The default configuration of OWC Application is oriented to DHuS operations, so it foresees that the DHuS UI and APIs are accessible from URL including "/dhus" substring (via proxy).

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

If the DHuS UI and APIs are accessible from different kind of URL, here follows details about how to visualize properly OWC UI after DHuS installation and start up:

- **Step 1:** Change directory to the following folder

```
<dhus_installation_folder>/var/tomcat/webapps/new/elements
```

- **Step 2:** Open the elements.html file in editing mode
- **Step 3:** Look for the string "window.location.href.split"
- **Step 4:** In this code block

```
var config = {  
    baseUrl: window.location.href.split("dhus")[0] + "dhus",  
    clientUrl: window.location.href.split("dhus")[0] + "dhus/new"  
    //baseUrl: "../", // dev config  
    //clientUrl: "./" // dev config  
};
```

comment the default baseUrl and clientUrl variables and remove comment from baseUrl and clientUrl for development configuration, as shown below:

```
var config = {  
    //baseUrl: window.location.href.split("dhus")[0] + "dhus",  
    //clientUrl: window.location.href.split("dhus")[0] + "dhus/new"  
    baseUrl: "../", // dev config  
    clientUrl: "./" // dev config  
};
```

Where:

- baseUrl is the URL composed by the IP address of the host and the port at which the DHuS is exposed,
- clientUrl is the URL used to access the local resources.

Step 5: Reload the web page http://<dhus_host>/new/ to play with OWC Application

Warning: The elements.html file is overwritten at each restart of the DHUS, so it is recommended to create a Backup copy of the file

6.2 Single Instance installation (HFS)

- **Step 1:** Create a user named dhus. Every step in the installation procedure, if not explicitly mentioned, shall be performed as dhus user.
- **Step 2:** Create the installation directory: `mkdir -p [install-dir]`
- **Step 3:** Download the DHuS package (shar package) and save it into the installation directory
- **Step 4:** Change the permissions on the file: `chmod +x dhus-XX.XX.XX.shar`
- **Step 5:** Launch `./dhus-XX.XX.XX.shar` (the package will autoinstall). Once executed, the system setting configuration file can be accessed and updated.

- **Step 6:** Edit the `etc/dhus.xml` configuration file and modify the `varFolder` variable to an absolute path of your choice. This directory will contain the databases folders (if deployed in embedded mode), the dump folders, the tomcat folder, etc. Eg: `<!ENTITY varFolder "/home/dhus/local_dhus">`
- **Step 7:** Start the DHuS entering the following command in the installation directory:

```
nohup /bin/bash ./start.sh &
```

The log files will be created in the `etc` directory after the installation.

6.3 Scalability 2.0 Mode installation

Please refer to [RD 10] for details about how install DHuS in externalized mode. In the manual the following information are detailed:

- Installation and Configuration procedures
- Migration of existing embedded HSQLDB to external PostgreSQL
- Migration of existing embedded Solr to external Solr
- Export/Import of external PostgreSQL DB and Solr

7. Administrator Procedures

7.1 Login

7.1.1 How to login as administrator

7.1.1.1 Via Graphical interfaces

The "Login" button is displayed in the upper right side of the DHuS Home page.

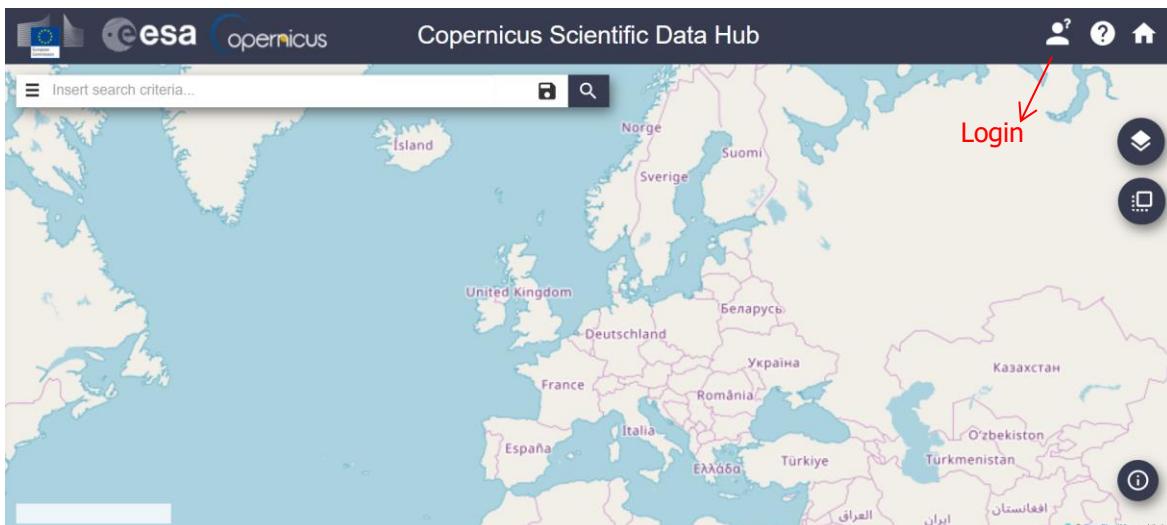


Figure 7 DHuS Login (AJS)

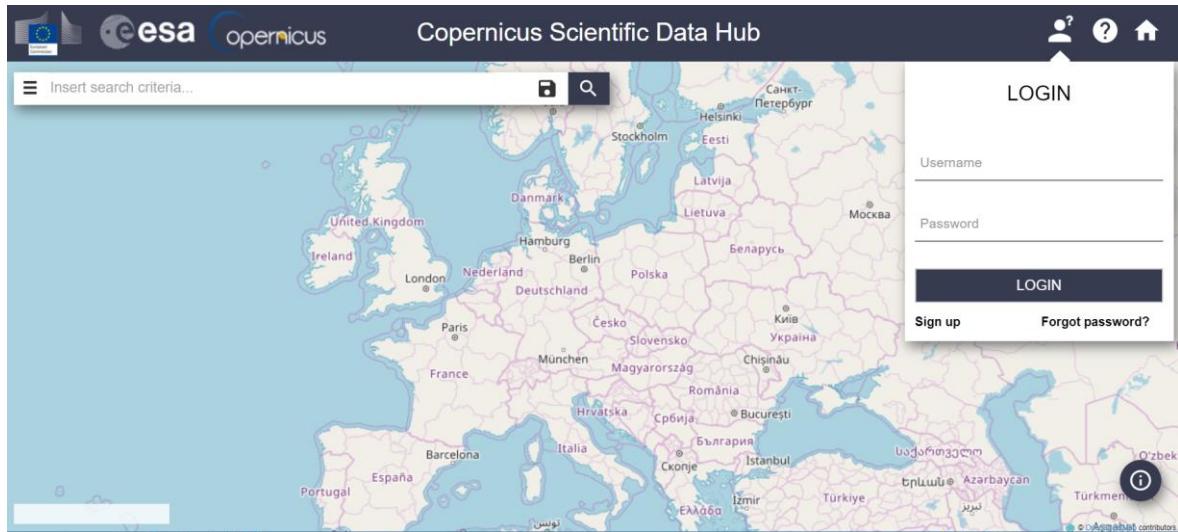


Figure 8 DHuS Login Panel (AJS)

Once the user has logged in, the Login badge displays information related to the logged user (email, allowed functions, etc...), refer to Figure 9.

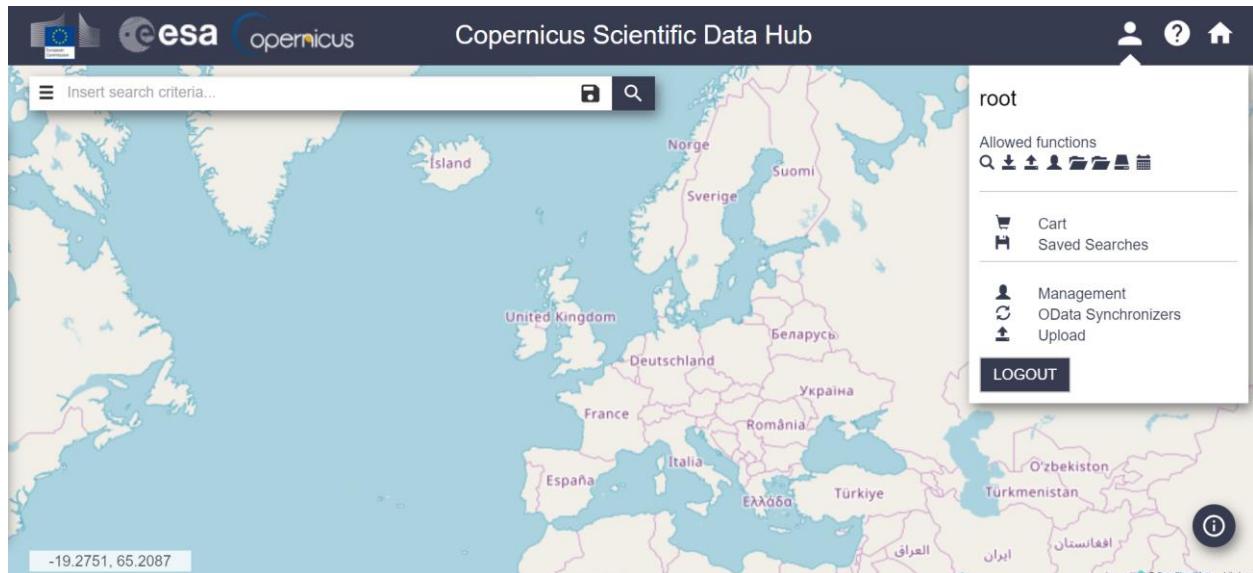


Figure 9 User Information display (AJS)

In OWC, the login button is displayed in the left side of the Home web page.

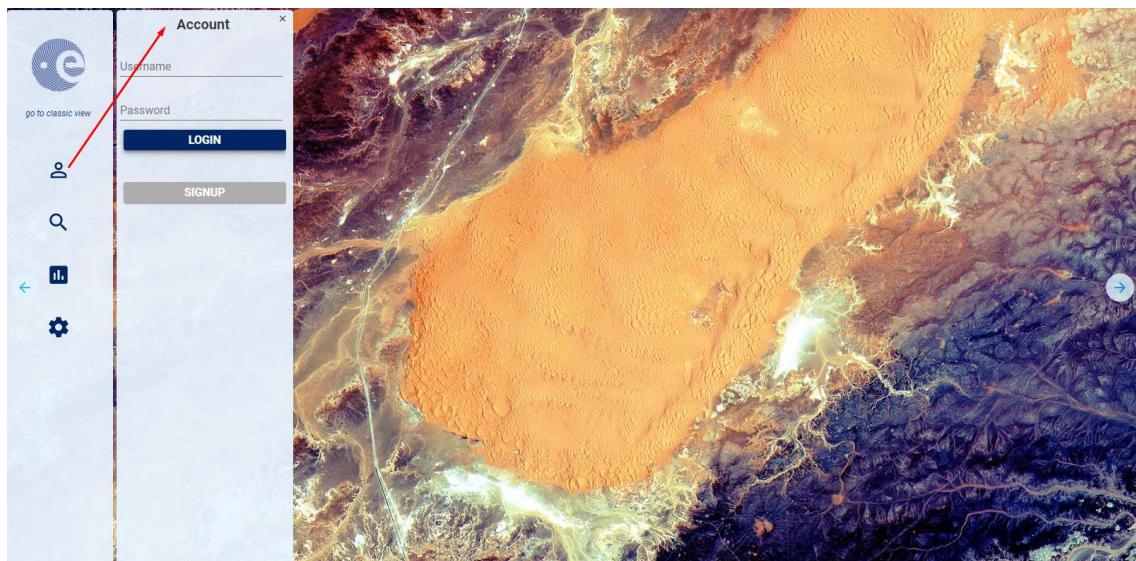


Figure 10 Login panel (OWC)

7.2 Product Ingestion

7.2.1 How to upload ad hoc product

7.2.1.1 Via Graphical interfaces

The product upload is accessible by the "Upload" menu:

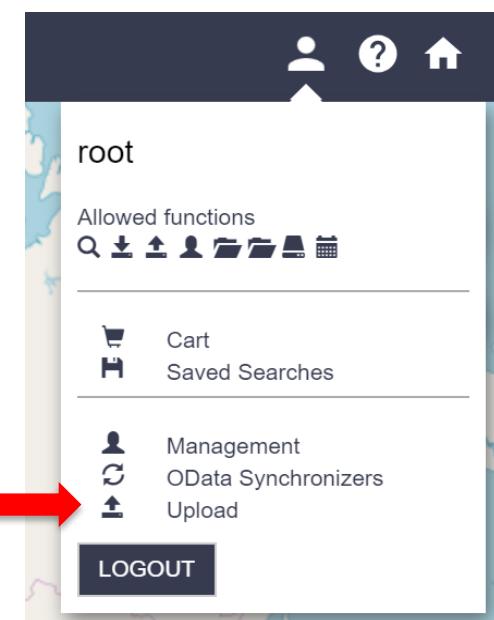


Figure 11 Products Upload (AJS)

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

Once in the "Upload panel", it is possible to perform the upload of a product: select the input products, then (optionally) select a collection in the list of collections and click on the "Upload" button. The upload will start and at the end of it, a pop up will notify that the upload is over.

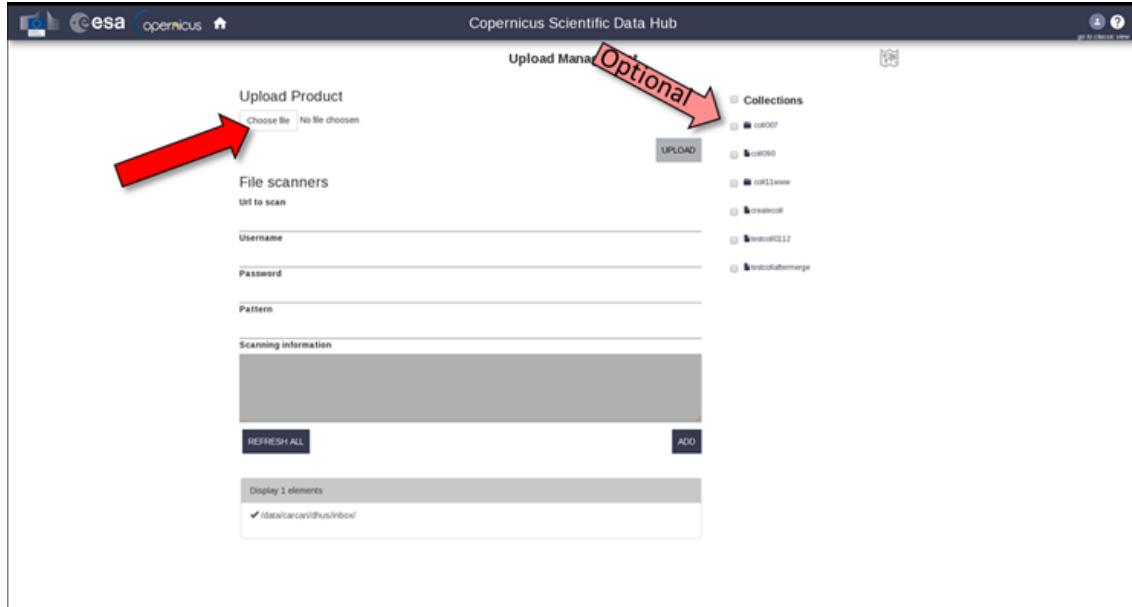


Figure 12 Upload Panel (AJS)

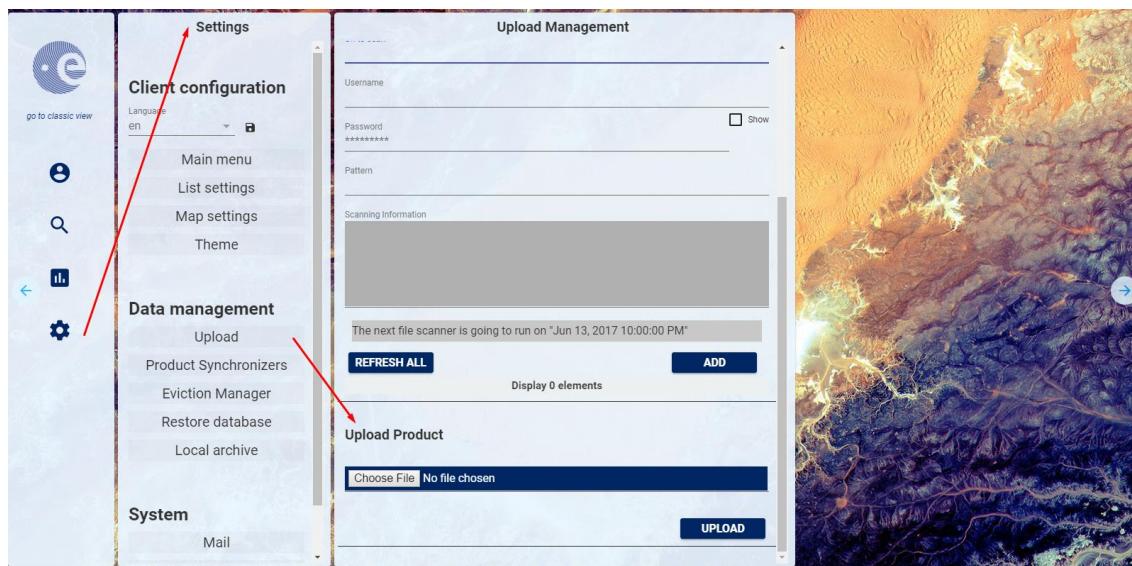


Figure 13 Upload Panel (OWC)

7.2.1.2 Via OData

The ingestion functionality can be triggered via OData API, the implementation of such process is currently not supporting the multi-thread management: one product at the time can be ingested.

How to ingest via OData?

- **Step 1:** Log in the machine where DHuS is installed
- **Step 2:** Download scripts to trigger the ingestion and put them in a working directory
https://github.com/SentinelDataHub/Scripts/tree/master/dhus_ingest_scripts
- **Step 3:** Set the environment variables to connect to the syncing instance of the DHuS eg:
 - export DHOST=localhost:8080
 - export DLOGIN=root
 - export DPASS=a
- **Step 4:** To trigger the ingestion process use the following command:

```
./ingest <path/to/data.zip>
```

Products will be visible under http://DHuS_address/odata/v1/Ingests

How to delete an ingestion record via OData?

To delete an ingestion record via OData launch the following command:

```
curl -X DELETE --basic -u ${DLOGIN}:${DPASS} \
"http://${DHOST}/odata/v1/Ingests(ID)"
```

or launch the command:

```
./deleteIngest <ID>
```

For deleting products ingested with methods different from the OData API please refer to Section 7.3.

7.2.2 How to upload via File Scanner

7.2.2.1 Via Graphical Interfaces

If the upload has to be periodic, a scanner can be configured with the panel highlighted by the red arrow in figure below (still accessible from the upload panel).

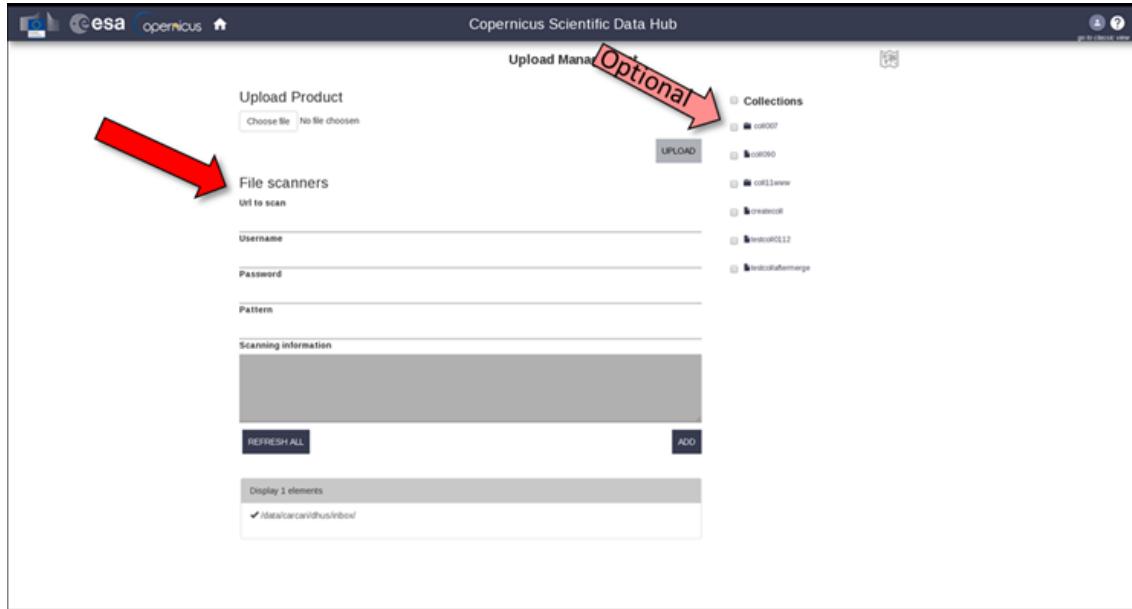


Figure 14 File Scanner Panel (GUI)

To create a file scanner:

- **STEP 1:** Access the upload panel;
- **STEP 2:** Fill the URL to scan field with the path of the folder containing the products (if the products are in the same machine where DHuS is installed, the field shall be filled as 'path/of/the/folder').
- **STEP 3:** If the products are located on an external data provider (accessible via ftp), configure the username and password to access the machine; otherwise the username and passwords will not be necessary.
- **STEP 4:** To upload just specific types of product, configure the 'Pattern' field according to the regular expression roles explained in <http://docs.oracle.com/javase/7/docs/api/java/util/regex/Pattern.html> (e.g. "S1[AB]_\p{Upper}{2}_(SLC|GRDM).*" to upload only the SLC and the GRDM products)
- **STEP 5:** Click on the 'add' button. In the lower part of the page it will be written 'when' the file scanner will be activated again.

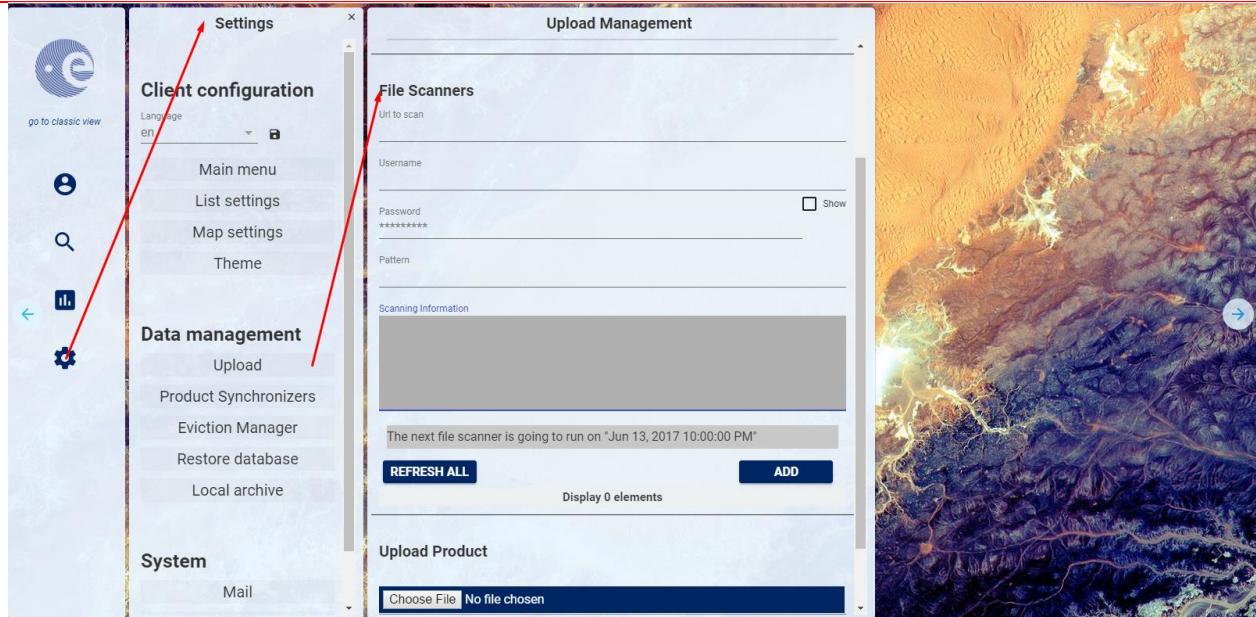


Figure 15 File Scanner Panel (OWC)

7.2.2.2 Via OData

Ingestion process can be executed using FileScanners; DHuS software supports the management of FileScanner using OData API.

Here follows command line samples, using cURL command, in order to manage the FileScanner creation, update and deletion at runtime:

- **FileScanner creation**

```
curl -u root:password -X POST -H "Content-Type: application/json" --data
'{"Url":"/path/to/inbox/directory", "Status":"", "StatusMessage":"",
"Active": false, "Username": "", "Password": "", "Pattern" : ""}'
http://localhost:8081/odata/v1/Scanners
```

- **FileScanner access**

```
curl -u root:password -X GET --basic
"http://localhost:8081/odata/v1/Scanners(2L)"
```

- **Filescanner update**

```
curl -u root:password -X PUT -H "Content-Type: application/json" --data
'{"Active": true}' "http://localhost:8081/odata/v1/Scanners(1L)"
```

- **FileScanner deletion**

```
curl -u root:password -X DELETE --basic
"http://localhost:8081/odata/v1/Scanners(0L)"
```

Please note that the OData interface allows the creation, update and deletion of the FileScanners, but it is not able to trigger them yet, i.e. it is not possible to start and stop the FileScanner.

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

Please also note that it is possible to schedule and de-schedule the created FileScanner (see the “FileScanner update” example in the list above).

Please find below an example of “Status” and “StatusMessage” that can be used during the creation; please note that these fields are not mandatory and can be left empty.

```
<d>Status>added</d>Status>
```

```
<d>StatusMessage>Added on Wednesday 23 August 2017 - 10:55:19</d>StatusMessage>
```

7.3 Product deletion

7.3.1 How to delete a product

7.3.1.1 Via Graphical Interfaces

Step 1: Select a product to be deleted from the UI list/search panel. It is possible to select set of products also using the “select all” button (bulk deletion);

Step 2: Delete products clicking on the trashbin icon in the lower side of the panel itself

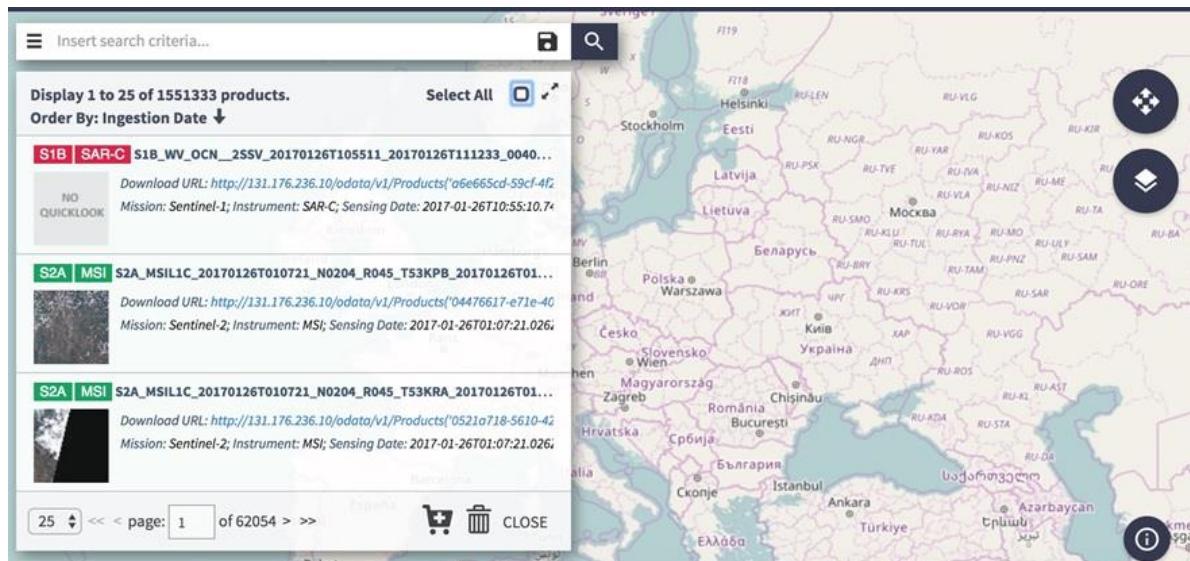


Figure 16 Deletion via OData API

A prompt message is showed up to allow inserting a deletion cause:

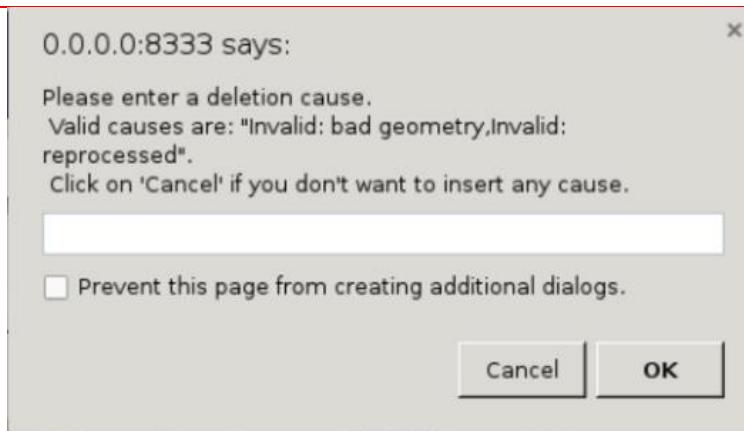


Figure 17 Deletion cause panel

The set of deletion causes shown in the panel can be configured as explained in Section 9.1 (AJS GUI appconfig.json).

If an operator do no choose one of the configured deletion causes, the DHuS set the deletion cause value as "Invalid".

7.3.1.2 Via OData

The Data Hub provides to administrators the possibility to delete products via OData.

All deleted (and evicted) products are accessible to all kind of users at:

[http://\[DHUS_IP_ADDRESS\]/odata/v1/DeletedProducts](http://[DHUS_IP_ADDRESS]/odata/v1/DeletedProducts)

How to delete a product via Odata?

- With no cause:

```
curl -u user:password "http://localhost:8080/odata/v1/Products('91890b5e-049a-47c9-97c1-cb9542a694b5')" -X DELETE
```

In this case, the DeletionCause is « null »

- With a generic cause:

```
curl -u user:password "http://localhost:8080/odata/v1/Products('91890b5e-049a-47c9-97c1-cb9542a694b5')?cause=Size%20Needed" -X DELETE
```

- Determining it as "Invalid" :

```
curl -u user:password "http://localhost:8080/odata/v1/Products('91890b5e-049a-47c9-97c1-cb9542a694b5')?cause=Invalid" -X DELETE
```

It is strongly recommended to identify invalid products by starting their deletionCause with "Invalid". Such practice will make easier to set up a regular expression in future (for example: "Invalid, bad geometry" or "Invalid, reprocessed").

Please note that evicted products have their DeletionCause automatically set to "Automatic Eviction".

7.4 Synchronizer

7.4.1 How to create a new Synchronizer

7.4.1.1 Preconditions

The FE/BE instances shall be configured as follows:

- 1) BE: DHuS instance with no quota limitation and having a user with the 'archive management' function enabled.
- 2) FE: having the synchronization functionality enabled, meaning that the dhus.xml of the FE shall contain the following setting:

```
<executor enabled="true" batchModeEnabled="false">
```

In case a proxy has been set up between BE and FE, please change the network properties in order to include the BE in the list of IP that can access to BE without going through the proxy.

7.4.1.2 Via Graphical Interfaces

The OData Synchronizers panel is the User interface allowing the creation and update of synchronizers among two or more DHuS instances.

The Administrator shall:

- **Step 1:** Log in as Root in the Front end DHuS instance and select the tab "user profile"
- **Step 2:** Select the panel "OData synchronizers"

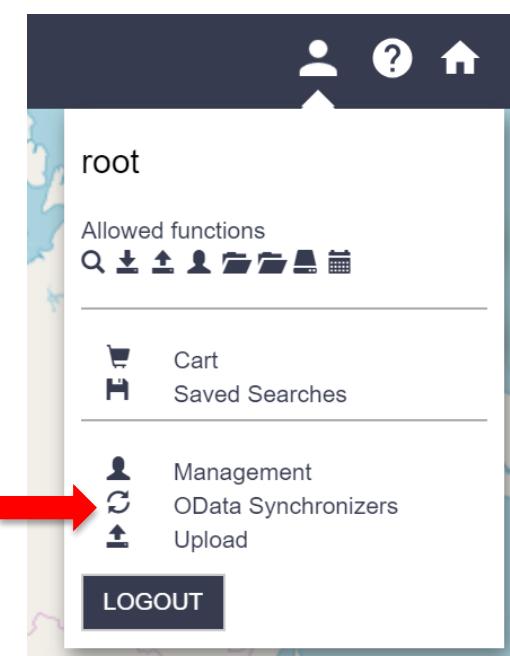


Figure 18 OData Synchronizer Access

- **Step 3:** Click on “Create synchronizer”

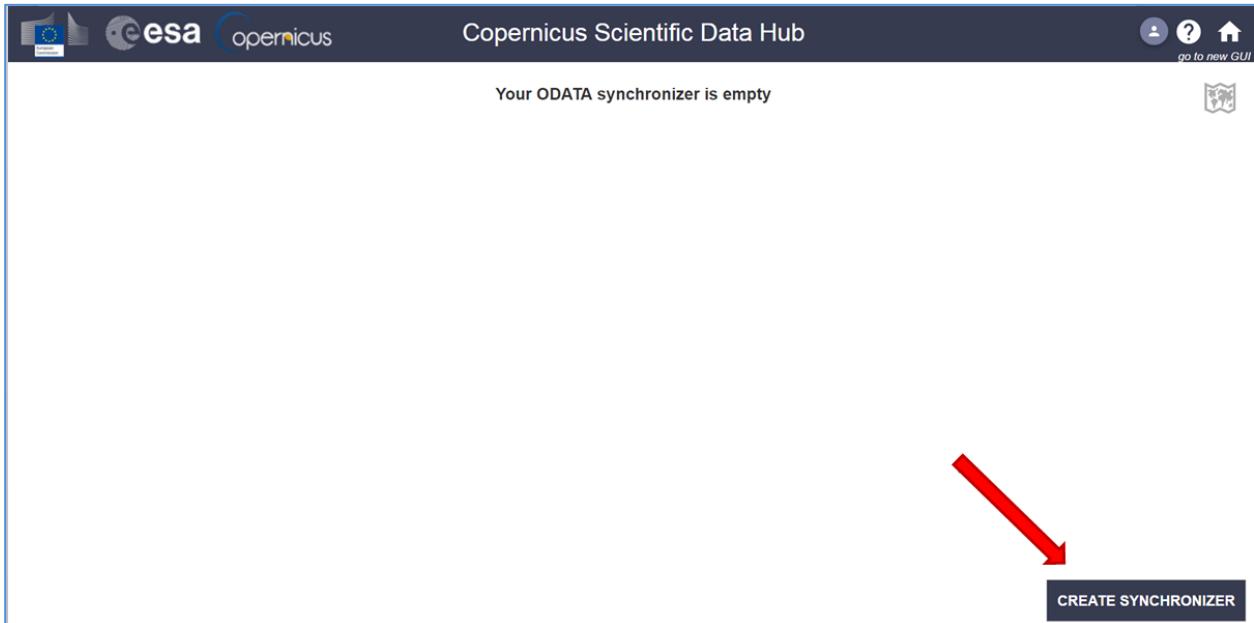


Figure 19 OData Synchronizer Panel (AJS)

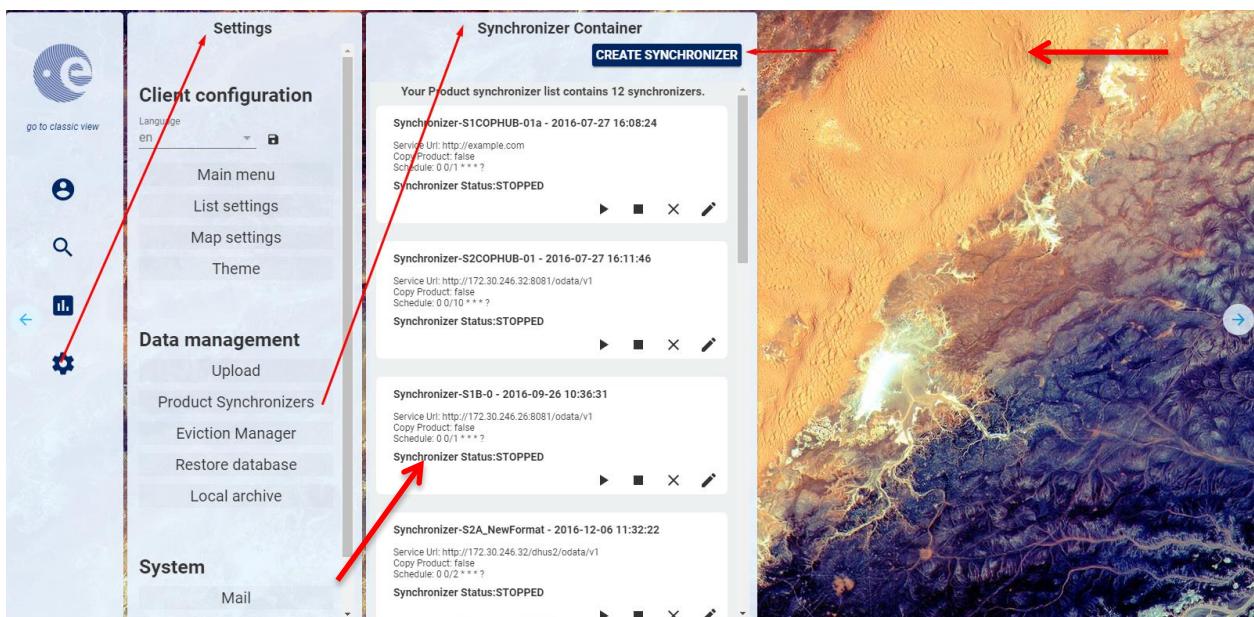


Figure 20 OData Synchronizer Panel (OWC)

- **Step 4:** Fill the records in “Synchronizer properties” in order to configure its parameters (see Section 5.4):

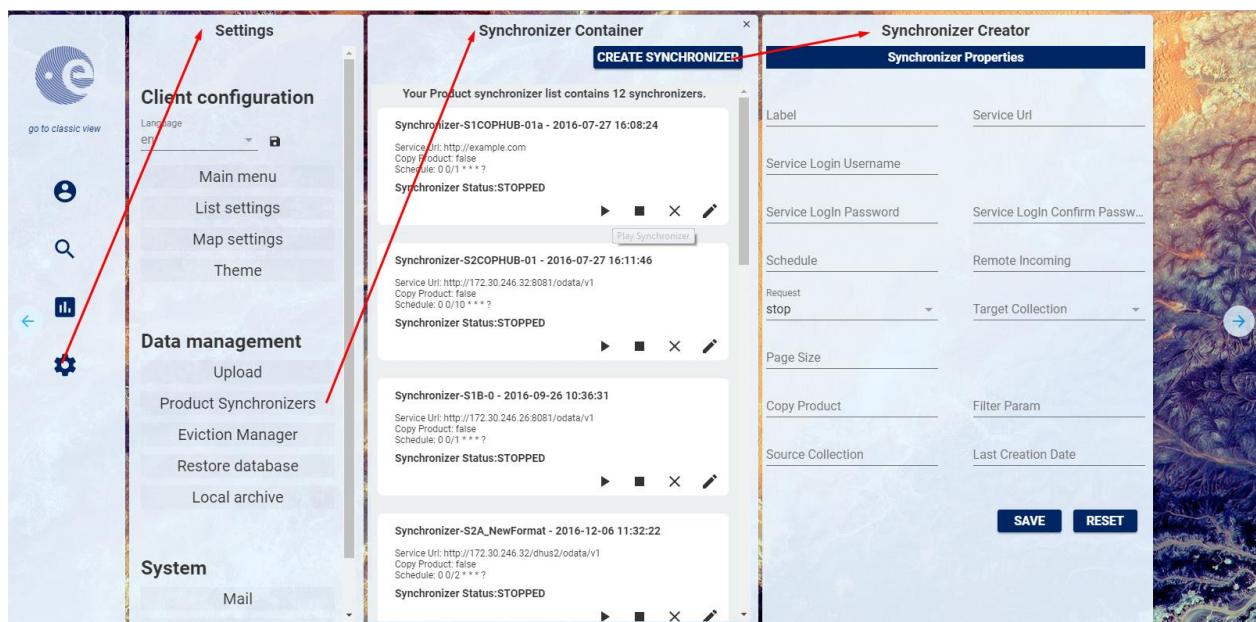
Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

Synchronizer Properties

Label	Service Url
Service Login Username	
Service Login Password	Service Login Confirm Password
Schedule	Page size
Request Stop	Target Collection
Copy product	Remote Incoming
Source Collection	Last Creation Date
Filter Param	Geo Filter

Action Buttons:  

Figure 21 Create Synchronizer Panel (AJS)



The screenshot shows the Client configuration interface with several panels open:

- Client configuration** panel on the left: Shows settings like Language (en), Main menu, List settings, Map settings, Theme, Data management (Upload, Product Synchronizers, Eviction Manager, Restore database, Local archive), System (Mail), and a gear icon for Settings.
- Synchronizer Container** panel in the center: Displays a list of synchronizers:
 - Synchronizer-S1COPHUB-01a - 2016-07-27 16:08:24: Service Url: http://example.com, Copy Product: false, Schedule: 0 0/1 * * ? Status: STOPPED
 - Synchronizer-S2COPHUB-01 - 2016-07-27 16:11:46: Service Url: http://172.30.246.32.8081/odata/v1, Copy Product: false, Schedule: 0 0/10 * * * Status: STOPPED
 - Synchronizer-S1B-0 - 2016-09-26 10:36:31: Service Url: http://172.30.246.26.8081/odata/v1, Copy Product: false, Schedule: 0 0/1 * * ? Status: STOPPED
 - Synchronizer-S2A_NewFormat - 2016-12-06 11:32:22: Service Url: http://172.30.246.32.dhus2/odata/v1, Copy Product: false, Schedule: 0 0/2 * * ? Status: STOPPED
- Synchronizer Creator** panel on the right: A form for creating a new synchronizer with fields: Label, Service Url, Service Login Username, Service Login Password, Service Login Confirm Password, Schedule, Request (stop), Target Collection, Page Size, Copy Product, Filter Param, Source Collection, Last Creation Date, and two buttons: SAVE and RESET.

Figure 22 Create Synchronizer Panel (OWC)

- **Step 5:** Click on the button with the “floppy disk” shape.

7.4.1.3 Via script

- **Step 1:** Log in the machine where the Front End DHuS is installed.

- **Step 2:** On the DHuS acting as FE, set the environment variables (export <variable>=value) that configure the DHuS:
 - DHOST DHuS Host address eg: localhost:8080, scihub.copernicus.eu/dhus/
 - DLOGIN DHuS login username to connect to the \$DHOST DHuS
 - DPASS DHuS login password to connect to the \$DHOST DHuS
- **Step 3:** Create a working folder containing the scripts available at:
https://github.com/SentinelDataHub/Scripts/tree/master/dhus_sync_script
- **Step 4:** To create the synchronizer use the command:

```
./createSynchronizer           <-D_SCHEDULE=cron_expression>           <-
D_SERVICEURL=URL_to_remote_DHuS_to_sync>   <-D_SERVICELOGIN=account>       <-
D_SERVICEPASSWORD=password>      [-D_LABEL=my_sync]      [-D_PAGESIZE=X]      [-
D_REQUEST=start|stop]          [-D_COPYPRODUCT=true|false]      [-
D_FILTERPARAM=filter_expression]  [-D_SOURCECOLLECTION=resource/path]      [-
D_LASTCREATIONDATE=date]      [-D_GEOFILTER=geofilter]
```

Where parameters between angle brackets (chevrons) are mandatory, parameters between brackets ([]]) are optional. For details about the parameters meaning, please refer to Section 5.4.

Please note that there are options to nullify nullable properties: -D_LABEL_NULL, -D_FILTERPARAM_NULL, -D_SOURCECOLLECTION_NULL, -D_GEOFILTER_NULL

7.4.2 How to update a Synchronizer

7.4.2.1 Via Graphical Interfaces

The Administrator shall:

- **Step 1:** Log in as Root in the Front end DHuS instance and select the tab “user profile”;
- **Step 2:** Select the panel “OData synchronizers”, stop the synchronizer clicking on the “squared” icon, then click on the “pencil” next to the synchronizer to be updated

Figure 23 Updating a synchronizer



- **Step 1:** Edit the records to be updated
- **Step 2:** Click on the button with the “floppy disk” shape and re-start the synchronizer

7.4.3 How to delete a Synchronizer

7.4.3.1 Via Graphical Interfaces

The Administrator shall:

- **Step 1:** Log in as Root in the Front end DHuS instance and select the tab “user profile”;
- **Step 2:** Select the panel “OData synchronizers” and then click on the “X” shaped button next to the synchronizer to be updated.

Next to an existing synchronizer tab, there are also buttons for starting and stopping the item. The “play” button is to start the synchronizer; the “square” button is to stop it.

7.5 User Synchronizer

7.5.1 How to create a User Synchronizer

- **Step 1:** Log in the machine where the FE DHuS is installed. This DHuS instance shall be filled with users registered in the BE.
- **Step 2:** On the DHuS acting as FE, set the environment variables (export <variable>=value) that configure the DHuS:
 - DHOST DHuS Host address eg: localhost:8080, scihub.copernicus.eu/dhus/
 - DLOGIN DHuS login username to connect to the \$DHOST DHuS
 - DPASS DHuS login password to connect to the \$DHOST DHuS
- **Step 3:** Install M4 Macro Language package running
 - apt-get install m4 (on Debian platforms)
 - yum install m4 (on RedHat/CentOS platforms)
- **Step 4:** Create a working folder containing the scripts available here
https://github.com/SentinelDataHub/Scripts/tree/master/dhus_sync_script
- **Step 5:** To create the user synchronizer use the command:

```
./createSynchronizer <-D_SCHEDULE=...> <-D_REQUEST=start|stop> <-  
D_SERVICEURL=...> [-D_LABEL=...] [-D_SERVICELOGIN=...] [-  
D_SERVICEPASSWORD=...]
```

Example:

```
. /createSynchronizer -D_SCHEDULE="0 */*3 * * ?" -D_REQUEST=stop \  
-D_SERVICEURL="http://192.168.0.105:8080/odata/v1" -D_LABEL=my_user_syncer \  
-D_SERVICELOGIN=root -D_SERVICEPASSWORD=a
```

Where:

- **Schedule**= how often the synchronizer shall be running. This shall be configured according to the crontab syntax
- **Request**= "start" or "stop"
- **Label**= Name of the synchronizer
- **Service URL**= [https://\[Back-End_DHuS_address\]/odata/v1](https://[Back-End_DHuS_address]/odata/v1)
- **Service Login Username**= User name of a user registered in the Back end with user manager role.
- **Service Login Password**= password of the user in the previous step
- **Page size**= number of users synchronized at each synchronizer run
- **Cursor**= if the cursors is set as "i" (where i is such that 0<i<total amount of users) the synchronizer will import in the FE all the users of the BE from the i-th.
- **Force**=true/false. This parameter defines if this UserSynchronizer should overwrite namesakes. If Forced is set to true, namesakes will be overwritten thus after a complete synchronisation of all users from the data source, there will be no namesakes on the synchronising DHuS.

7.5.2 How to start/stop a User Synchronizer

To start an existing synchronizer execute the following command:

```
./updateSynchronizer <id> -D_REQUEST=start|stop
```

Example:

```
./updateSynchronizer 0 -D_REQUEST=start
```

7.5.3 How to update a User Synchronizer

To update an existing synchronizer execute the following command:

```
./updateSynchronizer <id> [-D_SCHEDULE=...] [-D_REQUEST=start|stop] \ 
[-D_SERVICEURL=...] [-D_LABEL=...] [-D_SERVICELOGIN=...] [-D_SERVICEPASSWORD=...]
```

Example:

```
./updateSynchronizer 0 -D_SCHEDULE='0 24 * * ?'
```

7.5.4 How to delete a User Synchronizer

To delete an existing user synchronizer launch the following command:

```
./deleteSynchronizer <id>
```

7.6 Management Panels

The section describes the management panels implemented in DHuS and accessible via AJS UI and OWC UI.

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

For the AJS UI, the first step is to access the “Management” area by means of the button at the right side of the Home web page.

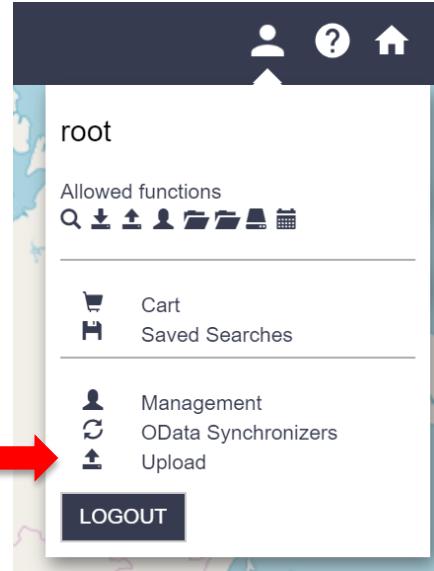


Figure 24 Management area (AJS)

The following four subpanels are present:

- Users
- Collections (not available in OWC)
- System
- Eviction

The screenshot shows the Copernicus Scientific Data Hub Management Subpanel. At the top, there are logos for the European Union, ESA, and Copernicus. The title "Copernicus Scientific Data Hub" is visible. Below the title, there are tabs for "USERS", "COLLECTIONS", "SYSTEM", and "EVICTION". The "USERS" tab is selected. A message "Display 1 to 2 of 2 users." is shown above a table. The table has columns for "Search:" and two user entries. The first user entry is "Display 1 to 2 of 2 users.".

Figure 25 Management Subpanel (AJS)

In OWC, the subpanels are located in the “Settings” Tab:

- Users
- System
- Eviction
- Main Menu settings

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

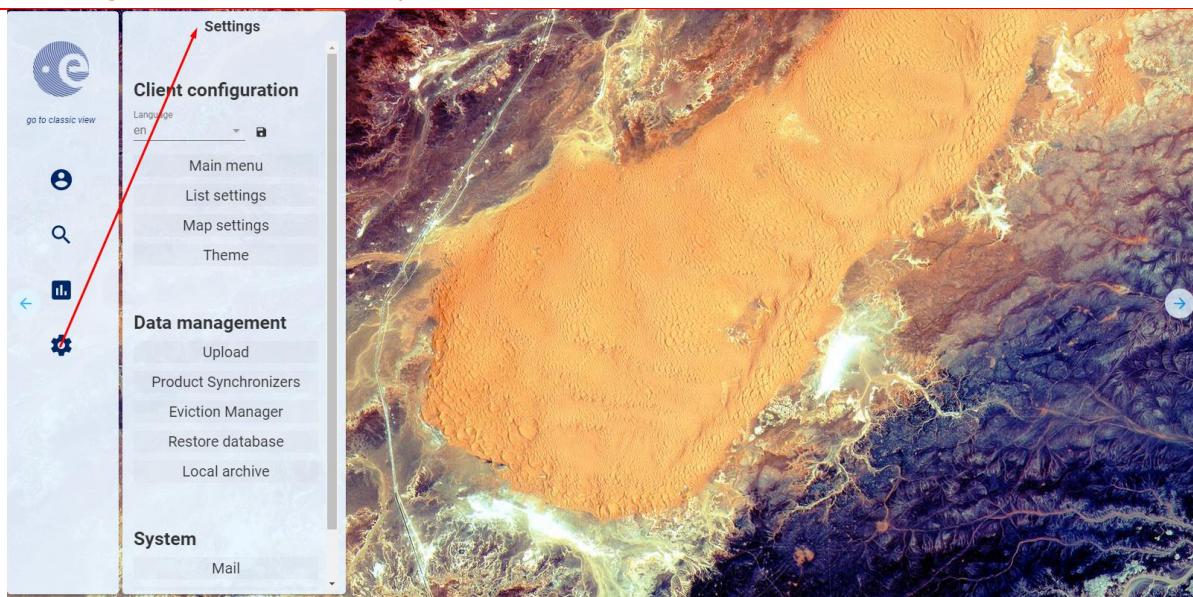


Figure 26 Management Subpanel (OWC)

7.7 User Management

7.7.1 How to manage users via GUI

Access the administrator management panel as described in Section 7.6 and go to the Users panel.

The screenshot shows a list of users from the Copernicus Scientific Data Hub. The top navigation bar includes links for USERS, COLLECTIONS, DATA RIGHT ACCESS, SYSTEM, and EVICTION. The main content area displays a table with the following columns: Name, E-mail, Firstname, Lastname, Country, and isLocked. There are 9853 users listed, with the first few entries shown below:

Name	E-mail	Firstname	Lastname	Country	isLocked
3452emasos	E-mail: odontsov@mines.edu	Oleg	Dontsov	United States	false
woldai	E-mail: t.woldai07@gmail.com	Tsehale	Woldai	Netherlands	false
191hardbeat	E-mail: moselewski@outlook.de	Philipp	Moselewski	Germany	false
1054064262	E-mail: 1054064262@qq.com	zhi	Li	China	false
1102159420@qq.com	E-mail: 1102159420@qq.com	xiaohong	ma	China	false
111222	E-mail: chamber20@itacali.co.uk	Michael	Dawson	United Kingdom	false
1113530041	E-mail: 1113530041@qq.com	qwe	qwe	China	false
1156129056	E-mail: 1156129056@qq.com	hua	liu	China	false
12053226	E-mail: 792167567@qq.com	wenli	qiao	China	false
12345	E-mail: dehnavi.kntu@gmail.com	Sahar	dehnavi	Iran, Islamic Republic of	false
1290516461	E-mail: 1290516461@qq.com	xiaojun	qiao	China	false
137870911	E-mail: 137870911@qq.com	jinglong	dong	China	false

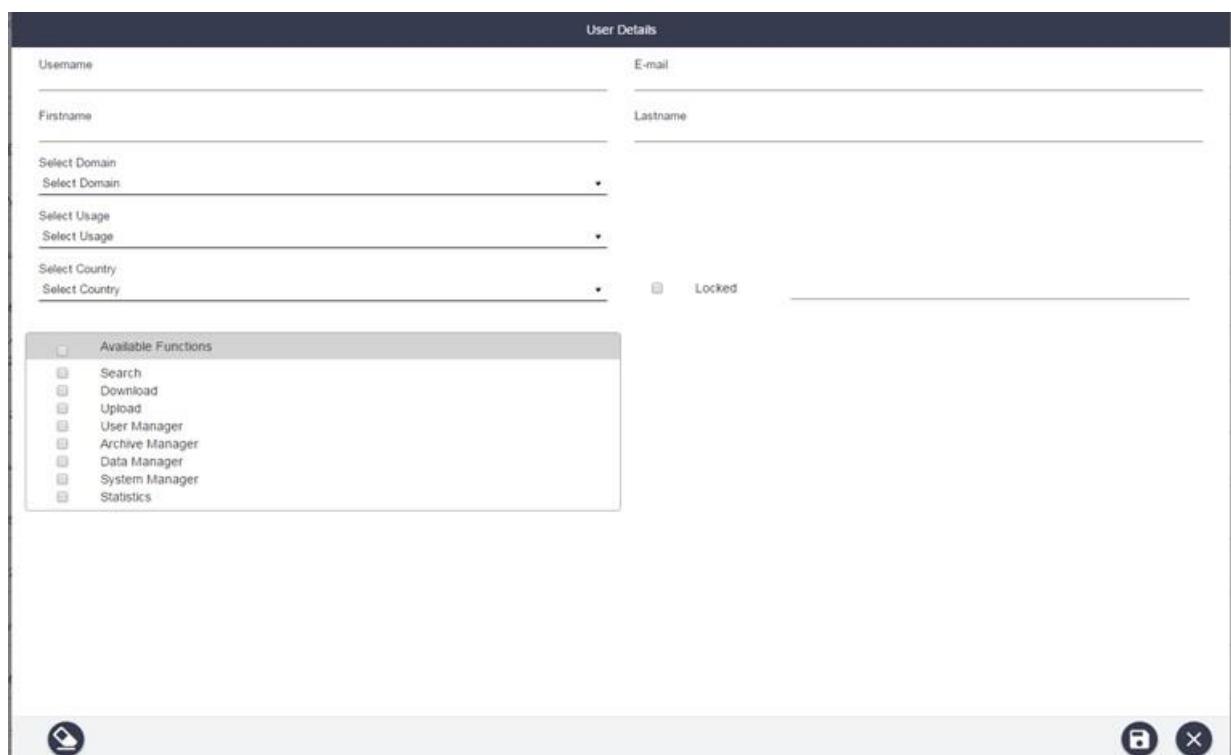
At the bottom, there are pagination controls: 'Users per page: 25', 'Page #: 1 of 395', and a 'CREATE USER' button.

Figure 27 DHuS User Management Panel (AJS)

7.7.1.1 Create a new user

The Administrator shall:

- **Step 1:** Access the DHuS page;
- **Step 2:** Perform the login;
- **Step 3:** Select the Management Panel and then select the Users management panel;
- **Step 4:** Click on the “Create” button in the lower part of the User management page, which will enable



User Details

Username	E-mail
Firstname	Lastname
Select Domain Select Domain	
Select Usage Select Usage	
Select Country Select Country	<input checked="" type="checkbox"/> Locked

Available Functions

- Search
- Download
- Upload
- User Manager
- Archive Manager
- Data Manager
- System Manager
- Statistics

Figure 28 User Creation Form (AJS)

the form here below.

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

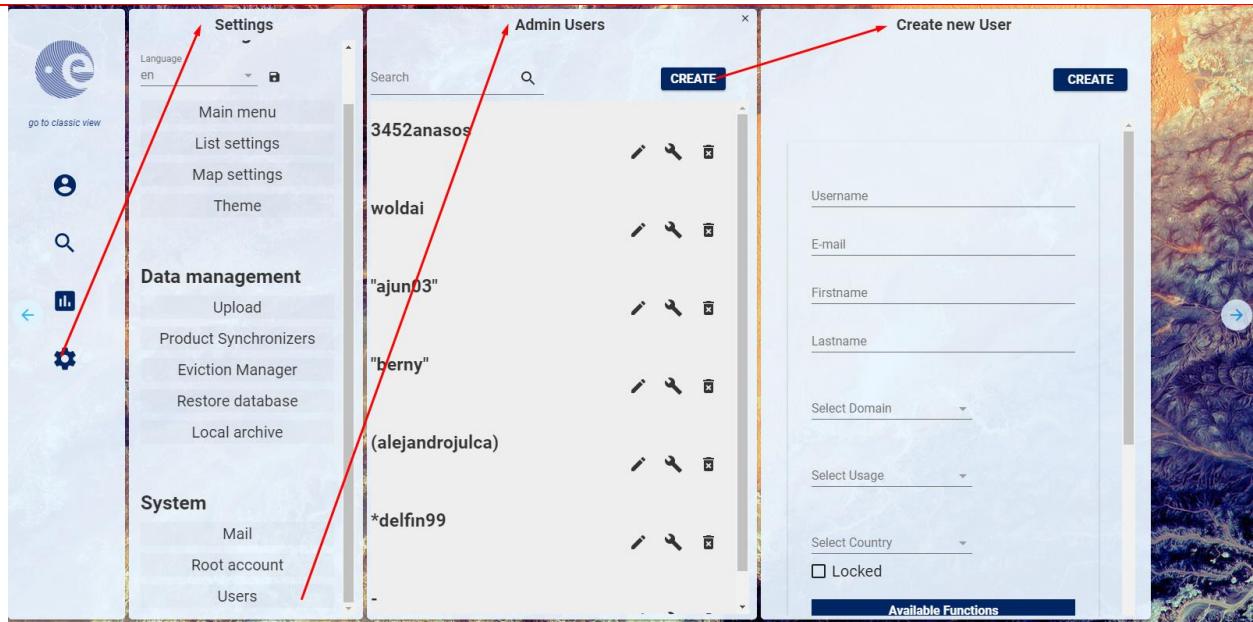


Figure 29 User Creation form (OWC)

- **Step 1:** Fill in the new user creation form (note that the fields marked with an asterisk are mandatory) and click on the functions that the user shall be able to use
- **Step 2:** Click on the "save" button to complete;
- **Step 3:** Then the email notification service will send an e-mail to the user with his profile information (login, first name, last name, available services...) including a generated password.

The administrator has the possibility to modify user's authorization settings and information. To modify whatever authorization setting or user information, the Administrator, before executing the following "how to" procedures, has to:

- **Step 1:** Access the DHuS page
- **Step 2:** Perform the login
- **Step 3:** Select the Management Panel
- **Step 4:** Select the Users Management Panel
- **Step 5:** Select the name of the user in the users list on the left side of the user management panel

7.7.1.2 Lock selected user

Click on the "locked" checkbox under the Registration form in the right side of the panel. The administrator shall also indicate the reason of this locking process in the box on the right

- **Step 1:** Click on the "save" button to complete;
- **Step 2:** Then the email notification service will send an e-mail to the user with his profile information (login, first name, last name, available services...) including locking notification and its relative reason, if it has been indicated.

Warning: the locking restriction and its relative reason are registered in the OData "Restriction" EntityType, under the corresponding "User". Please refer to [RD 6 RD 6] for more details.

7.7.1.3 Delete selected user

Click on the "Delete" button to delete



Figure 30 Update, delete users, modify user password on AJS

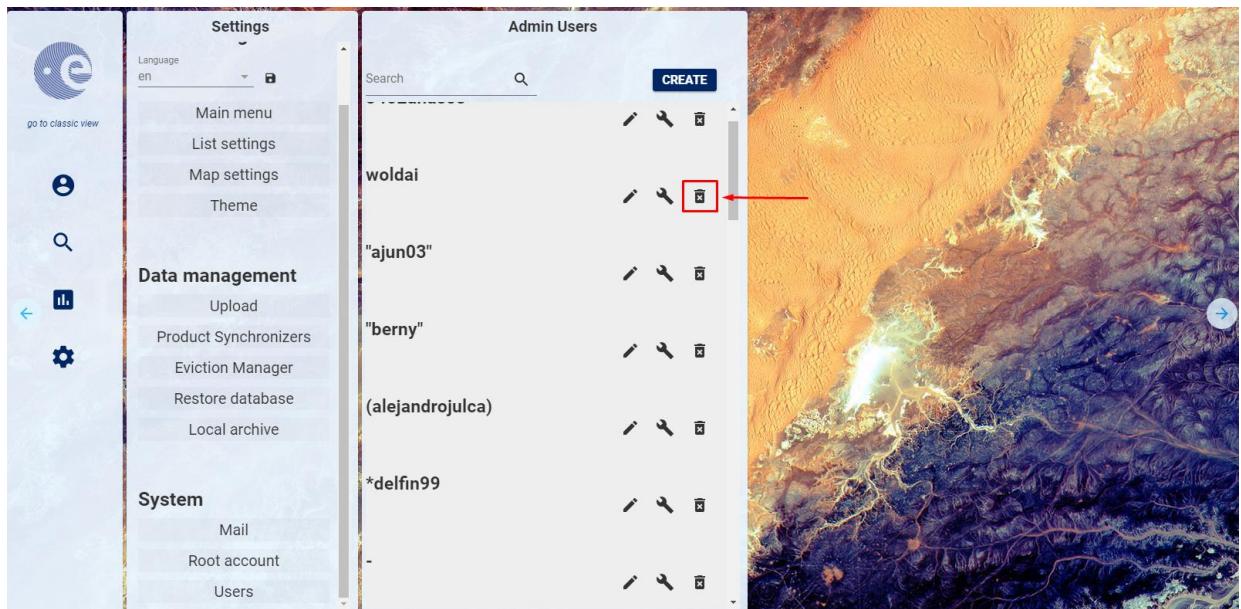


Figure 31 Delete users (OWC)

The email notification service will send an e-mail to the deleted user with the communication of the deletion process.

7.7.1.4 Change user password

Click on the "edit user password" checkbox and the following panel will appear:

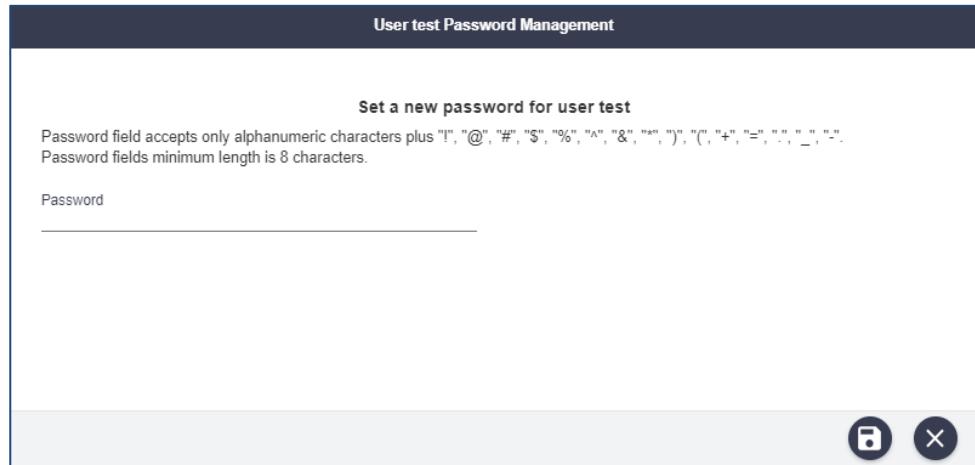


Figure 32 Update, edit user password and delete users AJS

Including the new password and, then, clicking on the save button, the user password will be updated.

7.7.2 How to manage users via API

7.7.2.1 JSON ODATA representation for user configuration

Here after a json file showing the needed parameters to manage users, as an example:

```
{  
  "Username": "User_Name",  
  "Email": "E-mail_address",  
  "FirstName": "User_FirstName",  
  "LastName": "User_LastName",  
  "Country": "Italy",  
  "Phone": null,  
  "Address": null,  
  "Domain": "Climate",  
  "SubDomain": null,  
  "Usage": "Commercial",  
  "SubUsage": null,  
  "Password": "password",  
  "SystemRoles": {"__deferred": {"uri": "SystemRoles('DOWNLOAD')"}},  
  "SystemRoles": {"__deferred": {"uri": "SystemRoles('SEARCH')"}},
```

```
"SystemRoles": {"__deferred": {"uri": "SystemRoles('ARCHIVE_MANAGER')"}}}
```

7.7.2.2 Create a new user

```
curl -H "Content-Type: application/json" -X POST -d "@/path/to/user.json" -u ${DLOGIN}:${DPASS} "http://${DHOST}/odata/v1/Users"
```

where the "user.json" is the name of the JSON file containing the proper configuration for the creation of the user, detailed in Section 7.7.2.1.

7.7.2.3 Lock selected user

```
curl -H "Content-Type: application/json" -X POST -u ${DLOGIN}:${DPASS} "http://${DHOST}/odata/v1/LockUser?username='User_Name'&reason='This%20is%20a%20reason%20for%20lock.'
```

Please note that "%20" characters should be used instead of white spaces in the description of the reason for the lock action.

7.7.2.4 Unlock selected user

```
curl -H "Content-Type: application/json" -X POST -u ${DLOGIN}:${DPASS} "http://${DHOST}/odata/v1/UnlockUser?username='User Name'&restriction=RESTRICTION_UUID"
```

Please note that the RESTRICTION_UUID to be set in the restriction field can be retrieved via the following OData query for locked users:

```
http://${DHOST}/odata/v1/Users('User_Name')/Restrictions
```

7.7.2.5 Update user profile

```
curl -H "Content-Type: application/json" -X PUT --data "@/path/to/user.json" -u ${DLOGIN}:${DPASS} "http://${DHOST}/odata/v1/Users('User_Name')"
```

where the "user.json" is the name of the JSON file containing the proper configuration for the creation of the user, detailed in Section 7.7.2.1. All the fields in the json file can be updated.

7.7.2.6 Delete selected user

```
curl -H "Content-Type: application/json" -X DELETE -u ${DLOGIN}:${DPASS} "http://${DHOST}/odata/v1/Users('User_Name')"
```

7.8 Collection Management

The Collection management panel lists a set of products to be attached to the collection. The selection of collections is possible by browsing the collection hierarchy on the left. To access the collection management panel, the Administrator has to click on the “collections” link, sited in the upper left side of the management panel. The collection management panel here below will open. It contains the list of collections on the left and the list of archived products on the right.

Figure 33 Collection Management Panel (AJS)

The collection management can be performed only after the following preliminary actions:

- **Step 1:** Access the DHuS page;
- **Step 2:** Perform the login;
- **Step 3:** Go to the collection management panel.

7.8.1 How to create a new collection

- **Step 1:** Click on the “create” button in the collection management panel;
- **Step 2:** Create collection;
- **Step 3:** Insert the collection information in the upper right side of the panel (the name of collection is mandatory), (optional) select (by clicking on the associated check box) the products to be added to the collection;
- **Step 4:** Click on the “save” button to register the new collection or click on the “cancel” button to abort the ‘creation of collection’ procedure.

7.8.2 How to delete a collection

- **Step 1:** Click on the collection to delete;
- **Step 2:** Click on the “delete” button.

Note that the collection management page includes a searching box. It is useful to know if a product is collected somewhere.

7.9 System Management

Access the administrator management panel as described in Section 7.6 and go to the System management panel.

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

Copernicus Scientific Data Hub

Mail Configuration

SMTP Server	Port	TLS
131.178.235.251	25	<input checked="" type="checkbox"/>

Username	Password
solhub@copernicus.esa.int	*****

Expeditor Mail	Expeditor Name
solhub@copernicus.esa.int	Sentinels Scientific Data Hub Support

Reply to	solhub support@copernicus.esa.int
----------	-----------------------------------

Mail user when:	<input checked="" type="checkbox"/> creating account	<input checked="" type="checkbox"/> administrator updates his account	<input checked="" type="checkbox"/> deleting account
-----------------	--	---	--

Support

Mail	Name
solhub support@copernicus.esa.int	Sentinels Scientific Data Hub - Support Team

Registration mail	solhub.support@copernicus.esa.int
-------------------	-----------------------------------

Root configuration

Old Password	New Password
Confirm Password	

Restore Database

Choose database dump that will be restored

Figure 34 System management sections (AJS)

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

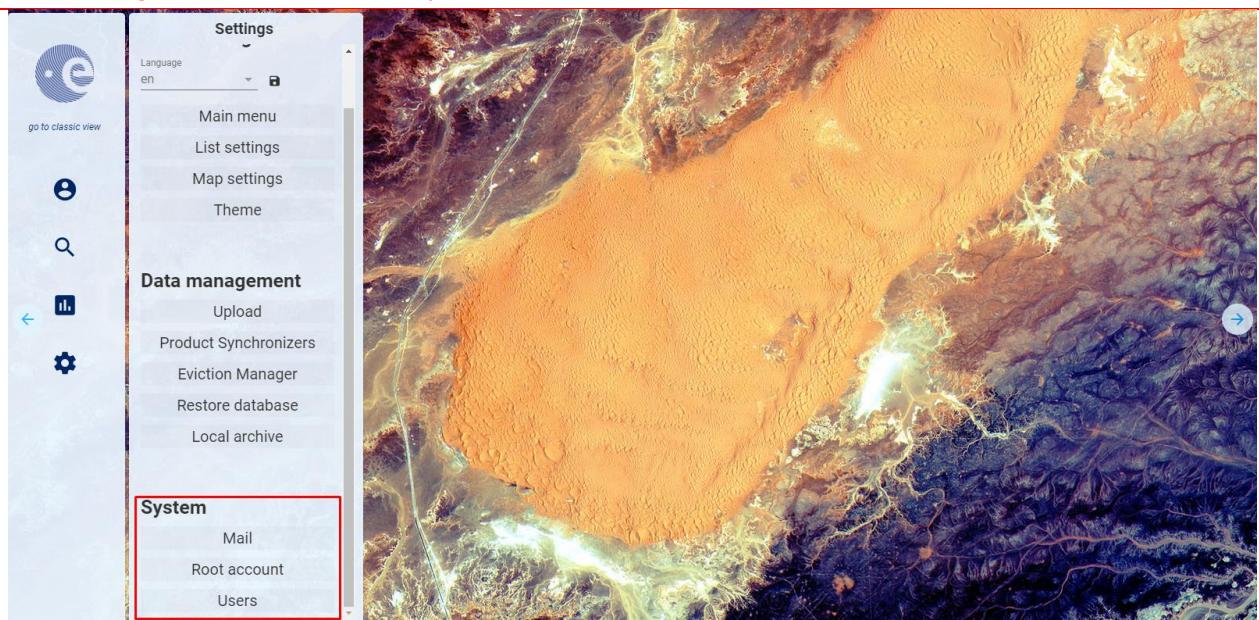
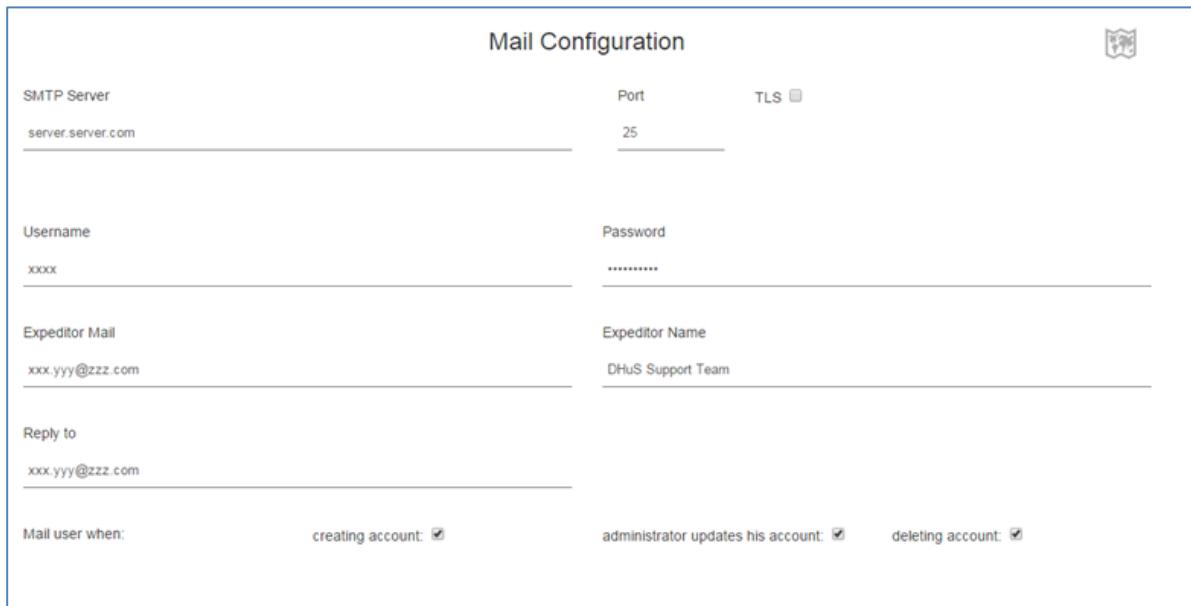


Figure 35 System management sections (OWC)

The following Sections detail the possible actions from this panel.

7.9.1 How to set mail configuration

In this form it is possible to configure the SMTP server address, the username, password and e-mail account details to send communications to the users.



The form is titled "Mail Configuration". It has fields for SMTP Server (server.server.com), Port (25), and TLS (checkbox). There are also fields for Username (xxxx), Password (*****), Expeditor Mail (xxx.yyy@zzz.com), Expeditor Name (DHuS Support Team), Reply to (xxx.yyy@zzz.com), and Mail user when: (checkboxes for creating account, administrator updates his account, and deleting account).

Figure 36 Mail configuration management Panel (AJS)

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

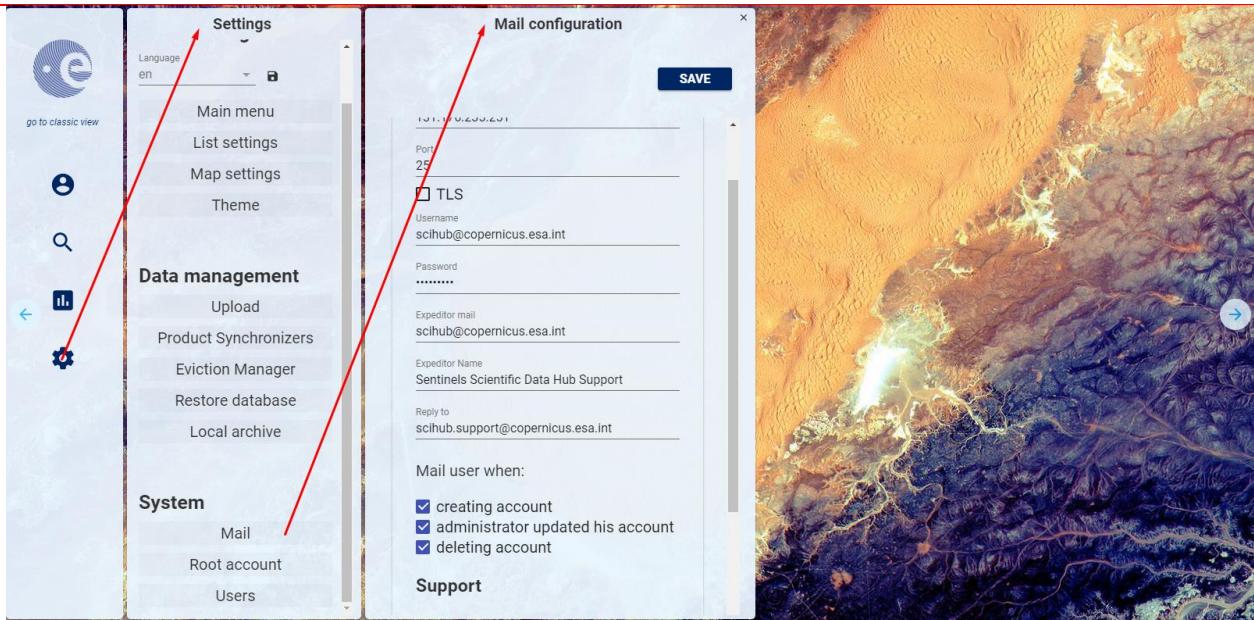
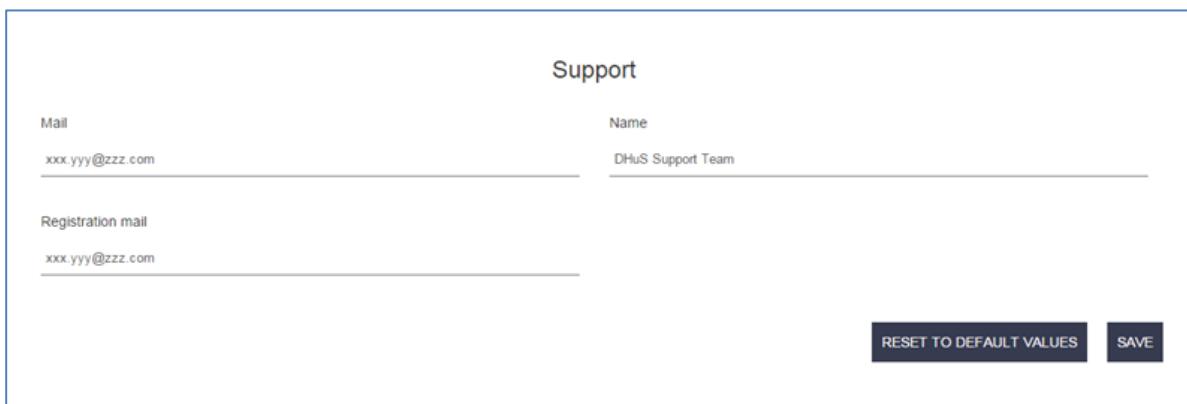


Figure 37 Mail configuration management Panel (OWC)

7.9.2 How to set Support information

For any support information it is possible to contact the DHuS Support Team sending an e-mail to eosupport@copernicus.esa.int. This is not present in OWC.



The screenshot shows the AJS support configuration panel. It has two main sections: 'Mail' and 'Name'. In the 'Mail' section, there are fields for 'Mail' (xxx.yyy@zzz.com) and 'Registration mail' (xxx.yyy@zzz.com). In the 'Name' section, there is a field for 'Name' (DHuS Support Team). At the bottom right are 'RESET TO DEFAULT VALUES' and 'SAVE' buttons.

Figure 38 Support configuration management panel (AJS)

7.9.3 How to change root password

From this panel it is possible to change the administrator password. To do so, insert the old password, the new one and then confirm the new password.

Root configuration

Old Password	New Password
<input type="password"/>	<input type="password"/>
Confirm Password	
<input type="password"/>	
SAVE	

Figure 39 Root configuration management panel (AJS)

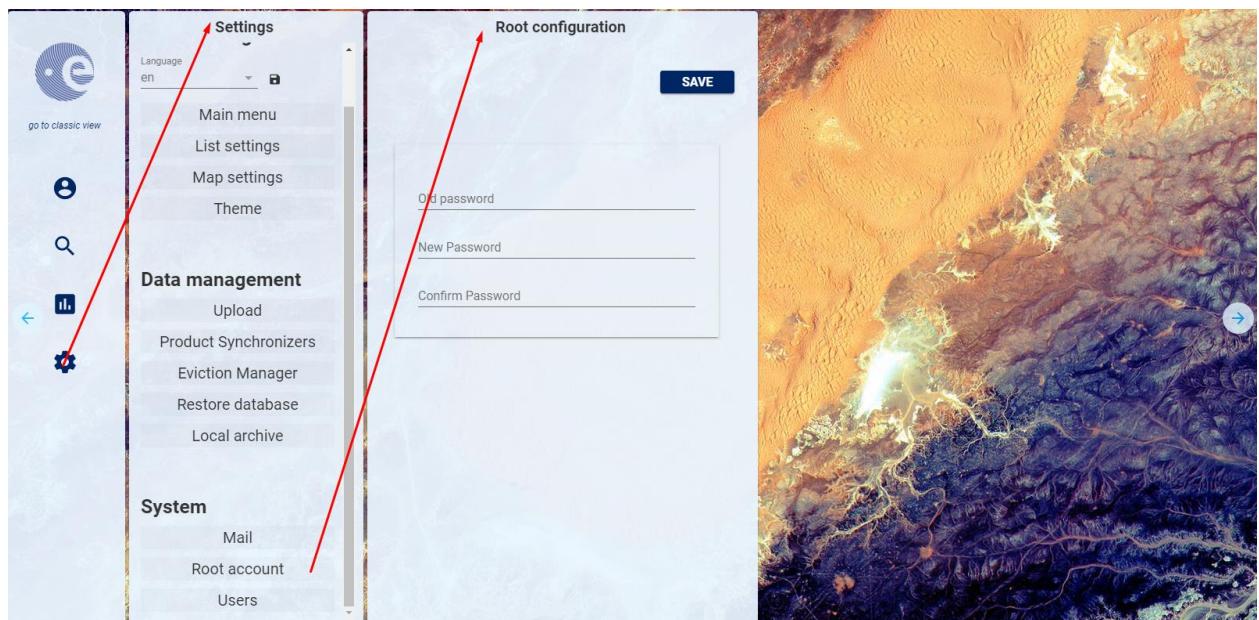


Figure 40 Root configuration management panel (OWC)

7.9.4 How to restore database

In the `dhus.xml` file it is possible to configure DHuS so that it performs a periodical dump of the database. From this panel it is possible to restore the database dump.

Restore Database

Choose database dump that will be restored	<input type="button" value="RESTORE"/>
<input type="button" value="..."/>	

Figure 41 Restore Database panel (AJS)

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

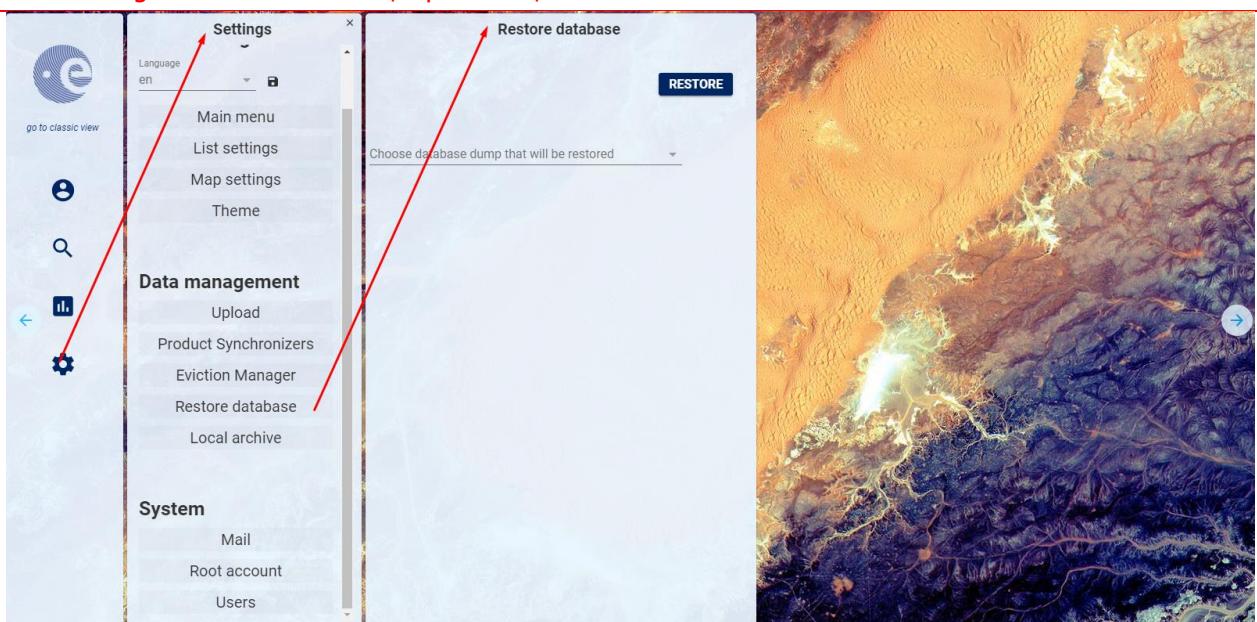


Figure 42 Restore Database panel (OWC)

To do so, perform the following steps:

- **Step 1:** Click on the drop-down menu in the “Restore Database” section: the list of available dumps will be displayed through a list of dates (date during which the dumps have been performed).
- **Step 2:** Select the desired date and then click on “RESTORE”. DHuS will automatically stop and restart. Once DHuS will be up again, it will contain just the data inserted before the selected dump date.

7.10 Eviction Management

7.10.1 How to manage a Customizable Eviction via AJS GUI

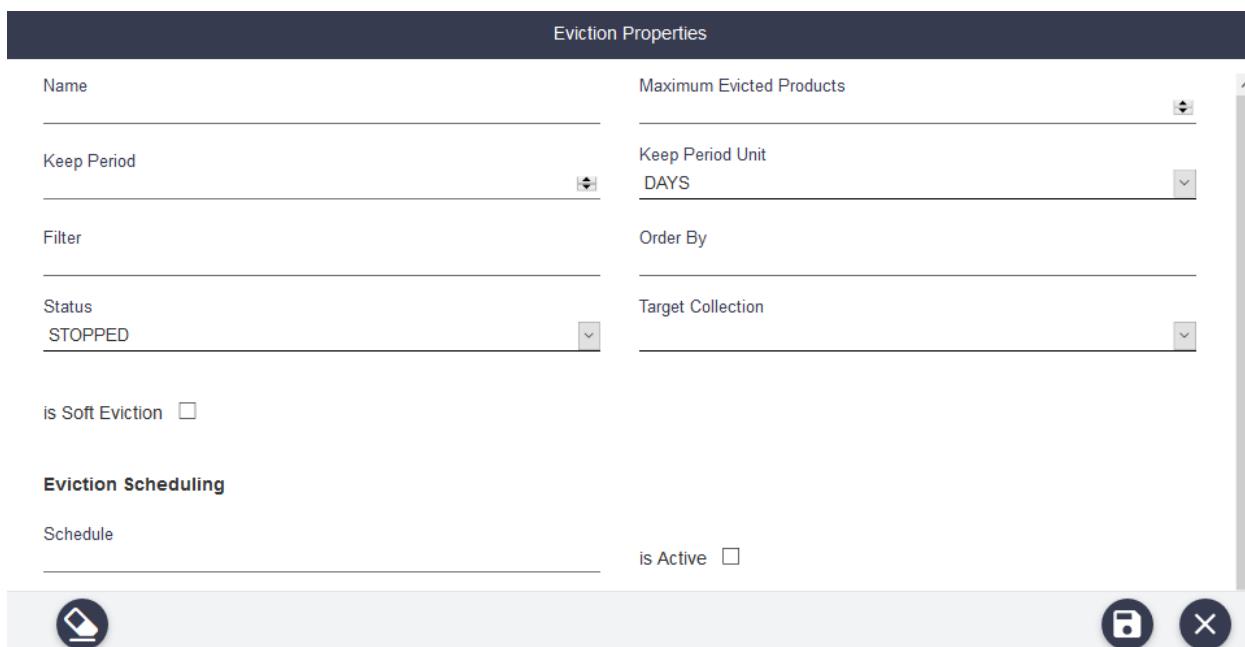
7.10.1.1 Creation

In order to activate a Customizable Eviction via GUI perform the following steps:

- **Step 1:** Access the DHuS page;
- **Step 2:** Perform the login;
- **Step 3:** Access the administrator management panel as described in Section 7.6;
- **Step 4:** Select the Eviction Management and click on “CREATE EVICTION” button;
- **Step 5:** Configure the following Eviction Properties in the pop-up window, as per Figure 43 Eviction Management panel (GUI)Figure 43 (please refer to Section 5.9.1 for further details):
 - *Name* – insert a unique eviction name
 - *Maximum Evicted Products* – insert the maximum number of product to be evicted at each run

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

- *Keep Period* – insert the number representing the time after which a product can be evicted
- *Keep Period Unit* – insert the time unit via drop down menu
- *Filter* – filtering clause to be set by using the same OData syntax used on the Products entity set, restricts which products can be evicted
- *Order By* – ordering clause to be set by using the same OData syntax used on the Products entity set, defines the order in which products will be evicted
- *Target Collection* – selectable via drop down menu if it has been created previously as in Section 7.8.1.
- *Soft Eviction* – to be ticked or not to perform a Soft or a Hard eviction, respectively.
- **Eviction Scheduling**
 - *Schedule* – shall be configured according to the crontab syntax
 - *Is active* – to be ticked to activate the Eviction with the set cron scheduling



The screenshot shows the 'Eviction Properties' configuration screen. It includes fields for Name, Keep Period, Filter, Status, Maximum Evicted Products, Keep Period Unit, Order By, Target Collection, and Eviction Scheduling. The 'Eviction Scheduling' section contains fields for Schedule and Is Active, along with save and cancel buttons.

Eviction Properties	
Name	Maximum Evicted Products
Keep Period	Keep Period Unit DAYS
Filter	Order By
Status STOPPED	Target Collection
is Soft Eviction <input type="checkbox"/>	
Eviction Scheduling	
Schedule	is Active <input type="checkbox"/>
	

Figure 43 Eviction Management panel (GUI)

- **Step 5:** check that the message "EVICTION CREATED" is correctly shown in the bottom right corner of the page.

7.10.1.2 Update

- **Step 1:** In the Eviction Management panel (please refer Step 1 - Step 4 of section 7.10.1.1) move mouse under the eviction to be updated in order to display the following icons highlighted in Figure 44

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

Configured Evictions: 1.



Evict_S1_FE

Keep Period: 1 - Keep Period Unit: MINUTES

Eviction is NOT active - Scheduled at: 0 0/1 * * * ?

SOFT EVICTION - Eviction Status: STOPPED



Figure 44 Eviction update via GUI

- **Step 2:** click on the “pencil icon”;
- **Step 3:** Modify the properties to be updated in the Eviction properties panel (Figure 43);
- **Step 4:** Save the updates clicking in the “floppy disk icon” (Figure 43);
- **Step 5:** check that the message “EVICTON UPDATED” is correctly shown in the bottom right corner of the page.

7.10.1.3 Deletion

In order to delete a Customizable Eviction via GUI perform the following steps:

- **Step 1:** In the Eviction Management panel (please refer Step 1 - Step 4 of section 7.10.1.1) move mouse under the eviction to be deleted in order to display the icons shown in Figure 43.
- **Step 2:** click on the “cross icon” (refer to Figure 45)

Configured Evictions: 3.



Evict_S1_FE

Keep Period: 1 - Keep Period Unit: MINUTES

Eviction is NOT active - Scheduled at: 0 0/1 * * * ?

SOFT EVICTION - Eviction Status: STOPPED

Evict_S2

Keep Period: 1 - Keep Period Unit: DAYS

Eviction is NOT active - Scheduled at: 0 0/1 * * * ?

SOFT EVICTION - Eviction Status: STOPPED

Evict_S3

Keep Period: 2 - Keep Period Unit: DAYS

Eviction is NOT active - Scheduled at: 0 0/3 * * * ?

HARD EVICTION - Eviction Status: STOPPED



Figure 45 Eviction delete via GUI

- **Step 3:** check that the message “EVICTON DELETED” is correctly shown in the bottom right corner of the page.

7.10.2 How to manage a Customizable Eviction via OData API

Evictions can be inspected and configured using the Evictions entity set on the DHuS ODatav4 API:

`http://SERVICE_URL/odata/v2/Evictions`

Only users with the System Manager role on OData can access Evictions.

7.10.2.1 JSON ODATA representation for a customizable eviction

Here after a JSON ODATA representation as an example for configuring the customizable eviction property, listed in 5.9.1:

```
{  
  "Name": "Eviction_Name",  
  "KeepPeriod": 15,  
  "KeepPeriodUnit": "MINUTES",  
  "MaxEvictedProducts": 100,  
  "Filter": "startswith(Name, 'S2')",  
  "OrderBy": "IngestionDate desc",  
  "TargetCollection": Collection_Name,  
  "SoftEviction": true,  
  "Status": "STOPPED",  
  "Cron": {"Active": false, "Schedule": "0 0/1 * * * ?"}  
}
```

7.10.2.2 Creation

The following instruction has to be performed via command line in order to create an eviction:

```
curl -H "Content-Type: application/json" -X POST --data "@/path/to/eviction.json"  
-u ${DLOGIN}:${DPASS} "http://${DHOST}/odata/v2/Evictions"
```

where the "eviction.json" is the name of the JSON file containing the proper configuration for the creation of the Customizable Eviction.

7.10.2.3 Update

The following instruction has to be performed via command line in order to update an eviction:

```
curl -H "Content-Type: application/json" -X PATCH -data "@/path/to/eviction.json"  
-u ${DLOGIN}:${DPASS} "http://${DHOST}/odata/v2/Evictions('Eviction_Name')"
```

where the "eviction.json" is the name of the JSON file containing the proper configuration for the update of the Customizable Eviction.

7.10.2.4 Deletion

The following instruction has to be performed via command line in order to delete an eviction:

```
curl -X DELETE --basic -u ${DLOGIN}:${DPASS}  
"http://${DHOST}/odata/v2/Evictions('Eviction_Name')"
```

7.10.2.5 Stop

The following instruction has to be performed via command line in order to stop the current eviction run:

```
curl -H "Content-Type: application/json" -X POST --data '{}' -u ${DLOGIN}:${DPASS} "http://localhost:8081/odata/v2/Evictions/OData.DHuS.StopEviction"
```

Please note that StopEviction action only serves to stop the currently running eviction and if the eviction is set as "Active" at the next run the eviction will work again.

7.10.2.6 Queue

The following instruction has to be launched to perform a manual activation of the eviction (QueueEviction action on OData):

```
curl -X POST --basic -u ${DLOGIN}:${DPASS} "http://${DHOST}/odata/v2/Evictions('Eviction_Name')/OData.DHuS.QueueEviction"
```

7.10.2.7 Cancel

This OData action will cancel an eviction that has been queued, setting its status to "CANCELED", meaning this eviction will not be executed once the previously running evictions are finished.

```
curl -X POST --basic -u ${DLOGIN}:${DPASS} "http://${DHOST}/odata/v2/Evictions('Eviction_Name')/OData.DHuS.CancelEviction"
```

Please note that CancelEviction action serves to cancell the next queued Eviction but not the cron: if the eviction is set as "Active" at the next run the eviction will work again.

7.10.3 How to manage a SAFE Eviction via OData API

Every Customizable Soft eviction can be triggered in SAFE mode. It means that after having configured a Soft Eviction, as per steps of 7.10.1 or 7.10.2, it can be triggered in order to act in a SAFE mode targeting only to a specific DataStore.

Prerequisites:

- Soft Eviction creation and configuration as in 5.9.1, 7.10.1 or 7.10.2;
- Setting the schedule property as not active:

```
"Cron": {"Active":false, "Schedule":"0 0/1 * * * ?"}
```

Please note that Cron property can also be set to null.

Procedure:

- **Step 1:** Trigger the SAFE eviction performing a queue action as the following one:

```
curl -H "Content-Type: application/json" -X POST --data "/path/to/safe_eviction.json" -u ${DLOGIN}:${DPASS} "http://${DHOST}/odata/v2/Evictions('Eviction_Name')/OData.DHuS.QueueEviction"
```

where in the json file are specified the following parameters:

- TargetDataStore, the name of the DataStore which products have to be evicted from;

- SafeMode parameter.

```
{  
  "TargetDataStore": "datastore-name",  
  "SafeMode": true  
}
```

7.10.4 How to manage an Automatic On-Insert Eviction

This eviction is tied to the DataStore configuration and all related actions can be performed via OData when DHuS is running or before startup using dhus.xml.

7.10.4.1 Activation

The Automatic On-Insert Eviction can be activated setting the `AutoEviction` property to true.

Include the following lines in the DataStore configuration and set them according to the needs. Please refer to 7.16.

```
"MaximumSize": 20000000000,  
"CurrentSize": 0,  
"AutoEviction": true
```

7.10.4.1.1 How to link Customizable eviction to On-Insert Automatic Eviction

It is possible to link a Customizable existing eviction to a DataStore's configuration.

Prerequisites:

- DataStore creation and configuration as in 7.10.4.1, including:

```
"AutoEviction":true
```
- Customizable Eviction existing and configured as in 5.9.1, 7.10.1 or 7.10.2, setting the schedule property as not active:

```
"Cron": {"Active":false, "Schedule":"0 0/1 * * * ?"}
```

Please note that Cron property can also be set to null.

Procedure:

- **Step 1:** create a JSON file with the following OData request:

```
{  
  "@odata.id": "http://localhost:8081/odata/v2/Evictions('Eviction_Name')"  
}
```

Where "Eviction_Name" is the name of the existing Customizable eviction to be linked.

- **Step 2:** create a link between the DataStore and the existing eviction configuration by performing an HTTP PATCH request via command line:

```
curl -H "Content-Type: application/json" -X PATCH --data
"@/path/to/link_file.json" -u {DLOGIN}:{$DPASS}
http://localhost:8081/odata/v2/DataStores('DataStore_Name')/Eviction/\$ref
```

Where the `link_file.json` is the file created at Step 1.

7.10.4.1.2 How to use linked Customizable eviction to evict LTA-retrieved products from GMPDataStore

As per any other DataStore it is possible to link a Customizable Soft-Eviction to the On-Insert Automatic eviction configured in the GMPDataStore.

This is the use case of the LTA chain, please refer to 5.11. In this frame products retrieved from LTA can be evicted again according to criteria set in the linked eviction configuration.

Here after are listed the steps suggested for the configuration of the eviction for the LTA chain:

- GMP DataStore creation and configuration as in 7.19.1, including:

```
"AutoEviction":true
```

and the `MaximumSize` parameter configured in order to set a size threshold for the folder where products are put after restoring from LTA.

- Customizable Eviction existing and configured as in 5.9.1, 7.10.1 or 7.10.2 with the following criteria:
 - `Cron` property set as not active:
`"Cron": {"Active":false, "Schedule":""}`
 - `KeepPeriod` configured according to the need, in order to keep online restored products only within a time range.
 - `Online` property set as filter
`"Filter": Online eq true`
 - `OrderBy` criteria set for `CreationDate` in order to evict products in FIFO way:
`"OrderBy": CreationDate asc`
- Link the configured soft eviction to the GMP DataStore as per 7.10.4.1.1

7.11 Main Menu Settings

In **OWC**, with the Administrator privileges, it is possible to manage the accessibility to all the functionalities/items in the left side of the Home web page (red arrows), by means of the Main Menu Panel in the Settings section.

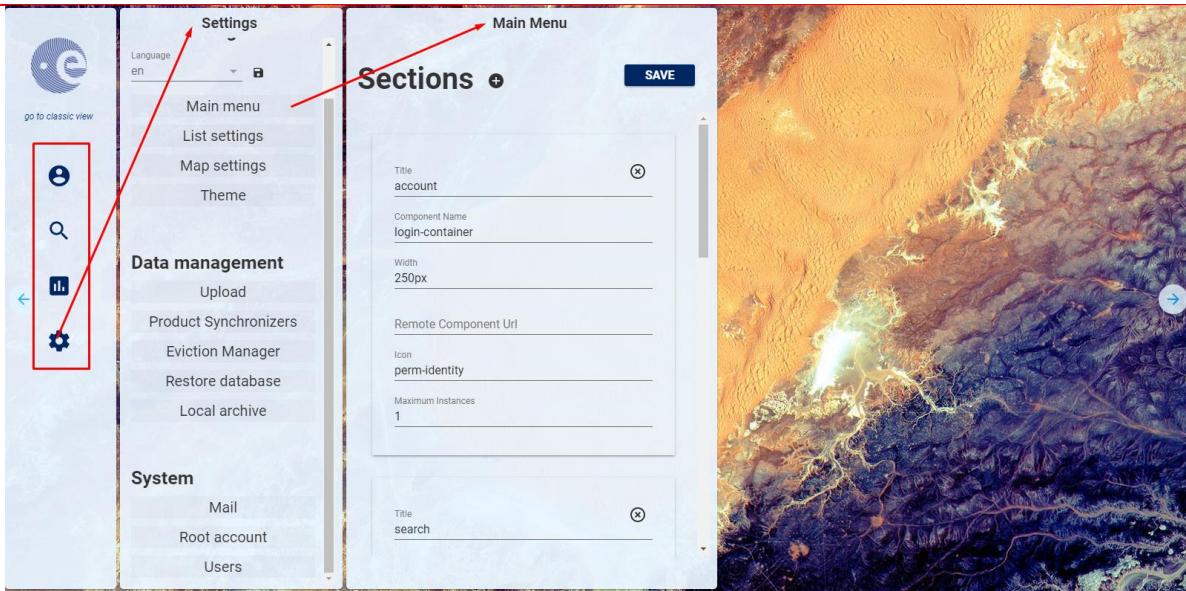


Figure 46 Main Menu Settings panel (OWC)

7.12 Auxiliary Data Files

7.12.1 How to configure ADF retrieval chain

Pre-requisites

The following instances are needed to set up the Auxiliary Data Files retrieval chain:

- DHuS instance containing Sentinel-3 Products. Users will access to ADF Search feature from OWC GUI installed on this DHuS instance.
- DHuS instance containing Sentinel-3 Auxiliary Data Files. It is recommended to use the same distribution version of DHuS instance containing Sentinel-3 Products. This DHuS instance has to be accessible to DHuS instance containing Sentinel-3 Products via HTTP connection.

Procedure

- **Step 1:** Create an account with search and download rights (e.g. username: adfsearch, password adfsearch) on the DHuS instance containing Sentinel-3 Auxiliary Data Files.
- **Step 2:** Connect to the machine where DHuS instance containing Sentinel-3 Products is installed and check that the application is not running (`ps -eaf | grep dhus`)
- **Step 3:** Edit file `etc/conf/ExternalDHuS.properties` adding the following parameters needed to connect with DHuS instance containing ADF:
 - **external_dhus_host:** hostname or IP address of DHuS instance containing Sentinel-3 auxiliary files, ending with slash ('/') character (e.g. [http://\[DHuS_address\]/](http://[DHuS_address]/))
 - **external_dhus_username:** username of the account on DHuS instance containing Sentinel-3 auxiliary files (e.g. adfsearch)

- **external_dhus_password:** password of the account on DHuS instance containing Sentinel-3 Auxiliary files (e.g. adfsearch)

An example of configuration is the following:

```
external_dhus_host=http://192.168.10.79:8091/  
external_dhus_username= adfsearch  
external_dhus_password= adfsearch
```

- **Step 4:** Start the DHuS as single instance entering the following command in the installation directory:

```
nohup /bin/bash ./start.sh &
```
- **Step 4:** Edit the file `/DHUSDIR]/var/tomcat/webapps/ROOT/config/appconfig.json` and enable the Auxiliary File Search feature:

```
"adf"=true
```

7.13 User details information

It is possible to visualize user details via OData (see [RD 6]) by accessing to the entity "Users" at:

```
https://DHuS_URL/odata/v1/Users
```

Users with the user administration right are allowed to details of all the users registered in the Hub, whereas normal users are allowed to visualize only their own.

Please refer to [RD 6] for further details about the "Users" OData entities and its information such as system roles, connections, restrictions and user cart.

7.14 Connections information

Administrators can visualize all the connections managed by DHuS via OData if `server.xml` file has been configured properly (8.2). In fact, every user request is available in the "Connections" entity (see [RD 6]). Moreover it is possible to monitor the activity of a specific user visualizing the requests sent by him/her at:

```
https://DHuS_URL/odata/v1/Users('Username')/Connections
```

Please refer to [RD 6] for the list of information exposed for each connection stored in the DHuS cache.

7.14.1 How to identify Active users

Administrators can list active users on DHuS through OData Connections entity. The following query shows all the users in status PENDING, i.e. users whose requests are being processed by DHuS at the time of the verification:

```
https://DHuS_URL/odata/v1/Connections?$filter>Status%20eq%20%27PENDING%27&$expand=User
```

7.15 Network information

OData exposes to administrators some information about the network used by the Hub. To access such info it is necessary to connect via OData to the "Networks" entity at:

```
https://DHuS_URL/odata/v1/Networks
```

Please refer to [RD 6] for the information exposed for each Network stored in the DHuS cache and its Statistics.

7.16 DataStore Management

7.16.1 JSON ODATA representation for HFSDataStore

All the HFSDataStore properties listed in 6.1.2 can be configured via the following JSON OData representation:

```
{
  "@odata.type": "#OData.DHuS.HFSDataStore",
  "Name": "HFSDataStore_Name",
  "ReadOnly": false,
  "Priority": 0,
  "MaximumSize": 1000,
  "CurrentSize": 0,
  "AutoEviction": true,
  "Path": "/path/to/incoming/folder",
  "MaxFileDepth": 10,
  "MaxItems": 1024
}
```

7.16.2 JSON ODATA representation for Openstack DataStore

All the Openstack DataStore properties listed in 6.1.2 can be configured via the following JSON OData representation:

```
{
  "@odata.type": "#OData.DHuS.OpenStackDataStore",
  "Name": "Openstack_DataStore_Name",
  "ReadOnly": false,
  "Priority": 0,
  "MaximumSize": 1073741824,
  "CurrentSize": 0,
```

```
"AutoEviction": false,  
"Provider": "openstack-swift",  
"Identity": "tenantName:tenantId",  
"Credential": "password",  
"Url": "https://auth.cloud.ovh.net/v2.0",  
"Region": "SBG3",  
"Container": "Container_Name"  
}
```

7.16.3 DataStore CRUD operations

The DataStores entity set supports CRUD (Create, Read, Update, Delete) operations on DataStores and allows to manage the configuration of DataStores within the DHuS at runtime.

An individual DataStore can be accessed using its reference property, which is the Name:

```
[DHUS_HOST]/odata/v2/DataStores('DataStore_Name')
```

In the following Sections command line samples, using cURL commands, are provided in order to manage the DataStore creation, update and deletion at runtime. When an action related to DataStore entities is performed, it is reported at real-time on the dhus.xml file.

DataStore can be also created directly on dhus.xml file before the first start of the DHuS service. Please refer to Section 8.1 for details about DataStores configuration on dhus.xml file.

Warning: please note that white spaces are not allowed in the DataStore name.

7.16.3.1 How to create a DataStore

The following instruction has to be performed via command line in order to create a DataStore:

```
curl -H "Content-Type: application/json" -X POST --data  
"@/path/to/file/DataStore.json" -u ${DLOGIN}:${DPASS}  
"http://localhost:8081/odata/v2/DataStores"
```

where the "DataStore.json" is the name of the JSON file containing the proper configuration for the creation of the DataStore. Refer to:

- Section 7.16.1 for example related to HFS DataStore;
- Section 7.16.2 for example related to Openstack DataStore;
- Section 7.19.1 related to GMP DataStore.

Warning: please note that white spaces are not allowed in the DataStore name.

7.16.3.2 How to Update a DataStore

The following instruction has to be performed via command line in order to update a DataStore:

```
curl -H "Content-Type: application/json" -X PATCH --data
"@/path/to/file/DataStore.json" -u ${DLOGIN}:${DPASS}
"http://localhost:8081/odata/v2/DataStores('DataStore_Name')"
```

where the "DataStore.json" is the name of the JSON file containing the updates and "DataStore_Name" is the name of the DataStore to be updated.

7.16.3.3 How to Delete a DataStore

```
curl -u ${DLOGIN}:${DPASS} -X DELETE --basic
"http://localhost:8081/odata/v2/DataStores('DataStore_Name')"
```

where "DataStore_Name" is the name of the DataStore to be deleted.

7.16.3.4 How to list products stored in a DataStore

Administrators can list products stored in a DataStore performing the following OData query:

```
http://localhost:8081/odata/v2/DataStores('DataStore_Name')/Products
```

Products are listed through their Id property.

7.17 OpenStack storage command line samples

In this section, we report some OpenStack storage command line samples, using curl & jq commands:

- Get token and swift url for specific user / project (tenant)

```
curl -s -d '{"auth": {"tenantName": "<tenantName>",
"passwordCredentials": {"username": "<username>", "password": "<password>"} }' -H "Content-type: application/json"
https://keystone9915.openstack.ovh.net:35358/v2.0/tokens | jq '{url:
.access.serviceCatalog[] | select(.name=="swift") | .endpoints[].publicURL,
token: .access.token.id}'
```

- List containers in the account

```
curl -i -H "X-Auth-Token: <token>" <url> -X GET
```

- Create container

```
curl -i -H "X-Auth-Token: <token>" <url>/<container_name> -X PUT
```

- Info about container content

```
curl -i -H "X-Auth-Token: <token>" <url>/<container_name> -X HEAD
```

- Delete container

```
curl -i -H "X-Auth-Token: <token>" <url>/<container_name> -X DELETE
```

Please note that a container can be deleted only if it is empty.

- Info about object in a container

```
curl -H "X-Auth-Token: <token>" <url>/<container_name>/<object_name> --head
```

- Delete object

```
curl -i -H "X-Auth-Token: <token>" <url>/<container_name>/<object_name> -X  
DELETE
```

7.18 NetCDF Temporary Cache setting

7.18.1 How to disable the Cache

The following option should be added in the `start.sh` file to disable the cache, if needed:

- `-Dfr.gael.streams.cache.inactive=true`

7.18.2 How to configure the Cache

7.18.2.1 Default cache configuration

The default configuration of the cache mechanism is the following:

- 1000 maximum cached files;
- 10 GB maximum total size.

7.18.2.2 Customized cache configuration

Since the temporary cache files can take a substantial amount of disk space for large products, it is possible:

- to choose the location of the cache;
- to limit the number of files in the cache;
- to limit the size of files in the cache.

Location

The location of the temporary cache folder can be configured by setting the Java system property "fr.gael.streams.tmpdir" to the path of the desired folder.

This system property is set using `-Dfr.gael.streams.tmpdir` in the `start.sh` startup script of the DHuS instance as follows:

- `-Dfr.gael.streams.tmpdir="/my/temp/file/location" \`

The default temporary cache folder is the Tomcat tmp folder contained in the DHuS var directory.

Size management

The size of the cache can be managed in two ways:

- Limit the number of files that can be created in the cache
- Limit the total size of the cache

Temporary cache files have no expiration date and will remain until they are evicted due to size management limits. These limits can be configured in a `temporary_files_ehcach.xml` file that should be created (if not present) in the `etc` folder of the DHuS instance.

Here follows an example of `temporary_files_ehcach.xml`.

```
<ehcache name="TemporaryFilesCacheManager">
<cache name="TemporaryFilesCache"
maxEntriesLocalHeap="1000"
maxBytesLocalDisk="10000000000" >
<cacheEventListenerFactory
class="fr.gael.streams.TemporaryFilesCacheListener" />
</cache>
</ehcache>
```

The `maxEntriesLocalHeap` limits the number of temporary cache files while the `maxBytesLocalDisk` limits the total size of the cache in bytes.

In order to configure the `maxBytesLocalDisk` parameter with the appropriate size preventing errors due to cache too small, please refer to the following formula:

$$\text{maxBytesLocalDisk} = \text{corePoolSize} * \text{maximumSize} * 1.3$$

where `corePoolSize` is the value set in the `dhus.xml` file of the DHuS instance performing ingestion and `maximumSize` is the maximum known size of the ingested products. The result has to be expressed in bytes.

7.18.2.3 How to configure the shared cache

DHuS deployed in Scalability 2.0 mode support the sharing of netCDF temporary cache using Ehcache and Java RMI (Remote Method Invocation) providing remote communication between DHuS instances (nodes).

The procedure below describes the steps needed to share the cache folder. It assumes that all DHuS instances are stopped.

Step ID	User	Tasks and Commands
1.	Standard user	Create a folder where the temporary cache files will be stored and shared between all DHuS nodes, making sure this folder is achievable by all the DHuS instances (nodes).
2.	Standard user	Configure the location of the netCDF cache folder at Step 1 in the <code>start.sh</code> as in 7.18.2.
3.	Standard user	Access the DHuS <code>etc</code> configuration folder in the machine where the selected DHuS instance (node) has been installed and perform a backup of the existing <code>temporary_files_ehcach.xml</code> file. The new file (named

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

		<p>“temporary_files_ehcache.xml”) will contain the configuration that will overwrite the default one.</p> <pre>> cd <DHuS_INST_DIR>/etc > cp temporary_files_ehcache.xml temporary_files_ehcache.xml.bk</pre>
4.	Standard user	<p>Open the temporary_files_ehcache.xml file and set the properties of cacheManagerPeerProviderFactory attribute in order to use manual peer discovery, as follows:</p> <pre><cacheManagerPeerProviderFactory class="net.sf.ehcache.distribution.RMICacheManagerPeerProviderFactory" properties="peerDiscovery=manual, rmiUrls=<RMI_URLS> propertySeparator=","/></pre> <p>Manual peer configuration requires the IP address and port of each listener to be known. Peers cannot be added or removed at runtime.</p> <p>The rmiUrls property contains the list of all cache peers for all cache listener (detailed via IP address and port); cache peers are separated by means of “ ” character. Do not include the server being configured in the list.</p> <p>Refer to Step 7. of this procedure for an example containing cacheManagerPeerProviderFactory attribute setting.</p>
5.	Standard user	<p>Configure the CacheManagerPeerListener listening for messages from peers to the current CacheManager. In the cacheManagerPeerListenerFactory attribute set properties as follows:</p> <pre><cacheManagerPeerListenerFactory class="net.sf.ehcache.distribution.RMICacheManagerPeerListenerFactory" properties="hostName=<HOST>,port=<PORT>,remoteObjectPort=<RO_PORT>, socketTimeoutMillis=<TIMEOUT>" /></pre> <p>Please note that:</p> <ul style="list-style-type: none"> • hostname is the hostname/IP of the host the listener is running on; • port is the port the listener listens on; • remoteObjectPort is the port number on which the remote objects bound in the registry receive calls; • socketTimeoutMillis is the TCP/IP Socket timeout when waiting on response (i.e. number of seconds client sockets will wait when sending messages to this listener until they give up). <p>Please refer to Step 7. of this procedure for an example of cacheManagerPeerListenerFactory attribute setting.</p>
6.	Standard user	<p>Add the following properties in the <cache> </cache> section:</p> <pre><cacheEventListenerFactory class="net.sf.ehcache.distribution.RMICacheReplicatorFactory" properties="replicateAsynchronously=true" /> <bootstrapCacheLoaderFactory class="net.sf.ehcache.distribution.RMIBootstrapCacheLoaderFactory"</pre>

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

		properties="bootstrapAsynchronously=true" propertySeparator="," />
7.	Standard user	<p>Please find below an example of temporary_files_ehcache.xml file.</p> <pre><ehcache name="TemporaryFilesCacheManager"> <!-- Manual discovery --> <cacheManagerPeerProviderFactory class="net.sf.ehcache.distribution.RMICacheManagerPeerProviderFactory" properties="peerDiscovery=manual,rmiUrls=/172.30.246.20:40001/TemporaryFilesCache /172.30.246.46:400001/TemporaryFilesCache" propertySeparator=","/> <!-- create cluster listener on port 40001 --> <cacheManagerPeerListenerFactory class="net.sf.ehcache.distribution.RMICacheManagerPeerListenerFactory" properties="hostName=172.30.246.11,port=40001,remoteObjectPort=9999 ,socketTimeoutMillis=2000" /> <cache name="TemporaryFilesCache" eternal="false" overflowToDisk="false" diskPersistent="false" diskExpiryThreadIntervalSeconds="0" maxEntriesLocalHeap="1000" maxBytesLocalDisk="10000000000" memoryStoreEvictionPolicy="FIFO"> <cacheEventListenerFactory class="fr.gael.streams.TemporaryFilesCacheListener" /> <cacheEventListenerFactory class="net.sf.ehcache.distribution.RMICacheReplicatorFactory" properties="replicateAsynchronously=true" /> <bootstrapCacheLoaderFactory class="net.sf.ehcache.distribution.RMIBootstrapCacheLoaderFactory" properties="bootstrapAsynchronously=true" propertySeparator="," /> </cache> </ehcache></pre>
8.	Standard user	Repeat Steps 2-8 for all the DHuS nodes.

7.19 Sentinel-1 PDGS LTA component Interface setting

7.19.1 JSON ODATA representation for GMPDataStore

All the GMPDataStore properties listed in Section 6.1.2 can be configured via the following JSON OData representation.

```
{
"@odata.type": "#OData.DHuS.GMPDataStore",
"Name": "GMPDataStore_Name",
```

```
"Priority": 200,
"ReadOnly": false,
"MaximumSize": 20000000000,
"CurrentSize": 0,
"AutoEviction": false,
"GMPRepoLocation": "/path/to/gmpRepoLocation",
"HFSLocation": "/path/to/hfsLocation",
"MaxQueuedRequest": 25,
"Quotas":
{
  "MaxQueryPerUser": 1,
  "Timespan": 60000
},
"IsMaster": true,
"MySQLConnectionInfo":
{
  "DatabaseUrl":
    "jdbc:mysql://172.30.246.46:3306/ivvgmp?serverTimezone=Europe/Paris",
  "User": "gmp_user",
  "Password": "gmp_password"
},
"Configuration":
{
  "AgentId": "wget-auth",
  "TargetId": "pac2oda"
}
```

7.19.2 How to manage GMPDataStore

Please refer to Section 7.16.3 for details on how to create/update/delete a DataStore.

7.20 Sentinel-2 Archive Interface setting

7.20.1 JSON ODATA representation for RemoteDHus DataStore

All the RemoteDHuSDataStore properties listed in Section 6.1.2 can be configured via the following JSON OData representation.

```
{
  "@odata.type": "#OData.DHuS.RemoteDHuSDataStore",
  "Name": "IVVRemoteDHuSDataStore",
  "ServiceUrl": "https://dhus-s2.gael-systems.com/odata/v1",
  "Login": "ivvuser",
  "Password": "password",
  "Priority": 150,
  "ReadOnly": true
}
```

{}

7.20.2 How to manage RemoteDHuSDataStore

Please refer to Section 7.16.3 for details on how to create/update/delete a DataStore.

8. DHuS configuration file parameters

8.1 dhus.xml

This section explains in details the parameter contained in the `dhus.xml` configuration file.

The whole DHuS configuration is contained inside the `<configuration></configuration>` tag.

In Appendix A is reported the complete `dhus.xml` of the DHuS distribution described in this document, as it is before the DHuS start without having configured Eviction, Synchronizer, DataStore and FileScanners. In the `dhus.xml` file coming from the DHuS distribution, additional sections related to SmartSynchronizers and Sources are present; please note that this kind of feature is not fully supported in the DHuS release covered by this Administration Manual version. As a consequence, it is recommended to avoid the use of this feature.

Please note that since DHuS 0.14.X branch, after the first DHuS start, `dhus.xml` file is automatically reshaped by the DHuS.

It is recommended to configure a folder where DHuS will put its temporary files. Here follows details about the `workingDir` attribute, necessary for this purpose:

Parameter	Description	Syntax
<code>workingDir</code>	Folder where DHuS puts its temporary files. This folder is automatically deleted when the DHuS is switched off. Currently, DHuS uses this folder during ingestion. Avoid to set it as the DHuS installation folder.	<code>workingDir="/path/to/working/directory"</code>

Groups of configurable options are present inside the `<configuration></configuration>` tag:

- crons
- messaging
- network
- products
- search
- server
- system
- DataStores
- scanners

- synchronizers
- evictions

The following Sections describe the specific parameters for each Group, explaining how to configure them.

8.1.1 crons

[Groups]/ Parameter	Description	Usage
[crons]/archiveSynchronization	Cron used to synchronize local archive (whose path is the one specified in the parameter "archive" in the "system" group). In case of eviction, please schedule it after the eviction.	Activate it and specify the schedule.
[crons]/cleanDatabase	Cron used to clean database, like removing old statistics or old not confirmed users.	Activate it and specify the schedule. The parameter <code>statistics</code> and <code>tempUsers</code> define, respectively, for how long (in days) the system have to keep statistics and how many time (in days) the users have to confirm their registration request
[crons]/dumpDatabase	Cron used to dump database	Activate it and specify the schedule.
[crons]/cleanDatabaseDump	Cron used to clean database dumps.	Activate it and specify the schedule. The parameter <code>keep</code> defines how many dumps are stored.
[crons]/fileScanners	Cron used to execute user saved filescanners.	Activate it and specify the schedule. The parameter <code>sourceRemove</code> defines if found products shall be removed from source
[crons]/searches	Cron used to execute user saved searches and send results to users	Activate it and specify the schedule.
[crons]/sendLogs	Cron used to send system logs.	Activate it and specify the schedule. The parameter <code>addresses</code> logs recipients addresses. They shall be coma-separated.
[crons]/systemCheck	Cron used to check all system coherence, including database optimization	Activate it and specify the schedule.

Example:

```
<crons>
```

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

```

<cron:archiveSynchronization active="false" schedule="0 0 0/5 ? * *"/>
<cron:cleanDatabase active="false" schedule="0 0 1 ? * *">
    <cron:tempUsers keepPeriod="10" />
    <cron:statistics keepPeriod="90" />
</cron:cleanDatabase>
<cron:dumpDatabase active="true" schedule="0 0 3 ? * *"/>
<cron:cleanDatabaseDump active="true" schedule="0 0 4 ? * *" keep="10" />
<cron:fileScanners active="true" schedule="0 0/2 * * ?" sourceRemove="true"/>
<cron:searches active="false" schedule="0 0 5 ? * *"/>
<cron:sendLogs active="false" schedule="0 55 23 ? * *" addresses="ivv@ivv.com" />
<cron:systemCheck active="false" schedule="0 30 23 ? * *"/>
</crons>

```

8.1.2 messaging

[Groups]/ Parameter	Description	Usage
[messaging]/mail	<p>Mail configuration.</p> <p>This field defines the rules about mail notification to users.</p> <p>The set values are used only at first launch of the system. They shall be modified in Management Panel if needed.</p> <p>Every notification type needs to be activated. In case of activation of one or more notification ,the following information shall be configured:</p> <ul style="list-style-type: none"> • sender mail server specification • information about mail sender • "reply to" address of sent mails 	<p>Mail Services:</p> <ul style="list-style-type: none"> • <code>onUserCreate</code>: true false. It defines if system send mail when creating user • <code>onUserUpdate</code>: true false. It defines if system send mail when updating user • <code>onUserDelete</code>: true false. It defines if system send mail when deleting user <p>Server information:</p> <ul style="list-style-type: none"> • <code>smtp</code>: server address • <code>port</code>: server port • <code>tls</code>: defines if server is using TLS protocol • <code>username</code> and <code>password</code>: connection information • <code>name</code>: displayed name of "from" part of the notification email • <code>address</code>: displayed address of "from" part of the notification email • <code>reply to</code>: defines the "reply to" address of sent mails

Example:

```

<messaging>
    <msg:mail onUserCreate="true" onUserUpdate="true" onUserDelete="true">
        <msg:server smtp="131.176.235.251" port="25" tls="false"
username="scihub@copernicus.esa.int" password="$secreT123">
            <msg:from name="IVV DHuS Team" address="ivv@copernicus.esa.int" />
            <msg:replyTo>ivv@copernicus.esa.int</msg:replyTo>
        </msg:server>
    </msg:mail>
</messaging>

```

```
</msg:mail>
</messaging>
```

8.1.3 network

[Groups]/ Parameter	Description	Usage
[network]/outbound	<p>Outbound bandwidth configuration. These parameters are recharged every time the dhus is restarted.</p> <p>Parameters to be configured: channel name and its weight, including the user email pattern identifying the outbound channel, default user quota including maxconcurrent, max count, max size, max cumulative size and max bandwidth.</p>	<ul style="list-style-type: none"> maxConcurrent defines the maximum simultaneous accepted transfers. maxCount defines the maximum number of accepted transfers on a customizable period. "periodUnit" attribute defines the unit of the period. Possible units are "DAYS", "HOURS", "MICROSECONDS", "MILLISECONDS", "MINUTES", "NANOSECONDS", "SECONDS". "period" attribute is a sliding time window used to count number of transferred product from now to the past period delay. maxSize defines the maximum accepted size of transfer file. maxCumulativeSize defines the maximum cumulated accepted transfers size on a customizable period. maxBandwidth defines the maximum bandwidth authorized for the referred channel.
[network]/inbound	Inbound bandwidth configuration	see above

Example:

```
<network>
  <network:outbound>
    <network:channel name="PriorityChannel" weight="75">
      <network:classifier>
        <network:includes>
          <network:include>
            <network:userEmailPattern>.*@serco.com</network:userEmailPattern>
          </network:include>
        </network:includes>
      </network:classifier>
    </network:channel>
    <network:channel name="SelfRegisteredChannel" weight="25">
      <network:defaultUserQuotas>
        <network:maxConcurrent>2</network:maxConcurrent>
      </network:defaultUserQuotas>
    </network:channel>
  </network:outbound>
  <network:inbound>
```

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

```

<network:channel name="PriorityChannel" weight="75">
    <network:classifier>
        <network:includes>
            <network:include>
                <network:userEmailPattern>.*@serco.com</network:userEmailPattern>
            </network:include>
        </network:includes>
    </network:classifier>
</network:channel>
<network:channel name="SelfRegisteredChannel" weight="25">
    <network:defaultUserQuotas>
    </network:defaultUserQuotas>
</network:channel>
</network:inbound>
</network>

```

8.1.4 products

[Groups]/ Parameter	Description	Usage
[products]/download	<p>Download configuration.</p> <p>CompressionLevel parameter defines how rigorously the compressor looks for the longest string possible. As a general rule of thumb: Compressing at the maximum level 9 requires around twice as much processor time as compressing at the minimum level 1. For typical input, compressing at the maximum as opposed to the minimum level adds around 5% to the compression ratio. 0 value means no compression.</p> <p>checksumAlgorithms shall be comma separated.</p> <p>Warning: Starting from 0.14.2 version, publicData=parameter is no more present.</p>	<ul style="list-style-type: none"> • compressionLevel: is a value to build the file compression, from 1 to 9. • checksumAlgorithms: algorithm to use for checksum computations (e.g. "MD5").
[products]/quicklook	<p>Quicklook calculation parameters</p>	<ul style="list-style-type: none"> • height: height of generated quicklooks • width: width of generated quicklooks • cutting: true false. It allows system to cut image when processing quicklooks <p>Specify dimensions in pixel unit.</p>

[products]/thumbnail	<p>Thumbnail calculation parameters</p> <ul style="list-style-type: none"> • height: height of generated thumbnails • width: width of generated thumbnails • cutting: true false. It allows system to cut image when processing thumbnails <p>Specify dimensions in pixel unit.</p>
----------------------	---

Example:

```
<products>
  <product:download compressionLevel="4" checksumAlgorithms="MD5"/>
  <product:quicklook height="512" width="512" cutting="false" />
  <product:thumbnail height="64" width="64" cutting="false" />
</products>
```

8.1.5 search

[Groups]/ Parameter	Description	Usage
[search]/geocoder	Geocoder (nominatim) configuration.	<p>Set up the Nominatim geocoder and the Geoname configurations:</p> <ul style="list-style-type: none"> • boundingBox: defines if the geocoder is querying only the bounding box of the matching place from the Nominatim Web Service i.e. the four corners encompassing the place. Otherwise, it will query the complete polygon boundaries, that may have lower performance according to the number of vertices composing the place's boundaries. • maxPointNumber: maximum number of points that can be returned for a polygon • username: username used to connect to Geoname
[search]/odata	OData configuration.	<ul style="list-style-type: none"> • defaultTop: maximum rows returned by OData and OpenSearch services
[search]/solr	Solr configuration	<ul style="list-style-type: none"> • path: solr path • core: solr core name • schemaPath: solr schema path. Shall be empty. • synonymPath: path of solr synonyms file

Example:

```
<search>
  <search:geocoder url="http://nominatim.openstreetmap.org">
    <search:nominatim boundingBox="false" maxPointNumber="50" />
    <search:geoname username="dhus" />
  </search:geocoder>
  <search:odata defaultTop="50" />
  <search:solr path="/solr" core="dhus" schemaPath="" synonymPath="" />
</search>
```

8.1.6 server

[Groups]/ Parameter	Description	Usage
[server]/external	<p>External url viewed by users.</p> <p>Used in case of an apache proxy redirection for example.</p> <p>Empty values mean that server values are those which are viewed by users.</p>	protocol://host:port/path
[server]/ftp	Warning: Starting from 0.14.4 version, <code>ftp</code> parameter is no more present.	

Example:

```
<server>
  <server:external protocol="https" host="131.176.236.20" port="443" path="/dhus" />
</server>
```

8.1.7 system

[Groups]/ Parameter	Description	Usage
[system]/administrator	<p>Definition of principal administrator user.</p> <p>If User exists, DHuS will give him all rights at launch, but will not his password. This shall be done in Management panel of GUI.</p> <p>If User is not existing, DHuS will create it with defined password.</p>	Specify username and password
[system]/archive	<p>Definition of local archive path for the error folder where products got in error during ingestion/synchronization will be stored.</p> <p>For products that do not fall into</p>	<ul style="list-style-type: none"> • <code>errorPath</code>: path where to move products not ingested

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

	error please refere to 8.1.8.	
[system]/database	<p>Definition of database path, where dumps are stored and, optionally, the encryption type and encryption key.</p> <p>Warning: please note that, from the 0.10.3-4 version, the default behaviour is no encryption.</p>	<ul style="list-style-type: none"> <code>JDBCDriver</code>: component enabling DHuS to interact with the database <code>dumpPath</code>: defines the location where store database (valid only for HSQL embedded database) <code>JDBCUrl</code>: defines the database to be used and its properties <code>Login</code>: user to access database <code>Password</code>: credential to access database
[system]/name	Definition of system long name and short name.	<ul style="list-style-type: none"> <code>long</code>: system long name <code>short</code>: system short name
[system]/processing	<p>Processing configuration.</p> <p>Configure the <code>corePoolSize</code> as an integer number (>0).</p>	<ul style="list-style-type: none"> <code>corePoolSize</code>: defines maximal number of active threads for the ingestion process. Default is 1.
[system]/support	<p>Definition of support name and mail.</p> <p>These values are used only at first launch of the system.</p> <p>They shall be modified in Management Panel if needed via UI.</p>	<ul style="list-style-type: none"> <code>registrationMail</code>: used to send the administrative registration information. If this field is not set, DHuS is using support mail.
[system]/tomcat	Definition of tomcat path	<ul style="list-style-type: none"> <code>path</code>: defines the location where store tomcat folder
[system]/executor	<p>The background service that execute synchronizers.</p> <p>It must be enabled if you want to use the synchronisation feature</p> <p>In the batchmode the executor will run the synchronizers until there is no more to synchronize.</p>	<ul style="list-style-type: none"> <code>enabled</code>: true false <code>batchModeEnabled</code>: true false
[system]/trashPath	<p>Definition of trashpath path.</p> <p>The trashpath, if configured, is the folder used by DHuS to move there evicted products. It can be set empty ("") if there is no need to</p>	<ul style="list-style-type: none"> <code>path</code>: defines the location where store evicted products.

	keep product on filesystem.
--	-----------------------------

Example:

```

<system>
    <system:administrator name="root" password="password" />
    <system:archive path="&varFolder;/local_archive">
        <system:incoming maxFileNo="10" maxItems="10" path="/PUP2/dhus-test/var_IVV/incoming"
errorPath="/COMMON/IVV/errorpath"/>
    </system:archive>
    <system:database JDBCDriver="org.hsqldb.jdbcDriver"
        hibernateDialect="org.hibernate.dialect.HSQLDialect"
        dumpPath="&varFolder;/dump"
JDBCUrl="jdbc:hsqldb:file:&varFolder;/database/dhus;crypt_type=AES;crypt_key=5f0ebc384f232c
9108469a003724a5e2"
        login="sa"
        password="" />
    <system:name long="IVV Data Hub Service" short="IVV DHuS" />
    <system:processing corePoolSize="12" />
    <system:support name="DHuS Support Team" mail="ivv@copernicus.esa.int"
registrationMail="ivv@copernicus.esa.int" />
    <system:tomcat path="&varFolder;/tomcat" />
    <system:executor enabled="true" batchModeEnabled="false" />
    <system:trashPath path="" />
</system>

```

8.1.8 dataStores

It is possible to configure multiple DataStores in DHuS; four kinds of DataStores are currently available: Hierarchically File System (HFS), Openstack, GMPDataStore, RemoteDHuS DataStore.

[Groups]/ Parameter	Description	Usage
[dataStores]/dataStore	<p>This section is referred to the DataStore configuration.</p> <p>All DataStores share some common attributes:</p> <ul style="list-style-type: none"> • name • readOnly • priority <p>It is necessary to configure a DataStore for each of the archive where the products are located.</p> <p>Warning: GMPDataStore has to be configured with the lowest priority (i.e. high number). RemoteDHuS DataStore has to be configured with intermediate priority among the priority of the GMPDataStores and the other DataStore present.</p>	<ul style="list-style-type: none"> • type: define the DataStore implementation to use ("hfsDataStore", "openStackDataStore" "gmpDataStore" and "RemoteDHuSDataStore") • name: unique DataStore name • readOnly: true/false. If it is set to true the adding, moving and deleting actions are not supported for the DataStore. Default is read-write (i.e. readOnly: false) except for RemoteDHuSDatastore. • priority: priority with which DataStores are accessed within the system. DataStores with the smallest priority are accessed first. Default is 100. • maximumSize: threshold in byte of the DataStore size. It is the trigger for the automatic OnInsert eviction. • currentSize: size of the DataStore . Automatically calculated by DHuS. • autoEviction: true false. It means if the On-Insert automatic eviction is to be triggered or not. • path (only for HSF DataStore): local storage

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

	<p>path</p> <ul style="list-style-type: none"> • <code>maxFileNo</code> (only for HSF DataStore): maximum number of sub-folder for each stage • <code>maxItems</code> (only for HFS DataStore): maximum number of file in a folder • <code>provider</code> (only for Openstack DataStore): provider service to use, only "<code>openstack-swift</code>" is currently available • <code>identity</code> (only for Openstack DataStore): identifier for the authentication service • <code>credential</code> (only for Openstack DataStore): <code>project:password</code> of an openstack account • <code>url</code> (only for Openstack DataStore): URL of Openstack authentication service • <code>container</code> (only for Openstack DataStore): container to use. Please note that the container shall be created manually before configuring this parameter. • <code>region</code> (only for Openstack DataStore): Each Region has its own full Openstack deployment, including its own API endpoints and networks . The region is linked to the openstack account. The default value is <code>regionOne</code> • <code>GMPRepoLocation</code> (only for GMPDataStore): path to the folder where GMP software store retrieved products in .zip format • <code>HFSLocation</code> (only for GMPDataStore) : folder where the retrieved products are moved by GMPDataStore for being available for the user • <code>MaxQueueRequest</code> (only for GMPDataStore): it defines max quota for instance • <code>MaxQueryPerUser</code> (only for GMPDataStore): quota for users in the time range selected • <code>Timespan</code> (only for GMPDataStore): it defines time range in milliseconds for the user quota • <code>IsMaster</code> (only for GMPDataStore): true false. It defines the instance in charge of download and restore actions. • <code>DatabaseUrl</code> (only for GMPDataStore): GMP database path. • <code>User</code> (only for GMPDataStore): user to access GMP database • <code>Password</code> (only for GMPDataStore): password to access to GMP database • <code>AgentId</code> (only for GMPDataStore): agent identifier used by GMP software for the product retrieval • <code>TargetId</code> (only for GMPDataStore):
--	--

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

		<p>Processing Archive Centres acting for the product retrieval</p> <ul style="list-style-type: none"> ServiceUrl: Url of the OData v1 service of the Remote instance where products are stored and which the RemoteDHus DataStore has to interact with. Login: username to be used to request the remote DHuS instance. The user has to be previously created in the Remote DHuS instance with no quota limitation. Password: The password to be used to request the remote DHuS instance.
--	--	--

Example of configuration for HFS DataStore:

```
<ds:dataStore xsi:type="ds:dfsDataStoreConf"
    name="DataStore1_name"
    readOnly="true"
    priority="0"
    maximumSize="1000"
    currentSize="0"
    autoEviction="true">
<ds:path>/data/dhus-incoming</ds:path>
<ds:maxFileNo>10</ds:maxFileNo>
<ds:maxItems>1024</ds:maxItems>
</ds:dataStore>
```

Example of configuration for Openstack DataStore:

```
<ds:dataStore xsi:type="ds:openStackDataStoreConf"
    name="DataStore2_name"
    readOnly="false">
<ds:provider>openstack-swift</ds:provider>
<ds:identity> tenantName:username</ds:identity>
<ds:credential>password</ds:credential>
<ds:url>https://keystone9915.openstack.ovh.net:35358/v2.0</ds:url>
<ds:container>container_name</ds:container>
<ds:region>RegionOne</ds:region>
</ds:dataStore>
```

Example of configuration for GMPDataStore:

```
<ds:dataStore xsi:type="ds:gmpDataStoreConf"
    isMaster="true"
    name="IVVGMPDataStore"
    readOnly="false"
    priority="200"
    maximumSize="20000000000"
    currentSize="0"
    autoEviction="false">
<ds:mysqlConnectionInfo user="user"
password="password">jdbc:mysql://172.30.246.46:3306/ivvgmp?serverTimezone=Europe/Paris</ds
:mysqlConnectionInfo>
<ds:gmpRepoLocation>/COMMON/IVV/GMP/gmpRepoLocation</ds:gmpRepoLocation>
<ds:dfsLocation>/COMMON/IVV/GMP/dfsLocation</ds:dfsLocation>
<ds:maxQueuedRequest>1000</ds:maxQueuedRequest>
<ds:quotas maxQueryPerUser="1" timespan="60000"/>
</ds:dataStore>
```

Example of configuration for RemoteDHuS DataStore:

```
<ds:dataStore
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

```

xsi:type="ds:remoteDhusDataStoreConf" name="IVVRemoteDHuSDataStore" readOnly="true"
priority="150">
<ds:serviceUrl>https://dhus-s2.gael-systems.com/odata/v1</ds:serviceUrl>
<ds:login>ivvuser</ds:login>
<ds:password>password</ds:password>
</ds:dataStore>

```

8.1.9 scanners

[Groups]/ Parameter	Description	Usage
[scanners]/scanner	This section is referred to product upload via FileScanner. FileScanner configuration is automatically created when file scanners are created via UI or OData. It can be update or modified in the dhus.xml when DHuS is not running.	<ul style="list-style-type: none"> • Id: defines the unique scanner identifier • url: it is the path of the source folder where products are retrieved • active: it is the trigger for the ingestion via file scanner • collections: it means if products are ingested in collections

Example:

```

<conf:scanners>
    <scanner:scanner>
        <scanner:id>0</scanner:id>
        <scanner:url>/path/to/dataset/INBOX_folder</scanner:url>
        <scanner:active>true</scanner:active>
        <scanner:collections/>
    </scanner:scanner>
</conf:scanners>

```

8.1.10 synchronizers

[Groups]/ Parameter	Description	Usage
[synchronizers]/synchronizer	This section is referred to the synchronizers configuration. It is automatically created when synchronizers are created via UI or OData. It can be update or modified in the dhus.xml when DHuS is not running.	<ul style="list-style-type: none"> • Id: defines the unique synchronizer identifier • label: it is synchronizer name • serviceUrl: it is the URL SERVICE of the BE from which the synchronization is performed • serviceLogin: username of the user with Archive Manager privileges in the BE from which the synchronization is made • servicePassword: password of the user with Archive Manager privileges in the BE from which the synchronization is made • schedule: schedule of the synchronizer runs • active: trigger of the synchronization process

		<ul style="list-style-type: none"> • created: creation date of the synchronizer • modified: last update of the synchronizer • pagesize: number of product synchronized at each run • remoteIncoming: path of the product source in the BE instance • copyProduct: means if only product metadata are synchronized or the entire .zip file • lastCreated: creation date of the products. It define from which creation date the synchronization starts • skipOnError: it means if a product is skipped or not when a synchronization error occurs
--	--	--

Example:

```
<conf:synchronizers>
    <sync:synchronizer xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="sync:productSynchronizer">
        <sync:id>0</sync:id>
        <sync:label>synch_name</sync:label>
        <sync:serviceUrl>http://172.30.246.78:8081/odata/v1</sync:serviceUrl>
        <sync:serviceLogin>username</sync:serviceLogin>
        <sync:servicePassword>password</sync:servicePassword>
        <sync:schedule>0 0/2 * * * ?</sync:schedule>
        <sync:active>false</sync:active>
        <sync:created>2018-06-27T08:27:52.142Z</sync:created>
        <sync:modified>2018-06-27T12:34:56.744Z</sync:modified>
        <sync:page_size>100</sync:page_size>
        <sync:remoteIncoming>/PUP2/dhus-test/var_IVV/incoming</sync:remoteIncoming>
        <sync:copyProduct>false</sync:copyProduct>
        <sync:lastCreated>2018-06-27T12:15:57.742Z</sync:lastCreated>
        <sync:skipOnError>true</sync:skipOnError>
    </sync:synchronizer>
</conf:synchronizers>
```

8.1.11 evictions

[Groups]/ Parameter	Description	Usage
[evictions]/eviction	This section is referred to the configuration of the Customizable Eviction. It is automatically created when an eviction is created via UI or via OData. It can be update or modified in the dhus.xml when DHuS is not running.	<ul style="list-style-type: none"> • Name: unique eviction name • keepPeriod: it is the trigger of the eviction meaning that it is the minimal time after which products can be evicted • keepPeriodUnit: time unit corresponding to the KeepPeriod property. Default value is DAYS. • maxEvictedProducts: maximum number of products that can be evicted during a single run of the eviction

		<ul style="list-style-type: none"> filter: OData filter used to determine which products can be evicted softEviction: true false. It defines if the eviction is acting only on zip file or also on products metadata status: it's a read only property and shows if the eviction is running or not cron: it defines the schedule of each run of the eviction
--	--	--

Example:

```
<evict:eviction name="Evict_S1_FE" keepPeriod="1" keepPeriodUnit="MINUTES"
maxEvictedProducts="50" filter="startswith(Name, 'S1A')" softEviction="true"
status="STOPPED">
    <evict:cron active="true" schedule="0 0/1 * * * ?"/>
</evict:eviction>
```

8.2 server.xml

The following table summarize the configurable parameters of the `server.xml` file.

[Groups]/ Parameter	Description	Syntax
[Service]/Connector	<p>It is possible to configure several connectors in DHuS, two of them are configured by default. Such connectors define ports on which the application is in listening mode.</p> <p>The first default connector is the one for port 8081, which is the port towards which external requests are directed.</p> <p>The port 30333, configured in the second DHuS connector, is used for internal requests (when OpenSearch requests are sent to DHuS, the application redirect such requests internally towards 30333).</p>	<ul style="list-style-type: none"> port="8081" protocol="org.apache.coyote.http11.Http11NioProtocol" maxConnections="1000" maxThreads="400" keepAliveTimeout="2000" URIEncoding="ISO-8859-1" compression="on" compressionMinSize="1024" compressableMimeType="application/json,application/javascript,application/xhtml+xml,application/xml,text/html,text/xml,text/plain,text/javascript,text/css" port="30333" protocol="org.apache.coyote.http11.Http11NioProtocol" maxConnections="1000" maxThreads="400" keepAliveTimeout="2000" URIEncoding="ISO-8859-1" compression="on"

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

		<ul style="list-style-type: none"> compressionMinSize="1024" compressableMimeType="application/json,application/javascript,application/xhtml+xml,application/xml,text/html,text/xml,text/plain,text/javascript,text/css"
[Service]/Engine/Valve	<p>DHuS keep track of the requests it manages in a Rolling cache.</p> <p>From this section of the server.xml is possible to define which kind of request/connections should be recorded in the cache. The default configuration (reported in the usage here on the right) is such that DHuS record in its rolling cache every request it manages (OData, OpenSearch and UI request). Cache is reset at dhus stop.</p>	<p>Access Filter Settings are:</p> <ul style="list-style-type: none"> pattern: the regular expression to filter user request, i.e. "<code>^.*(/odata/v1/).*\$/</code>" only manages odata request and "<code>^(?!/(home new)/).*\$</code>" consider all request but the UI. useLogger="true false" show or hide the user access in logger output. <p>This setting does impact keeping internal track of the request.</p> <ul style="list-style-type: none"> enable="true false" activate/deactivate the valve.

Example:

```
<?xml version='1.0' encoding='utf-8'?>
<Server port="8005" shutdown="SHUTDOWN">
    <Service name="DHuS-Service">
        <Connector port="8081"
            protocol="org.apache.coyote.http11.Http11NioProtocol"
            maxConnections="1000"
            maxThreads="400"
            keepAliveTimeout="2000"
            URIEncoding="ISO-8859-1"
            compression="on"
            compressionMinSize="1024"

            compressableMimeType="application/json,application/javascript,application/xhtml+xml,application/xml,text/html,text/xml,text/plain,text/javascript,text/css" />
        <Connector port="30333"
            protocol="org.apache.coyote.http11.Http11NioProtocol"
            maxConnections="1000"
            maxThreads="400"
            keepAliveTimeout="2000"
            URIEncoding="ISO-8859-1"
            compression="on"
            compressionMinSize="1024"

            compressableMimeType="application/json,application/javascript,application/xhtml+xml,application/xml,text/html,text/xml,text/plain,text/javascript,text/css" />
        <Engine name="DHuS-Engine" defaultHost="localhost">
            <Host name="localhost" appBase="webapps" deployOnStartup="false" autoDeploy="false">
```

8.3 start.sh

DHuS command line follows Java VM standard command lines. Please find below DHuS internal properties that can be configured.

Parameter	Description & Usage	Optional/ Mandatory
XX:MaxPermSize Xms Xmx	<p>The maximum permanent generation size can be configured with the command-line option <code>-XX:MaxPermSize</code>.</p> <p>The flag <code>Xmx</code> specifies the maximum memory allocation pool for a Java virtual machine (JVM), while <code>Xms</code> specifies the initial memory allocation pool. It is recommended to set these parameters equally to avoid overload for the machine. The configuration of those parameters is depending on the users activity.</p> <p>Example:</p> <pre>-XX:MaxPermSize=1g -Xms24g -Xmx24g</pre>	M
java.library.path	Path to the native libraries of the DHuS distribution.	M
com.sun.media.jai.disableMediaLib	=true to be removed if media jai native library is provided. DHuS does not requires these libraries for optimization.	M
user.timezone	=UTC Mandatory parameter to force the DHuS timezone to a standard, not depending on the operating system settings.	M
sun.zip.disableMemoryMapping	=true Currently mandatory to avoid a crash in zip library usage.	M
java.util.logging.manager	It sets up the java.util.logging to Log4J bridge. <pre>-Djava.util.logging.manager=org.apache.logging.log4j .jul.LogManager \</pre>	M
http.proxyHost http.proxyPort http.nonProxyHosts	Property <code>http.proxyHost</code> defines the host name of the proxy server. Property <code>http.proxyPort</code> defines the port number, the default value being 80.	O

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

	<p>Property <code>http.nonProxyHosts</code> defines a list of hosts that should be reached directly, bypassing the proxy. This is a list of patterns separated by ' '. The patterns may start or end with a '*' for wildcards. Any host matching one of these patterns will be reached through a direct connection instead of through a proxy.</p> <p>Example:</p> <pre>-Dhttp.proxyHost=131.176.235.248 -Dhttp.proxyPort=3128 -Dhttp.nonProxyHosts="172.30.246.*"\</pre>	
<code>java.rmi.server.hostname</code>	<p>Insert this line in the Java options in order to allow clients to invoke methods on the remote object (needed for the cache replication in Scalability 2.0 mode):</p> <pre>-Djava.rmi.server.hostname=<DHuS_VM_IP></pre> <p>Where <code>DHuS_VM_IP</code> is the IP address of the VM hosting the DHuS service.</p>	O
<code>java.rmi.activation.port</code>	<p>Insert this line in the Java options in order to specify the port where peers access each other's cache (needed for the cache replication in Scalability 2.0 mode):</p> <pre>-Djava.rmi.activation.port=40001</pre>	O
<code>sun.rmi.transport.connectionTimeout</code> <code>sun.rmi.transport.tcp.handshakeTimeout</code> <code>sun.rmi.transport.tcp.responseTimeout</code> <code>sun.rmi.transport.tcp.readTimeout</code>	<p>Insert these lines in the Java options in order to set Java RMI runtime time-out times (needed for the cache replication in Scalability 2.0 mode):</p> <pre>-Dsun.rmi.transport.connectionTimeout=15000 \ -Dsun.rmi.transport.tcp.handshakeTimeout=15000 \ -Dsun.rmi.transport.tcp.responseTimeout=15000 \ -Dsun.rmi.transport.tcp.readTimeout=15000 \</pre>	O
<code>country.synonyms</code>	/path/to/file path to the definition of users countries synonyms	O
<code>webapp.excluded</code>	The name of the webapp to not start at system startup (i.e. "fr.gael.dhus.gwt.GWTWebapp")	O
<code>Archive.check</code>	=true false (default=false) Force system check at dhus startup	O
<code>Archive.forceReindex</code>	true false (default=false)	O

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

	Force all the products indexes being re-indexed.	
Archive.incoming.relocate	true false (default=false) Force the relocation of all the products of incoming	O
Archive.incoming.relocate.path	/path/to/relocation (default="") Give the new location path to relocate incoming directory. If no path is provided, incoming will be relocated in its current directory	O
Archive.processings.clean	true false (default=false) Clean all the interrupted processing instead of recover them.	O
force.public	true false (default=false) Force all the product contained into DHuS become public. This is an old function and this parameter is deprecated. Please do not use it.	O
Archive.synchronizeLocal	true false (default=false) Force re-synchronization of local archive path at system startup. This is an old function and this parameter is deprecated. Please do not use it.	O
users.search.notification.force.inactive	true false (default=false) Deactivates all the user search notifications.	O
checkUserConfiguration	true false (default=false) Activates schema aware validation of input xml configuration file	O
dhus.solr.reindex	true false (default=false) Recreate the Solr index from the database	O
action.record.inactive	true false (default=false) Full deactivates read/write statistics	O
dhus.sync.download_attempts	(default=10) During product synchronization. if a download is interrupted and the remote DHuS supports HTTP ranges, the parallelised download manager will attempt to resume the download. number of download attempts (-1 for infinite, must be at least 1)	O

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

dhus.search.innerTimeout	Inner timeout for SolR requests	O
http.timeout.socket http.timeout.connection http.timeout.connection_request	<p>Timeout for the OData synchronizers</p> <p>http.timeout.socket is the socket timeout, it occurs when no data has been received within the given time span, it is also known as SO_TIMEOUT.</p> <p>http.timeout.connection is the connection timeout, it occurs when the remote server does not answer our connection requests.</p> <p>http.timeout.connection_request is a timeout on a request to a connection manager, this is a purely programmatic timeout, and you should probably not set this timeout.</p> <p>The socket timeout is longer because it can occur in the middle of a transfer, we don't want to throw all the data that has already been downloaded to the rubbish if the remote server is not responsive only a few minutes.</p>	O
max.product.page.size	<p>It limits the number of products that can be displayed at once via OData Products entity set and OpenSearch service . Please note that processing quotas shall be deactivated to let this parameter be considered by DHuS.</p> <p>Default:</p> <ul style="list-style-type: none"> • OData have no limitation • OpenSearch has the default limitation to 100 <p>Setting it to "-1", OData and OpenSearch will have no limitation.</p>	O
fr.gael.streams.cache.inactive=true	This parameter is needed to disable the cache mechanism (see Section 7.18)	O
fr.gael.streams.tmpdir="/my/temp/file/location"	location of the temporary cache folder (see Section 7.18)	O
com.sun.management.jmxremote	=true false Set the following system property to allow the JMX client access to the DHuS Java VM. This is needed for SYSMA monitoring feature.	O
com.sun.management.jmxremote.port	It defines the port number through which you want to enable JMX RMI connections. This is needed for SYSMA monitoring feature.	O
com.sun.management.jmxremote	=true false.	O

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

te.ssl	It allows to enable/disable SSL (Secure Sockets Layer) when monitoring remotely. This is needed for SYSMA monitoring feature.	
com.sun.management.jmxremote.authenticate	=true false. It allows to enable/disable the password authentication for remote monitoring. Default values is true. This is needed for SYSMA monitoring feature.	O
com.sun.management.jmxremote.local.only	=true false. It allows to define if to accept or not external connections. Default value is true to avoid any incoming connection except from localhost (127.0.0.1). This is needed for SYSMA monitoring feature.	O

8.4 log4j2.xml

Log4j2 file defines the log level of processes run by the Data Hub, in fact in each logger it is possible to raise/lower the level. The RollingFile section of the `log4j2.xml` defines the name of the application log, its location and its rolling policy.

Log4j2 is a Java-based logging utility now project of the Apache Software Foundation. Here in below follows an overview of the configuration parameters used in the default `log4j2` file. Please refer to <https://logging.apache.org/log4j/2.0/manual/> for configuration details.

[Groups]/ Parameter	Description	Usage
[Properties]/Property	Configuration and definition of the pattern used as template of the DHuS log. The pattern available in the default distribution of the <code>log4j2.xml</code> is such that the log has the following shape: [DHuS_version] [timestamp in UTC] [level] [log message]	<ul style="list-style-type: none"> name=name of the template to be used
[Appenders]/Console	Configuration of the messages logged for <u>standard output</u> (output of SW run) and <u>standard error</u> (SW errors), including <Filters> configuration that allows Log Events to be evaluated to determine if or how they should be published.	<ul style="list-style-type: none"> Name= name of the console (stdout/stderr) Target= location where ConsoleAppender writes its output. Default value is System.out. Patternlayout <ul style="list-style-type: none"> pattern= template used in the standard/stderr output log

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

		<ul style="list-style-type: none"> • ThresholdFilter <ul style="list-style-type: none"> ◦ level= level of messages to be filtered. Anything at or below this level will be filtered out if maxBurst has been exceeded. The default is WARN meaning any messages that are higher than warn will be logged regardless of the size of a burst that is "10" by default ◦ onMatch= Action to take when the filter matches. May be ACCEPT, DENY or NEUTRAL. The default value is NEUTRAL. ◦ onMismatch= Action to take when the filter does not match. May be ACCEPT, DENY or NEUTRAL. The default value is DENY.
[Appenders]/RollingFile	Configuration of the rolling policy for DHuS log file, including policies to define logging criteria, i.e rules defining that all files matching a given file name pattern (filepattern) and older than a certain number of days are rolled.	<ul style="list-style-type: none"> • Name= name of the log file • Filename= location and name of the real time DHuS log • Filepattern= location and name of the rolled log files • Patternlayout <ul style="list-style-type: none"> ◦ pattern= template used in the standard/stderr output log • TimeBasedTriggeringPolicy <ul style="list-style-type: none"> ◦ interval= It is an integer that defines how often a rollover should occur based on the most specific time unit in the date pattern. For example, with a date pattern with hours as the most specific item and an increment of 4 rollovers would occur every 4 hours. The default value is 1. ◦ modulate= true false. It indicates whether the interval should be adjusted to cause the next rollover to occur on the interval boundary.
[Loggers]	Loggers allow using specific log	<ul style="list-style-type: none"> • Name= logger name

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

	<p>levels for classes.</p> <p>Beyond the loggers provided by the default in the <code>log4j2.xml</code>, the following loggers are available for Synchronizers, EvictionServices and DataStores, like in example reported below.</p>	<ul style="list-style-type: none"> Level= the level may be configured with one of TRACE, FATAL, DEBUG, INFO, WARN, ERROR, ALL or OFF. If no level is specified it will default to ERROR. Root= root logger. The root logger does not have the name attribute. The appenders attached are by default the <code>stdout</code> <code>stderr</code> and the <code>RollingFile</code>.
--	--	---

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<Configuration>
    <Properties>
        <Property name="pattern">
>[$ ${sys:fr.gael.dhus.version}] [%d{DEFAULT} {UTC}] [%-5p] %m (%file:%line - %t)%n%throwable
</Property>
    </Properties>
    <Appenders>
        <Console name="stdout" target="SYSTEM_OUT">
            <PatternLayout pattern="${pattern}" />
            <Filters>
                <ThresholdFilter level="DEBUG"/>
                <ThresholdFilter level="WARN" onMatch="DENY"
onMismatch="NEUTRAL"/>
            </Filters>
        </Console>
        <Console name="stderr" target="SYSTEM_ERR">
            <PatternLayout pattern="${pattern}" />
            <Filters>
                <ThresholdFilter level="WARN"/>
            </Filters>
        </Console>
        <RollingFile name="RollingFile" fileName="dhus.log"
filePattern="dhus-%d{yyyy-MM-dd-HH}.log">
            <PatternLayout>
                <Pattern>${pattern}</Pattern>
            </PatternLayout>
            <Policies>
                <TimeBasedTriggeringPolicy interval="6" modulate="true" />
            </Policies>
            <Filters>
                <ThresholdFilter level="DEBUG"/>
            </Filters>
        </RollingFile>
    </Appenders>
    <Loggers>
        <logger name="fr.gael.dhus" level="info"/>
        <logger name="fr.gael.drb.query.FunctionCallExpression" level="debug"/>
        <logger name="org.apache.cxf.jaxrs.utils.JAXRSUtils" level="error"/>
        <logger name="org.apache.solr" level="error"/>
        <logger name="fr.gael.dhus.sync" level="debug"/>
        <logger name="fr.gael.dhus.service.EvictionService" level="debug"/>
        <logger name="org.dhus.store.DataStore.hfs" level="debug"/>
        <Root level="info">
            <AppenderRef ref="stderr" />
        </Root>
    </Loggers>
</Configuration>
```

```

        <AppenderRef ref="stdout" />
        <AppenderRef ref="RollingFile" />
    </Root>
</Loggers>
</Configuration>

```

8.5 suggester.dic

The file `suggester.dic` can be customized with a list of strings to suggest via `/search`. Once restarted, the DHuS suggester can be request via <http://dhusurl/search/suggest/{query}> as shown her below:

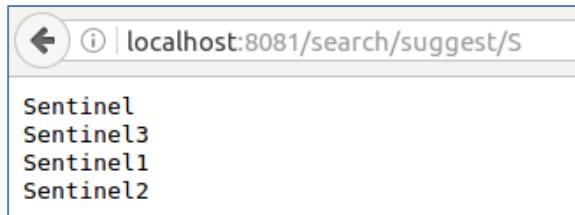


Figure 47 Suggester configuration

8.6 opensearch-description-file.xml

This file describes DHuS OpenSearch interface of Data Hub service. The OpenSearch description document describes the web interface of the DHuS search engine. Please refer to [RD 3] for further details about the OpenSearch description document.

8.7 temporary_files_ehcache.xml

[Groups]/ Parameter	Description	Usage
<code>[ecache] /ehcache</code>	It is the name of the cache manager.	<ul style="list-style-type: none"> • Name: name of the cache manager
<code>[ehcache] /cache</code>	<p>This field defines the properties of the netCDF cache.</p> <p>The following attributes shall be configured only in Scalability 2.0 mode:</p> <ul style="list-style-type: none"> • Eternal; • diskPersistent; • diskExpiryThreadIntervalSeconds; • bootstrapCacheLoaderFactor 	<ul style="list-style-type: none"> • Name: name of the temporary cache • maxEntriesLocalHeap: limits the number of temporary cache files (e.g. 1000) • maxBytesLocalDisk: limits the total size of the cache in bytes (e.g. 10000000000) • Eternal: true false • overflowToDisk: true false • diskPersistent: true false • diskExpiryThreadIntervalSeconds

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

		<ul style="list-style-type: none"> cacheEventListenerFactory bootstrapCacheLoaderFactory
[ehcache]/cacheManagerPeerProviderFactory	<p>This parameter shall be set only for scalability 2.0 mode.</p> <p>It configures the Peer Provider: ehcache provides two mechanisms for peer discovery, manual and automatic.</p> <p>To use one of the built-in peer discovery mechanisms, specify the class attribute as net.sf.ehcache.distribution.RMICacheManagerPeerProviderFactory</p> <p>Set the other attributes in order to configure manual discovery.</p>	<ul style="list-style-type: none"> Class="net.sf.ehcache.distribution.RMICacheManagerPeerProviderFactory" Properties: <ul style="list-style-type: none"> peerDiscovery= manual rmiUrls= <RMI_URLS>: list of all cache peers for all cache listener (detailed via IP address and port) <p>In the rmiUrls attribute use the " " character in order to separate each peer. Do not include the server in the list.</p> <p>Use the character set in the propertySeparator field to separate all these properties (e.g. ",")</p>
[ehcache]/cacheManagerPeerListenerFactory	<p>This parameter shall be set only for scalability 2.0 mode.</p> <p>This field enable the listening from peers to the current cache Manager.</p> <p>The attributes of cacheManagerPeerListenerFactory are:</p> <ul style="list-style-type: none"> class - a fully qualified factory class name properties - comma separated properties having meaning only to the factory. 	<ul style="list-style-type: none"> Class="net.sf.ehcache.distribution.RMICacheManagerPeerListenerFactory" properties <ul style="list-style-type: none"> hostName=<HOST>: hostname/IP of the host the listener is running. If unspecified, the hostname will use InetAddress.getLocalHost().getHostAddress(), which corresponds to the default host network interface. port: port of the host the listener is running remoteObjectPort: port number on which the remote objects bound in <p>Warning: Explicitly setting this to localhost refers to the local loopback of 127.0.0.1, which is not network visible and will cause no replications to be received from remote hosts</p>

		<p>the registry receive calls</p> <ul style="list-style-type: none">o socketTimeoutMillis: TCP/IP Socket timeout when waiting on response (i.e. number of seconds client sockets will wait when sending messages to this listener until they give up). By default this is 2000ms.
--	--	---

Example for DHuS deployed in embedded mode:

```
<ehcache name="TemporaryFilesCacheManager">
    <cache name="TemporaryFilesCache"
          maxEntriesLocalHeap="1000"
          maxBytesLocalDisk="1000000000000">
        <cacheEventListenerFactory
class="fr.gael.streams.TemporaryFilesCacheListener" />
    </cache>
</ehcache>
```

Example for DhuS deployed in Scalability 2.0 mode:

```
<ehcache name="TemporaryFilesCacheManager">
<!-- Auto discovery -->
    <cacheManagerPeerProviderFactory
        class="net.sf.ehcache.distribution.RMICacheManagerPeerProviderFactory"
        properties="peerDiscovery=manual,rmiUrls=/172.30.246.11:40001/TemporaryFile
sCache//172.30.246.46:40001/TemporaryFilesCache"
        propertySeparator=","/>
<!-- create cluster listener on port 40001 -->
    <cacheManagerPeerListenerFactory
        class="net.sf.ehcache.distribution.RMICacheManagerPeerListenerFactory"
        properties="hostName=172.30.246.20,port=40001,remoteObjectPort=9999,socketTimeoutMillis=200
0" />
    <cache name="TemporaryFilesCache" eternal="false" overflowToDisk="false"
          diskPersistent="false" diskExpiryThreadIntervalSeconds="0"
          maxEntriesLocalHeap="1000"
          maxBytesLocalDisk="10000000000"
          memoryStoreEvictionPolicy="FIFO">
        <cacheEventListenerFactory
            class="fr.gael.streams.TemporaryFilesCacheListener" />
        <cacheEventListenerFactory
            class="net.sf.ehcache.distribution.RMICacheReplicatorFactory"
            properties="replicateAsynchronously=true" />
        <bootstrapCacheLoaderFactory
            class="net.sf.ehcache.distribution.RMIBootstrapCacheLoaderFactory"
            properties="bootstrapAsynchronously=true"
            propertySeparator="," />
    </cache>
</ehcache>
```

9. AJS GUI configuration parameters

The DHuS is equipped with an AJS GUI. The files in charge of the GUI configuration management are located in:

- **appconfig.json**, located in [DHUSDIR]/etc/conf
- **styles.json**, located in [DHUSDIR]/var/tomcat/webapps/ROOT/config

9.1 appconfig.json

The configurability of the AJS GUI allows a wide set of configuration actions, which do not need a restart of DHuS to be applied. Due to the growth of the different centres and related installations, a configuration management module has been added into the AJS web app. It allows configuring various aspects of the GUI; mainly it is related to style, texts and layout:

- **Title** (shown in the header bar)
- **Sections visibility** (e.g. Cart, Profile, Sign In, Shapefile etc.)
- **URL and texts of the link logos** (shown in the header panel)
- **Version text** (shown in the info panel)
- **Data Hub Logo** (shown in the info panel)
- **Mission Tags** (shown in the Product List panel)
- **Mission footprint style and color** (shown in the Map panel)
- **Advanced Search Mission specific fields** (shown in Advanced Search Panel)
- **Map Layer** (shown in the Map View)

Please note that all the settings are included in the client side, thus it is possible to change a parameter without restarting the DHuS, but just doing a refresh via browser.

In the next Sections, instructions for specific features configuration are described.

9.1.1 Login badge

Two appconfig.json keys to manage the login badge configurability have been introduced:

- **hide_login_title**: true|false. Parameter that shows/hides login badge title.
- **login_title**: string visualized in the login badge title (if null, the default the string is: "Please login to access our services...")

```
"settings": {  
    "hide_login_title": false,  
    "login_title": ""  
...  
}
```

9.1.2 Missions

The configuration file `appconfig.json` manages mission specific filters. A `missions` section is present in the file, containing an array with the following structure:

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

```
"name": , "indexname": , "indexvalue": , "filters": [filter_array]
```

where [filter_array] is an array of mission-specific filters with the following structure in the case of all the mission except for Sentinel-5P one:

```
"indexname": , "indexlabel": , "indexvalues" , "regex": [OPTIONAL]
```

If a set of value is configured, it appears a combo box containing the list of all specified values.

Here follows an example:

```
"missions": [{

    "name": "Mission: Sentinel-1",
    "indexname": "platformname",
    "indexvalue": "Sentinel-1",
    "filters": [{

        "indexname": "filename",
        "indexlabel": "Satellite Platform",
        "indexvalues": "S1A_*|S1B_*",
        "regex": ".*"
    }, {
        "indexname": "producttype",
        "indexlabel": "Product Type",
        "indexvalues": "SLC|GRD|OCN",
        "regex": ".*"
    }, {
        "indexname": "polarisationmode",
        "indexlabel": "Polarisation",
        "indexvalues": "HH|VV|HV|VH|HH+HV|VV+VH",
        "regex": ".*"
    }
}]
```

9.1.2.1 Sentinel-5P Auxiliary Files filters

In the case of Sentinel-5P Mission the structure of [filter_array] is the following:

```
"indexname": , "indexlabel": , "complexindexes":
```

Here follows an example:

```
"filters": [
    {
        "indexname": "producttype",
        "indexlabel": "Input Auxiliary Files for Product Type",
        "complexindexes": [
            {

```

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

```

        "label": "FRESCO",
        "value": "(AUX_MET_TP OR AUX_MET_2D OR AUX_NISE__)"
    }, {
        "label": "AER_LH",
        "value": "(AUX_MET_TP OR AUX_MET_2D OR AUX_NISE__ OR L2_FRESCO OR L2_NP_BD6 OR L2_AER_AI)"
    }, {
        "label": "CO____",
        "value": "(AUX_MET_TP OR AUX_MET_QP OR AUX_MET_2D OR AUX_CTM_CO OR AUX_CTMCH4)"
    }, {
        "label": "CH4____",
        "value": "(AUX_MET_TP OR AUX_MET_QP OR AUX_MET_2D OR AUX_CTM_CO OR AUX_CTMCH4 OR L2_FRESCO OR L2_NP_BD6 OR L2_NP_BD7 OR L2_CO____)"
    }, {
        "label": "O3_PR",
        "value": "(AUX_MET_TP OR AUX_MET_2D OR AUX_NISE__ OR L2_FRESCO OR L2_AER_AI)"
    }, {
        "label": "O3_TPR",
        "value": "(AUX_MET_TP OR AUX_MET_2D OR AUX_NISE__ OR L2_FRESCO OR L2_AER_AI OR L2_O3_PR)"
    }, {
        "label": "SO2____",
        "value": "(AUX_MET_TP OR AUX_MET_QP OR AUX_MET_2D OR AUX_NISE__ OR AUX_CTMFCT OR AUX_CTMANA OR AUX_BGSO2_ OR L2_AER_AI OR L2_O3____ OR L2_CLOUD)"
    }, {
        "label": "CLOUD_",
        "value": "(AUX_MET_TP OR AUX_MET_QP OR AUX_MET_2D OR AUX_NISE__ )"
    }, {
        "label": "O3____",
        "value": "(AUX_MET_TP OR AUX_MET_QP OR AUX_MET_2D OR AUX_NISE__ OR L2_CLOUD)"
    }, {
        "label": "HCHO____",
        "value": "(AUX_MET_TP OR AUX_MET_QP OR AUX_MET_2D OR AUX_NISE__ OR AUX_CTMFCT OR AUX_CTMANA OR AUX_BGHCHO OR L2_AER_AI OR L2_CLOUD)"
    }, {
        "label": "AER_AI",
        "value": "AUX_MET_2D"
    }, {
        "label": "NO2____",
        "value": "AUX_MET_2D"
    }
}

```

```

        "value": "(AUX_MET_2D OR AUX_NISE__ OR AUX_CTMFCT OR L2__FRESCO OR
L2__AER_AI)"
    }, {
        "label": "O3_TCL",
        "value": "L2__O3____"
    }],
    "regex": ".*"
}
]
}

```

9.1.3 First node inspection

The first node inspection in the product details dialog can be managed by means of `regex_closed_by_default` parameter to be added in the settings section of the `etc/conf/appconfig.json` file.

```

"settings": {
    "regex_closed_by_default": "(.*)",
...
}

```

The value of the parameter should be a javascript regular expression with the identifier of products whose inspection is to be avoided. If all products must be inspectable when showing product details, there is no need to add the configuration parameter.

Here follow some examples:

1. no products inspectable at panel opening:

```
"regex_closed_by_default": "(.*)",
```

2. all products inspectable at panel opening:

```
"regex_closed_by_default": "",
```

3. no products inspectable for a specific mission at panel opening, e.g. Sentinel-5P:

```
"regex_closed_by_default": "^(S5P.*),
```

4. no products inspectable for several missions at panel opening, e.g. Sentinel-1, Sentinel-2A:

```
"regex_closed_by_default": "^(S1.*|^S2A.*),
```

9.1.4 Shapefile

The configuration parameter `enable_shapefile` of the file "appconfig.json" enable/disable the shapefile usage for area selection.

The following properties can be configured for the shapefile management:

- `shapefile_description`: description of shapefiles that can be uploaded;

- `shapefile_max_size`: max file size supported;
- `shapefile_max_size_message`: error messages visualized in case of too large shape file upload;
- `max_points_shape_file`: max number of points in the polygon used by DHuS to re-construct the AOI.

Here after a sample of `appconfig.json` with the configuration for the parameters listed above:

```
"enable_shapefile": true,
"shapefile_description": "You can also drag & drop the shapefile on map. Only files with extension .shp are supported. Shapefiles are limited to one record of type POLYGON and must be no larger than 5.00 MB.",
"shapefile_max_size": 5242880,
"shapefile_max_size_message": "Your shapefile cannot be uploaded because it is too large. Shapefiles must be no larger than 5.00 MB.",
"max_points_shape_file": 50
```

The configured shapefile description is displayed clicking on the "Hamburger Button" in the left corner of the Advanced Search Panel:

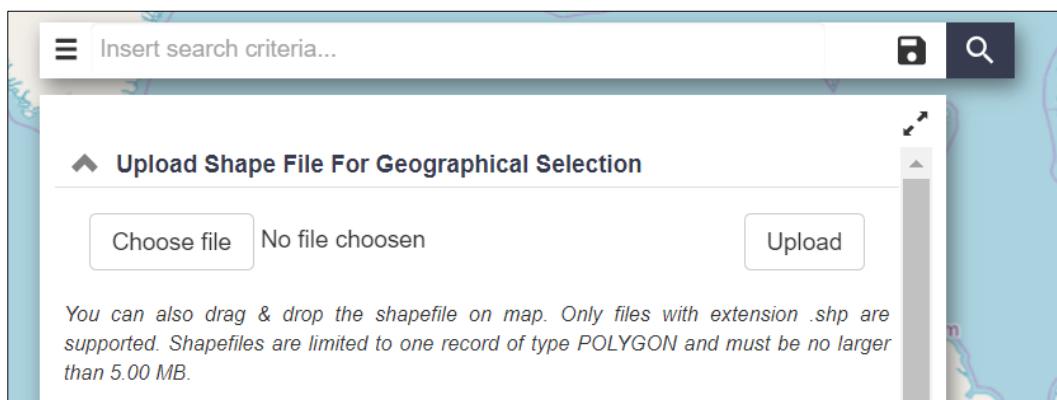


Figure 48 Shapefile description display

9.1.5 Terms and conditions link

The T&C link is configurable through the "`t_and_c`" parameter in the "settings" section.

9.1.6 List pagination

It is possible to configure the pagination limit via the parameter "`pagination_limit`". The recommended configuration is [25, 50, 75, 100, 125, 150].

Please note that configuring this parameter is mandatory and, since AJS uses the SolR engine, the pagination limit cannot exceed the row limitations configured for SolR (please note that the parameter defining the maximum number of rows retrievable via OpenSearch is `max.product.page.size` of the start.sh).

9.1.7 Deletion cause

AJS GUI configuration file allows the configuration of deletion causes with the following parameter:

```
"settings": {  
    ...  
    "deleted_product": { "show_deletion_cause": true, "causes": ["Invalid: bad  
geometry", "Invalid: reprocessed"], "default_cause": "Invalid" },  
    ...  
}
```

Here follows the explanation about the mentioned line:

- `deleted_product.show_deletion_cause` (true|false) : if true the prompt message to insert deletion cause is shown, if false the prompt message is not shown
- `deleted_product.causes`: is the array containing available deletion causes.
- `deleted_product.default_cause`: contains the default cause used when end user inserts a deletion cause not matching with one of the available deletion causes

9.1.8 Map polygon selection/pan management

The management of the map pan function and polygon selection has been upgraded:

- The UI presents one button on the up-right of the map. This button allows the activation of the pan functionality and the draw selection functionality when the user clicks on it.
- If the button is set on "Navigation Mode", the user is able to move the map using both left and right button of the mouse.
- If the button is set on "Area Mode", the user is able to:
 - draw a polygon clicking on the left button of the mouse (each click draws a vertex) and closing the polygon on double click;
 - draw a squared polygon when clicking and shifting the mouse on the map (holding the click);
 - move the map with the right button of the mouse.

The foreseen default mode for the Map is the "Navigation Mode" ("draw_by_default": false), but it is possible to switch into "draw" mode setting the following parameter in the appconfig.json:

```
{  
    ...  
    "settings": {  
        ...  
        "draw_by_default": true,  
        ...  
    }  
}
```

On "Area Mode" mode, the polygon drawing starts only if no selection area is present on the map, otherwise the click will clear the map.

Please note that the old configuration parameter "show_map_toolbar" has been removed, so it is no possible to restore the map toolbar via configuration file.

9.1.9 Map layers attributions configurability

It is possible to configure map layers attributions in order to display them in the lower right corner of the map, they can be added in the appconfig.json file.

The attribution of each layers are the following:

- Terrain:

```
Terrain: { Data © <a href="http://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors and <a href="#data">others</a>, Rendering © <a href="https://github.com/mapserver/basemaps">MapServer</a> and <a href="http://eox.at">EOX</a> }
```

- Overlay:

```
Overlay: { Data © <a href="http://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors, Processing & Rendering © <a href="https://github.com/mapserver/basemaps">MapServer</a> and <a href="http://eox.at">EOX</a> }
```

- OpenStreet:

```
Open Street: { Data © <a href="http://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors, Rendering © <a href="https://github.com/mapserver/basemaps">MapServer</a> and <a href="http://eox.at">EOX</a> }
```

- Sentinel-2 Cloudless

```
Sentinel-2 Cloudless: { Data © Contains modified Copernicus Sentinel data 2016 & 2017 }
```

- Blue Marble

```
Blue Marble { Data © <a href="http://nasa.gov">NASA</a> }
```

Map layers attributions can be added in the appconfig.json file in the following way (one for each layer's source):

```
"Road": {
    "sources": [
        {
            "class": "TileWMS",
            "attributions": "Open Street: { Data © <a href="http://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors, Rendering © <a href="https://github.com/mapserver/basemaps">MapServer</a> and <a href="http://eox.at">EOX</a> }",
            "params": {
                "url": "https://tiles.esa.maps.eox.at/wms",
                "wrapX": false,
                "params": { "LAYERS": "osm_3857", "TILED": true }
            }
        ],
        "title": "Open Street",
```

```
        "type": "base",
        "visible": true
    }
```

The configured attributions are displayed in the lower right corner of the map as Figure 49 (corresponding to the described example):

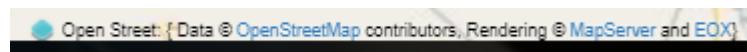


Figure 49 Map layer attribution display

9.1.10 Configuration of more than three map layers

It is possible to configure more than three map layers via `appconfig.json` file (as per the schema reported below).

Examples of map layers are Satellite, Hybrid, Road, Humanitarian, etc.

```
"map": {
    "Satellite": {
        ...
    },
    "Road": {
        ...
    },
    "Hybrid": {
        ...
    },
    "Humanitarian": {
        ...
    }
}
```

9.1.11 Map layer selector restyling

The map layer panel presents a preview of the configured layers, centered on central Italy.



Figure 50 Map layer panel

9.1.12 Sorting criteria for datatake sensing start and tile id

AJS GUI configuration file allows the configuration a sorting criteria for datatake sensing start and tile id. This sorting criteria is meaningful only for Sentinel-2 L1C products.

Here follow an example on how configure this criteria:

```
"sortOptions": [
{ ...},
{
    "name": "TileId",
    "value": "datatakesensingstart,hv_order_tileid",
    "message": "This sorting is applicable only for Sentinel-2 L1C products. Clicking on this checkbox, only Sentinel-2 L1C products will be shown in the list.",
    "filter": "platformname:Sentinel-2 AND producttype:S2MSI1C",
    "show_checkbox": true,
    "checkbox_checked": false,
    "disable_filters": true
}
]
```

The following parameters are managed within the "sortOptions" section of the `appconfig.json` file:

- filter: (String) a configurable filter associated to the selected sorting. This filter allows to show in the list only products for which the sorting is meaningful (e.g. tileid or cloudcoverpercentage)
- message: (String) a configurable message shown when selecting the sorting criteria explaining the effects of the selected sorting
- show_checkbox: (Boolean) a configurable parameter used to show a checkbox to the left of the message. Selecting or deselecting the checkbox, the filter associated to the sorting criteria is applied/removed
- checkbox_checked: (Boolean) a configurable parameter used to set as "checked" by default the checkbox shown at the left of the message. Obviously this parameter has sense only if show_checkbox parameters is defined and equals to true
- disable_filters: (Boolean) a configurable parameter used to disable the mission specific filters.

9.1.13 LTA management

The "enable_lta" configuration parameter for enabling/disabling LTA management is present in the `appconfig.json` file. The default value is `true` (LTA management enabled). Change it to `false` if you want to disable LTA management:

```
{
...
"settings": {
...
"enable_lta": true,
...
}
}
```

When the property "enable_lta" is set to "true", the LTA management is enabled and offline products are labelled as OFFLINE on AJS UI and their background in the product list is grey. In addition, messages for the download attempts and overquotas are displayed.

9.1.14 Enlarging thumbnail

The parameter "hidePopoverList" in the setting section of the appconfig.json file allows enabling the popover feature to enlarge thumbnail on mouseover.

Here follows the explanation about the mentioned line:

- "hidePopoverList": false makes the popover as active. This is the default behaviour, i.e. if the parameter is not present, thumbnails will enlarge at the mouse crossing.
- "hidePopoverList": true makes the popover feature disabled.

```
{
  ...
  "settings": {
    "hidePopoverList": false,
  ...
  }
}
```

9.2 styles.json

This configuration file allows configuration of product footprints layout, according to the product instrument.

Parameter	Description	Syntax
matches	Product Instrument	<ul style="list-style-type: none"> • name: it defines the group title • value: instrument name. It defines the flag that appears in the UI near each product name in the product list. Refer to Figure 51.
styles	RGB colour for footprint	<ul style="list-style-type: none"> • fill <ul style="list-style-type: none"> ◦ color= it defines the RGB colors for the footprint fill • stroke <ul style="list-style-type: none"> ◦ color= it defines the RGB colors for the footprint borders ◦ width= it defines the thickness of the footprint borders
selected_styles	RGB colour for footprint selection	See the configuration parameters for the <code>styles</code> field above.
highlighted_styles	RGB colour for highlighted footprint	See the configuration parameters for the <code>styles</code>

	i.e. when the mouse passes over them	field above.
label_style	Background color of the flag defining the mission that appears in the UI near each product name in the product list.	background-color: mission background color specified in HEX format.
instrlabel_style	Background color of the flag defining the instrument that appears in the UI near each product name in the product list. Color should be set in HEX format.	background-color: instrument background color specified in HEX format.

Example:

```
[
  {
    "matches": [
      {
        "name": "Instrument",
        "value": "SAR-C"
      }
    ],
    "styles": {
      "fill": {
        "color": [
          179,
          27,
          27,
          0.1
        ]
      },
      "stroke": {
        "color": [
          179,
          27,
          27,
          1
        ],
        "width": 1.5
      }
    },
    "selected_styles": {
      "fill": {
        "color": [
          255,
          0,
          0,
          0.3
        ]
      },
      "stroke": {
        "color": [
          255,
          0,
          0,
          0.3
        ]
      }
    }
  }
]
```

```

        0,
        0,
        1
    ],
    "width": 3.5
}
},
"highlighted_styles": {
    "fill": {
        "color": [
            179,
            27,
            27,
            0.3
        ]
    },
    "stroke": {
        "color": [
            179,
            27,
            27,
            1
        ],
        "width": 3
    }
},
"label_style": "background-color: #dc143c !important;",
"instrlabel_style": "background-color: #dc143c !important;"}
}
]

```

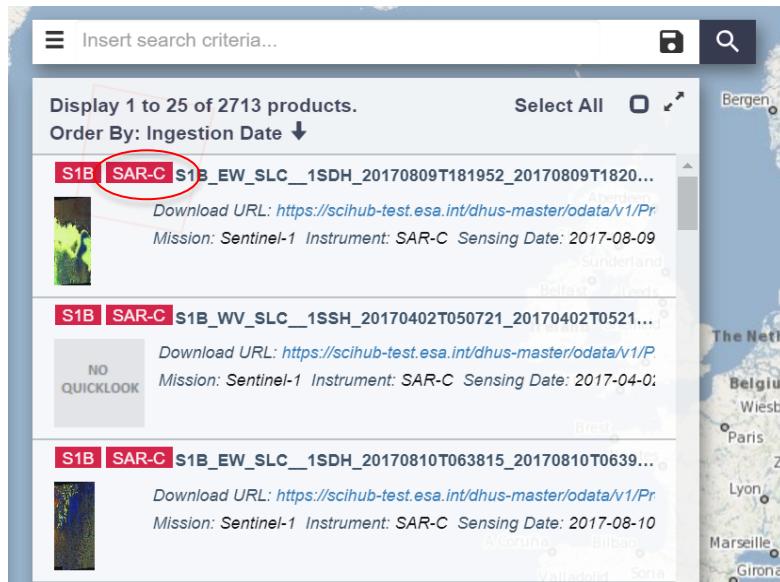


Figure 51 Instrument label as defined in the `value` field of the `styles.json` file.

10. OWC configuration parameters

The files in charge of the OWC configuration management are located in:

[DHUSDIR]/var/tomcat/webapps/new/conf

These files are:

- **appconfig.json**
- **styles.json**
- **menu.json**
- **theme.json**

In the following paragraphs the configurable parameters of these files are described.

10.1 appconfig.json

The following table summarize the configurable parameters of the `appconfig.json` file.

Parameter & Description	Synthax
Debug mode	<code>"debug": false,</code>
Show Auxiliary data file feature	<code>"adf": false,</code>
Calendar Setting	<code>"calendar_refresh_time": 360, "show_calendar": true,</code>
Max number of instances per panel pushed in the the navigation manager (main layout).	<code>"max_instances_per_panel": 3,</code>
Show signup page	<code>"show_signup": true,</code>
Term and conditions page link	<code>"t_and_c_link": "https://scihub.copernicus.eu/twiki/do/view/SciHubWebPortal/TermsConditions",</code>
Show the link to open the legacy user interface (AJS client)	<code>"show_oldgui_link": true, "oldgui_link": "...", "oldgui_label": "go to classic view",</code>
Default map layer (See elow)	<code>"default_layer": { "Road": { "sources": [{ "class": "OSM", "params": { "url": " http://tile2.openCycleMap.org/transport/{z}/{x}/{y}.png", "wrapX": false</code>

	<pre> }], "title": "Road", "type": "base", "visible": ":visible" } }, </pre>
Logos to show in the help page	<pre> "dashboard_logos": ["/images/serco.png", "/images/gael.png", "/images/copernicuslogo.png", "/images/logo_esa.png", "/images/european_union_emblem.png"], "timeline": { "styles": { "S1": "#dc143c", "S2": "#0f9d58", "S3": "#0000ff" } }, </pre>
Size of list page	"page_size":25

Here below we describe additional features:

10.1.1 Flex layout templates

Here below an example showing as to configure flex layout templates:

```

"component_layout": [
    {
        "name": "empty_layout",
        "icon": "./images/grey-layout/01_layout.png",
        "preview": "./images/black-layout/01_layout.png",
        "component": "empty-template"
    },
    {
        "name": "twocolumns_layout",
        "icon": "./images/grey-layout/02_layout.png",
        "preview": "./images/black-layout/02_layout.png",
        "component": "twocolumns-template"
    },
    {
        "name": "tworows_layout",
        "icon": "./images/grey-layout/03_layout.png",
        "preview": "./images/black-layout/03_layout.png",
        "component": "tworows-template"
    },
    {
        "name": "square_layout",
        "icon": "./images/grey-layout/04_layout.png",
        "preview": "./images/black-layout/04_layout.png",
        "component": "square-template"
    },
    {
        "name": "trilements_layout",
        "icon": "./images/grey-layout/05_layout.png",
        "preview": "./images/black-layout/05_layout.png",
        "component": "trilements-template"
    }
],

```

```

    "component": "trilements-template"
  },
  "name": "trilements2_layout",
  "icon": "./images/grey-layout/06_layout.png",
  "preview": "./images/black-layout/06_layout.png",
  "component": "trilements2-template"
]

```

10.1.2 WMS - Map servers configuration in the Client

The configuration of the map servers has been added in the new section "wms_server_config", that contains some parameters about the map server URL/credentials and the mapping rule between products and mapservers.

Currently the mapping rule is based on regular expression in order to find any possible pattern based on the product name.

An example of configuration could be:

```

"wms_server_config": [
  {
    "name": "Serco WMS",
    "url": "http://149.202.246.167/instance00/ows",
    "username": "",
    "password": "",
    "rule": "S3",
    "priority": "1"
  }
]

```

This component allows the configuration of a list of WMS server, specifying the following parameters:

- server name, i.e. server identifier
- url, i.e. wms request url
- username, i.e. username of the account enabled to perform WMS request (if needed)
- password, i.e. password of the account enabled to perform WMS request (if needed)
- rule, i.e. a regular expression on product identifier used to perform WMS request only for products matching the rule. This parameter is configurable by means of free text regular expression or simplified regular expression.

NOTE: the parameter "priority", so far, is not implemented (on 0.14.4-X).

Warning: the current implementation is a proof of concept, with the purpose of testing feature feasibility, so there is not yet a full configurability of this feature, since only a sample WMS with S3 OLCI layers is supported.

To deactivate the WMS interface delete the WMS block mentioned above.

10.1.3 Metadata plugin definition

Here below an example on how to configure metadata plugin:

```
"processor_plugins": [ {
```

```
        "name": "Products size average",
        "visualizer": "average-visualizer",
        "webworkers": ["processor-plugins/001.js"]
    }
```

Where:

- name: name of the plugin to show in the plugin selector
- visualizer: the web component to show the outcomes of the plugin
- webworkers: array with the url where are located the web workers to call for the specific plugin.

10.1.4 ADF feature

Auxiliary Data File (ADF) Search is a new feature provided by the DHuS and available in OWC Application. Auxiliary Data File Search fills the need of using the DHuS not only to disseminate products but also to inform the user on the availability of new information, such as new Auxiliary Data Files (ADFs), which can be used to re-process products with a better quality.

ADF feature is disabled by default. In order to enable it, setting `adf` property to true, as reported in the example below.

```
{
    "debug": false,
    "adf": true,
...
...
}
```

The configuration files for ADF component are present in two subfolders (rules, task_tables) contained in the etc folder of DHuS installation directory:

- ./rules/OL_1_EO_SelectionRules.properties
- ./rules/OL_1_RAC_SelectionRules.properties
- ./rules/OL_1_SPC_SelectionRules.properties
- ./rules/OL_2_SelectionRules.properties
- ./rules/SL_1_SelectionRules.properties
- ./rules/SL_2_SelectionRules.properties
- ./rules/SR_1_CAL_SelectionRules.properties
- ./rules/SR_1_MEAS_SelectionRules.properties
- ./rules/SR_2_SelectionRules.properties
- ./rules/SY_1_2_SelectionRules.properties
- ./rules/SY_2_VG_SelectionRules.properties
- ./task_tables/OL_1_EO_TaskTable.xml
- ./task_tables/OL_1_RAC_TaskTable.xml
- ./task_tables/OL_1_SPC_TaskTable.xml

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

- ./task_tables/OL_2_TaskTable.xml
- ./task_tables/SL_1_TaskTable.xml
- ./task_tables/SL_2_TaskTable.xml
- ./task_tables/SR_1_CAL_TaskTable.xml
- ./task_tables/SR_1_MEAS_TaskTable.xml
- ./task_tables/SR_2_TaskTable.xml
- ./task_tables/SY_1_2_TaskTable.xml
- ./task_tables/SY_2_VG_TaskTable.xml

10.2 styles.json

This file allows managing product footprint colours according to the product instrument.

Refer to section 9.2 for details on the configuration.

10.3 menu.json

The file menu.json contains the main menu sections definition.

Parameter & Description	Syntax
Array containing the sections of the main menu.	"sections": [...]

Every section is an object with the following parameters:

Parameter & Description	Syntax
The title of the section	"title": "section title"
The web-component to visualize when the user clicks on the button associated with this section in the main menu	"component": "login-component"
The icon to show for the section	"icon": "search"
Width of the panel that contains this component	"width": "150px"

10.4 theme.json

The file theme.json contains the global appearance customization for the owc application.

Parameter & Description	Syntax
Title of the application	"title":"title"
Url of the logo image	"logourl":"http://www.example.it/image.png"
Application panels Background color	"bgcolor":"#e9f4ff"
Background image url	"bgimageurl":"image.png"

11. Configuration converter script

A script named “updateConfiguration.sh” for converting the dhus.xml file of previous DHuS version is included in the distribution.

Starting for this version is needed to run such a script before starting the DHuS and before starting a migration process. Indeed this script will convert the dhus.xml file coming from previous DHuS versions in order to make it compliant with the current configuration implementation. In particular the `<system:archive><system:archive/>` section will be automatically modified in order to remove, if present, the OldIncoming configuration and convert the OldIncoming storage in the related HFSDatStore storage. All the product paths of previous ingested/synchronized products will be checked and make compliant with the current configuration schema.

In order to run the script it is needed to launch the following command in the machine where DHuS is installed:

```
./updateConfiguration.sh /path_to_DHUS_inst_folder/etc/dhus.xml
```

Appendix A. Example of dhus.xml file coming from distribution

```

<!DOCTYPE configuration [
    <!ENTITY varFolder "local_dhus">
]>
<!--
    DHuS configuration
    workingDir working directory path. default is "" which leads the DHuS to use a folder
    "dhus" created in the default temporary-file directory (specified by the system property
"java.io.tmpdir")
-->
<configuration workingDir="" xmlns="fr.gael.dhus.database.object.config"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:ds="org.dhus.store.datastore.config"
    xmlns:cron="fr.gael.dhus.database.object.config.cron"
    xmlns:msg="fr.gael.dhus.database.object.config.messaging"
    xmlns:network="fr.gael.dhus.database.object.config.network"
    xmlns:product="fr.gael.dhus.database.object.config.product"
    xmlns:search="fr.gael.dhus.database.object.config.search"
    xmlns:server="fr.gael.dhus.database.object.config.server"
    xmlns:system="fr.gael.dhus.database.object.config.system"
    xmlns:scanner="fr.gael.dhus.database.object.config.scanner"
    xmlns:evict="fr.gael.dhus.database.object.config.eviction"
    xmlns:sync="fr.gael.dhus.database.object.config.synchronizer"
    xmlns:source="fr.gael.dhus.database.object.config.source">
<!-- Cron configuration. They all have two parameters:
    - active: defines if the cron is currently active or not
    - schedule: defines the schedule of the cron.
        A schedule is configured according to this pattern:
            Seconds Minutes Hours Day-of-month Month Day-of-week [Year]
        You can find more information on http://www.quartz-scheduler.org/documentation/quartz-1.x/tutorials/TutorialLesson06
        Here are some simple examples:
            - "0 0 */1 * * ?": every hour.
            - "0 0 9-17 ? * MON-FRI": on the hour nine to five week days.
            - "0 0 0 25 DEC ?": every Christmas Day at midnight
            - "0 0 3 ? * *": every day at 3 AM -->
<crons>
    <!-- Cron used to synchronize local archive, defined in system/archive/@path -->
    <cron:archiveSynchronization active="false" schedule="0 0 0/5 ? * *"/>
    <!-- Cron used to clean database, like removing old statistics or old not confirmed
users. -->
    <cron:cleanDatabase active="true" schedule="0 0 1 ? * *">
        <!-- Definition of the time (in days) for user to confirm its registration -->
        <cron:tempUsers keepPeriod="10" />
        <!-- Definition of the time (in days) system is keeping statistics -->
        <cron:statistics keepPeriod="90" />
    </cron:cleanDatabase>
    <!-- Cron used to dump database -->
    <cron:dumpDatabase active="true" schedule="0 0 3 ? * *"/>
    <!-- Cron used to clean database dumps.
        - keep: defines how dumps are stored. -->
    <cron:cleanDatabaseDump active="true" schedule="0 0 4 ? * *" keep="10" />
    <!-- Cron used to execute user saved filescanners.
        - sourceRemove: defines if found products shall be removed from source -->
    <cron:fileScanners active="true" schedule="0 0 22 ? * *" sourceRemove="false"/>
    <!-- Cron used to execute user saved searches and send results to users -->
    <cron:searches active="true" schedule="0 0 5 ? * *"/>
    <!-- Cron used to send system logs.
        - addresses: logs recipients addresses. They shall be coma-separated -->
    <cron:sendLogs active="true" schedule="0 55 23 ? * *" addresses="dhus@gael.fr" />

```

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

```

<!-- Cron used to check all system coherence, including database optimization -->
<cron:systemCheck active="true" schedule="0 30 23 ? * *"/>
</crons>
<messaging>
    <!-- Mail configuration. These values are used only at first launch of
        the system. They shall be modified in Management Panel if needed.
        - onUserCreate: defines if system send mail when creating user
        - onUserUpdate: defines if system send mail when updating user
        - onUserDelete: defines if system send mail when deleting user -->
    <msg:mail onUserCreate="true" onUserUpdate="true" onUserDelete="true">
        <!-- Mail server definition.
            - smtp: server address
            - port: server port
            - tls: defines if server is using TLS protocol
            - username and password : connection information -->
        <msg:server smtp="smtp.xxx.xx" port="587" tls="false" username="dhus@xxx.xx"
password="password">
            <!-- Information used in "from" part of sent mails.
                - name: displayed name of "from" part
                - address: displayed address of "from" part -->
            <msg:from name="DHuS Support Team" address="dhus@xxx.xx" />
            <!-- Defines the "reply to" address of sent mails -->
            <msg:replyTo>dhus@xxx.xx</msg:replyTo>
        </msg:server>
    </msg:mail>
</messaging>
<!-- Network configuration -->
<network>
    <network:outbound>
        <network:channel name="PriorityChannel" weight="75">
            <network:classifier>
                <network:includes>
                    <network:include>
                        <network:userEmailPattern>.*@gael.fr</network:userEmailPattern>
                    </network:include>
                </network:includes>
            </network:classifier>
        </network:channel>
        <network:channel name="SelfRegisteredChannel" weight="25">
            <network:defaultUserQuotas>
                <!-- maxConcurrent defines the maximum simultaneous accepted
                    transfers. -->
                <network:maxConcurrent>5</network:maxConcurrent>
                <!-- maxCount defines the maximum number of accepted transfers on
                    a customizable period.
                    "periodUnit" attribute defines the unit of the period. Possible
                    units are "DAYS", "HOURS", "MICROSECONDS", "MILLISECONDS",
                    "MINUTES", "NANOSECONDS", "SECONDS".
                    "period" attribute is a sliding time window used to count
                    number of transferred product from now to the past period delay. -->
                <network:maxCount period="1" periodUnit="DAYS">50</network:maxCount>
                <!-- Maximum accepted size of transfer file. -->
                <network:maxSize>10737418240</network:maxSize> <!-- 10Gb -->
                <!-- maxCumulativeSize defines the maximum cumulated accepted
                    transfers size on a customizable period.
                    period/periodUnit attributes are defined in maxCount description. -->
                <!-- <maxCumulativeSize period="7"
periodUnit="DAYS">1099511627776</maxCumulativeSize> -->
                <!-- maxBandwidth maximum bandwidth authorized for this channel. -->
                <network:maxBandwidth>13107200</network:maxBandwidth> <!-- 100Mb/s -->
            </network:defaultUserQuotas>
        </network:channel>
    </network:outbound>
    <network:inbound>
        <network:channel name="PriorityChannel" weight="75">

```

```

<network:classifier>
    <network:includes>
        <network:include>
            <network:userEmailPattern>.*@gael.fr</network:userEmailPattern>
        </network:include>
    </network:includes>
</network:classifier>
</network:channel>
<network:channel name="SelfRegisteredChannel" weight="25">
    <network:defaultUserQuotas>
        <!-- maxConcurrent defines the maximum simultaneous accepted
            transfers. -->
        <network:maxConcurrent>5</network:maxConcurrent>
        <!-- maxCount defines the maximum number of accepted transfers on
            a customizable period.
            "periodUnit" attribute defines the unit of the period. Possible
            units are "DAYS", "HOURS", "MICROSECONDS", "MILLISECONDS",
            "MINUTES", "NANOSECONDS", "SECONDS".
            "period" attribute is a sliding time window used to count
            number of transferred product from now to the past period delay. -->
        <network:maxCount period="1" periodUnit="DAYS">50</network:maxCount>
        <!-- Maximum accepted size of transfer file. -->
        <network:maxSize>10737418240</network:maxSize> <!-- 10Gb -->
        <!-- maxCumulativeSize defines the maximum cumulated accepted
            transfers size on a customizable period.
            period/periodUnit attributes are defined in maxCount description. -->
        <!-- <maxCumulativeSize period="7"
periodUnit="DAYS">1099511627776</maxCumulativeSize> -->
        <!-- maxBandwidth maximum bandwidth authorized for this channel. -->
        <network:maxBandwidth>13107200</network:maxBandwidth> <!-- 100Mb/s -->
    </network:defaultUserQuotas>
</network:channel>
</network:inbound>
</network>
<!-- Products configuration -->
<products>
    <!-- Download configuration
        - compressionLevel: is a value to build the Deflater, from 1 to 9.
        The compression level defines how rigorously the compressor looks
        to find the longest string possible. As a general rule of thumb:
        - Compressing at the maximum level (9) requires around twice as
        much processor time as compressing at the minimum level (1)
        - For typical input, compressing at the maximum as opposed to
        the minimum level adds around 5% to the compression ratio.
        0 value means no compression.
        - checksumAlgorithms: shall be coma-separated -->
<product:download compressionLevel="4" checksumAlgorithms="MD5"/>
    <!-- Quicklook calculation parameters
        - height: height of generated quicklooks
        - width: width of generated quicklooks
        - cutting: allow system to cut image when processing quicklooks -->
<product:quicklook height="512" width="512" cutting="false" />
    <!-- Thumbnail calculation parameters
        - height: height of generated thumbnails
        - width: width of generated thumbnails
        - cutting: allow system to cut image when processing thumbnails -->
<product:thumbnail height="64" width="64" cutting="false" />
</products>
<!-- Search configuration -->
<search>
    <!-- Geocoder configuration
        - url: geocoder url -->
    <search:geocoder url="http://nominatim.openstreetmap.org">
        <!-- Nominatim geocoder configuration
            - boundingBox: defines if the geocoder is querying only

```

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

```

the bounding box of the matching place from the Nominatim Web
Service i.e. the four corners encompassing the place.
Otherwise, it will query the complete polygon boundaries, that
may have lower performance according to the number of vertices
composing the place's boundaries.
- maxPointNumber: maximum number of points that can be returned
for a polygon -->
<search:nominatim boundingBox="false" maxPointNumber="50" />
<!-- Geoname geocoder configuration
    - username: username used to connect to Geoname -->
    <search:geoname username="username" />
</search:geocoder>
<!-- Odata configuration.
    - defaultTop: default top used by Odata when none is given -->
<search:odata defaultTop="50" />
<!-- Solr configuration.
    path: solr path
    core: solr core name
    schemaPath: solr schema path. Shall be empty.
    synonymPath: path of solr synonyms file -->
    <search:solr path="&varFolder;/solr" core="dhus" schemaPath="" synonymPath="" />
</search>
<!-- Server configuration.-->
<server>
    <!-- External url (protocol://host:port/path) is the url viewed by users
        Used in case of an apache proxy redirection for example
        Empty values mean that server values are those which are viewed by users -->
    <server:external protocol="" host="" port="-1" path="" />
</server>
<!-- System configuration -->
<system>
    <!-- Definition of principal administrator user.
        If User exists, DHuS will give him all rights at launch, but will not
        his password. This shall be done in Management panel of GUI.
        If User is not existing, DHuS will create it with defined password.
        -->
    <system:administrator name="root" password="password" />
    <!-- Definition of local archive path. -->
    <system:archive errorPath="" />
    <!-- Definition of database path and where dumps are stored -->
    <system:database JDBCDataSource="org.hsqldb.jdbcDriver"
hibernateDialect="org.hibernate.dialect.HSQLDialect" dumpPath="&varFolder;/database_dump"

JDBCURL="jdbc:hsqldb:file:&varFolder;/database/hsqldb;shutdown=true;hsqldb.write_delay=false;
hsqldb.cache_size=1000000;hsqldb.cache_rows=200000;hsqldb.log_size=50;hsqldb.tx=mvcc;hsqldb.default_table_type=cached;hsqldb.nio_max_size=20480;hsqldb.large_data=true;" login="sa"
password="" />
    <!-- Definition of system long name and short name. -->
    <system:name long="Data Hub Service" short="DHuS" />
    <!-- Processing configuration.
        corePoolSize: defines maximal number of active threads. Default is 1 -->
    <system:processing corePoolSize="4" />
    <!-- Definition of support name and mail.
        - registrationMail: used to send the administrative registration information. If
this
        field is not set, DHuS is using support mail.
        These values are used only at first launch of the system.
        They shall be modified in Management Panel if needed. -->
    <system:support name="DHuS Support Team" mail="dhus@gael.fr"
registrationMail="dhus@gael.fr" />
    <!-- Definition of tomcat path -->
    <system:tomcat path="&varFolder;/tomcat" />
    <!-- Executor (background service that executes synchronizers)
        It must be enabled if you want to use the synchronisation feature.

```

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

```

batchMode: the executor will run the synchronizers until there is no more to
synchronize. -->
    <system:executor enabled="false" batchModeEnabled="false" />
    <!-- Evicted products will be copied in the trashPath repository -->
    <system:trashPath path="" />
</system>

<!-- DataStores Configuration -->
<dataStores>
    <!-- All DataStores share some attributes:
        name: Name of this dataStore (must be unique)
        readOnly: If this dataStore is read-only or read-write (default: read-
write(false))
        priority: Priority used to sort DataStores. A datastore with the lowest value will
answer first (default: 100) -->

    <!-- HFS DataStore configuration -->
    <ds:dataStore xsi:type="ds:hfsDataStoreConf"
        name="synchronized-hfs-without-copy"
        readOnly="true">
        <!-- Path on disk to this HFS DataStore -->
        <ds:path>/remote/incoming/path</ds:path>
        <!-- Maximum number of sub-folder in one folder (default: 10, at least 5)-->
        <ds:maxFileNo>10</ds:maxFileNo>
        <!-- Maximum number of items in one folder (default: 1024, at least 10)-->
        <ds:maxItems>1024</ds:maxItems>
    </ds:dataStore>

    <!-- OpenStack DataStore configuration -->
    <ds:dataStore xsi:type="ds:openStackDataStoreConf"
        name="secondaryDataStore"
        readOnly="false">
        <!-- Name of the provider -->
        <ds:provider>openstack-swift</ds:provider>
        <!-- Identity is tenantName:Username -->
        <ds:identity>my-identity</ds:identity>
        <!-- Password to access the Openstack storage -->
        <ds:credential>my-credential</ds:credential>
        <!-- Url of the Openstack storage -->
        <ds:url>authentication-service-url</ds:url>
        <!-- Target container on the Openstack storage -->
        <ds:container>container-name</ds:container>
        <!-- Region of the target container on the Openstack storage -->
        <ds:region>RegionOne</ds:region>
    </ds:dataStore>

    <!-- DHuS as DataStore -->
    <ds:dataStore xsi:type="ds:remoteDhusDataStoreConf" name="RemoteDHuS">
        <!-- URL to the OData V1 service endpoint -->
        <ds:serviceUrl>http://REMOTE_DHUS/odata/v1</ds:serviceUrl>
        <!-- Valid credentials on the remote DHuS -->
        <ds:login>root</ds:login>
        <ds:password>password</ds:password>
    </ds:dataStore>
</dataStores>
<scanners></scanners>
<synchronizers>
    <sync:synchronizer xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="sync:smartProductSynchronizer">
        <sync:id>0</sync:id>
        <sync:label>example</sync:label>
        <sync:serviceUrl>_deprecated_</sync:serviceUrl>
        <sync:serviceLogin>_deprecated_</sync:serviceLogin>
        <sync:servicePassword>_deprecated_</sync:servicePassword>
        <sync:schedule>0 0/3 *-* * ? *</sync:schedule>
    </sync:synchronizer>
</synchronizers>

```

Sentinels Rolling Archive User Access, Operations, Maintenance and Evolutions

```

<sync:active>false</sync:active>
<sync:created>2018-04-27T09:24:29.467Z</sync:created>
<sync:modified>2018-05-03T13:24:03.249Z</sync:modified>
<sync:pageSize>5</sync:pageSize>
<!-- Timeout -->
<sync:timeout>300000</sync:timeout>
<!-- Download attempts (per source) -->
<sync:attempts>10</sync:attempts>
<!-- Minimal download bandwidth (per download) before trying to switch source --
->
<sync:threshold>204800</sync:threshold>
<!-- Synchronizer source list -->
<sync:sources>
    <sync:source>
        <!-- Source id reference -->
        <sync:sourceId>0</sync:sourceId>
        <!-- Last creation date synchronizer index -->
        <sync:lastCreated>2018-04-23T15:59:06.732Z</sync:lastCreated>
    </sync:source>
    <sync:source>
        <sync:sourceId>1</sync:sourceId>
        <sync:lastCreated>1970-01-01T00:00:00.000Z</sync:lastCreated>
    </sync:source>
</sync:sources>
</sync:synchronizer>
</synchronizers>
<evictions>
    <!-- Eviction configurations.
        name: name of the eviction
        maxEvictedProducts: the maximum number of evicted products when running an
        eviction
        keepPeriod: the minimal time between product creation and eviction
        keepPeriodUnit: the time unit for the above value (default is DAYS)
        filter: OData filter used to query products to evict
        orderBy: OData ordering clause used to query products to evict
        targetCollection: name of the collection owning products to evict
        softEviction: boolean value (false : physical product and metadatas are deleted --
        true : metadatas are kept)
        status: STOPPED or STARTED, to activate or stop and eviction
    -->
    <evict:eviction name="DefaultEviction" maxEvictedProducts="1000" keepPeriod="10" />
</evictions>
<sources>
    <source:source>
        <!-- Source id -->
        <source:id>0</source:id>
        <!-- Source URL -->
        <source:url>http://dhusSource0/odata/v1</source:url>
        <!-- Source authentication username -->
        <source:username>usr_source_0</source:username>
        <!-- Source authentication password -->
        <source:password>pwd_source_0</source:password>
        <!-- Max concurrent downloads -->
        <source:maxDownload>5</source:maxDownload>
    </source:source>
    <source:source>
        <source:id>1</source:id>
        <source:url>http://dhusSource1/odata/v1</source:url>
        <source:username>usr_source_1</source:username>
        <source:password>pwd_source_1</source:password>
        <source:maxDownload>2</source:maxDownload>
    </source:source>
</sources>
</configuration>

```



<END OF DOCUMENT>