D5.3 First report on maintenance and integration of federation and collaboration services

Lead Partner:	KIT
Version:	2
Status:	Under EC review
Dissemination Level:	Public
Document Link:	https://documents.egi.eu/document/3503

Deliverable Abstract

The document outlines the first report on maintenance and integration of federation, access enabling and collaboration services, one of the key components of the EOSC Federating Core, a fundamental asset that EOSC-hub provides to EOSC. It provides a technical description of enhancements for the EOSC-Hub services in Work Package 5 (WP5), results of integration activities and collaboration work with other initiatives made during the first year of the project. The report identifies the integration gaps and elaborates the plans for WP5 services.

COPYRIGHT NOTICE

DELIVERY SLIP

Date	Name	Partner/Activity	Date
From:	Nicolas Liampotis	GRNET	
	Kostas Koumantaros	GRNET	
	Themis Zamani	GRNET	
	Roksana Rozanska	CYFRONET	
	Cyril L'orphelin	IN2P3	
	Pavel Weber	KIT	
	Marcus Hardt	KIT	
	Jie Yuan	KIT	
	David Vincente	BSC	
	Marious Chatziangelou	IASA	
	Alexandros Nakos	IASA	
	Ivan Diaz Alvarez	CESGA	
	Jens Jensen	STFC	
	Greg Corbett	STFC	
	Raphael Ritz	MPG	
	Daniel Kouril	CESNET	
	Christos Kanellopoulos	GEANT	
	Sander Apweiler	Julich	
	Mischa Salle	Nickef	
	Slavik Licehammer	CESNET	
	Enrico Vianello	INFN	
Moderated by:	Malgorzata Krakowian	WP1/EGI.eu	
Reviewed by:	Malgorzata Krakowian	WP1/EGI.eu	
	Diego Scardaci	WP10/EGI.eu	10.07.2019
Approved by:	AMB		

DOCUMENT LOG

Issue	Date	Comment	Author
v.0.1	18.02.2019	Table of Content ready	Pavel Weber
v.0.2	2.04.2019	The first draft with all sections ready	
v.03	17.05.2019	All contributions are provided	WP5 service/tool owners
v.0.4	21.05.2019	Added executive summary	Pavel Weber
v.0.5	16.06.2019	Ready for review	

v.0.6	10.07.2019	Review	Diego Scardaci
v.0.7	22.07.2019	Corrections from review are implemented	Pavel Weber
v.1.0	04.09.2019	Submitted to EC	Malgorzata Krakowian
v.2	16.01.2020	Ready for AMB review	Pavel Weber

TERMINOLOGY

https://wiki.eosc-hub.eu/display/EOSC/EOSC-hub+Glossary

Terminology/Acronym	Definition
AAI	Authorization and Authentication Infrastructure
AARC	Authentication and Authorisation for Research and Collaboration
AC	Attribute Certificate
AppDB	Applications Database
AppDB IS	AppDB Information Service
AppDB VMOps	AppDB VM Operations
AUP	Acceptable Use Policies
BDII	Berkeley Database Information Index
CA	Certification Authority
CDI	Collaborative Data Infrastructure
CMDB	Configuration Management Database
DPMT	Data Project Management Tool
EGI	European Grid Infrastructure
EOSC	European Open Science Cloud
EUDAT	European Data Infrastructure
GDPR	EU General Data Protection Regulation
GGUS	Global Grid User Support
GOCDB	Grid Operations Configuration Management Database
HA	High Availability
IAM	Identity and Access Management System
IdP	Identity Provider
IGTF	Interoperable Global Trust Federation
LB	Load Balancing
OIDC	OpenID Connect
OLA	Operational Level Agreement
PKIX	Public-Key Infrastructure (X.509)
SLA	Service Level Agreement
StaR	Storage Accounting Record
SOCRM	Service Order and Customer Relationship Management
PID	Persistent Identifier
SP	Service Provider
SMS	Service Management System
SAML	Security Assertion Markup Language
TRL	Technology Readiness Level
VM	Virtual Machine
VO	Virtual Organisation
VOMS	Virtual Organization Membership Service

Contents

Executive summary

EOSC-hub contributes to EOSC by providing the central Hub - an integration and management system with a set of services, policies and procedures - acting as a central entry point for researchers to discover, access and use the EOSC resources.

The federation, access enabling, and collaborative services managed and maintained in Work Package 5 (WP5) are one of the key components of the EOSC Federating Core, a fundamental asset that EOSC-hub provides to EOSC. They guarantee the operation of the EOSC Hub and support the integration of generic/common (WP6), thematic (WP7) services, as well as any new onboarded service of research communities or academic infrastructures in the EOSC ecosystem. It is worth to mention that some of these tools have been adopted by the EOSC Portal, like the EOSC white-label AAI, based on EGI Check-in technology, and the EOSC Hub Service Catalogue and Marketplace.

WP5 focus is maintenance and constant enhancement of the federation and collaborative services their adoption to the changing technical requirements of research communities and other stakeholders. In collaboration with WP10, WP5 also fosters the definition of interoperability guidelines enabling tools, offering the same or similar functions but operated by different providers, to jointly work, in an integrated manner, in the EOSC scenario. Furthermore, WP5 supports the EOSC federated service management system (SMS) providing the tools to support and automate the main SMS processes, enabling an efficient service management in the large EOSC federated environment of multiple independent providers.

This document provides an overview of achievements made in integration of federation, access enabling and collaborative services during the first year of the project, it describes the maintenance work and outlines evolved roadmaps and future plans for the WP5 services. Below a summary of major achievements during the first project year is given.

The EOSC Hub Authentication and Authorization Infrastructure (AAI) enables seamless access to EOSC Hub services and resources based on the solutions delivered by EGI, EUDAT and INDIGO-DataCloud. During the first reporting period, the AAI development and integration strategy based on the recommendations defined in the AARC project[1] has been finalized. The main focus was put on the initial integration of AAI solutions, aimed at demonstrating technical ability for communities using either B2ACCESS or Check-in for their community AAI to use services behind the EGI and EUDAT e-Infrastructure SP Proxies. The future plans in AAI area include the improvement of this initial integration between Check-in and B2ACCESS as well as the interconnection of others EOSC Hub AAI services, i.e. GEANT eduTEAMS and INDIGO IAM, and the finalisation of the remaining harmonisation activities on the technical and policy level with the final aim to enable a real single sign-on on the EOSC resources and services.

The EOSC Hub Order Management System (OMS) comprises several central components including Service Catalogue and Marketplace (MP), Service Portfolio Management Tool (SPMT), Service Order Management Back Office (SOMBO) and integrates many other systems to facilitate promotion, discovery, access and ordering of the productional EOSC Hub services. It aims to support Service

---

[1] https://aarc-project.eu/

Order and Customer Relationship Management (SOCRM) process by providing technical vehicle to accomplish objectives defined by this process. The major achievements during the first reporting period in this area is related to strategic enhancements of the Marketplace Platform, extending it with new functionalities, its integration with EOSC Portal, enabling of ordering of services from EOSC service catalogue, initial integration of the Marketplace with SOMBO. Apart from the Marketplace also other components of OMS were actively developed. The SOMBO component has been developed to provide facilities to manage orders registered in Marketplace and distribute them to the resource and service providers. By the end of the first year the development of SOMBO has not been accomplished. As originally planned, the module is expected to be put in production during the second year of the project. The integration of SPMT with OMS according to initial integration plan and evolving requirements of Service Portfolio Management process has been started.

The components of the EOSC Hub Configuration Management Database (CMDB) have been enhanced to meet the requirements of the Configuration Management process. The production instance of GOCDB has become accessible over IPv6. The architecture of the GOCDB has been reviewed and the implementation of EOSC Hub centric dashboard to represent EOSC's topology has been started. The Data Project Management Tool (DPMT) has implemented API to allow machine agents to invoke actions via REST calls. The DPMT has been integrated in the EOSC Hub Accounting system. The SVMON Web UI has been enhanced and the service has been integrated with DPMT, GOCDB and Pakiti client.

The Data Management Planning Tool consists of a front-end component OpenDMP with a web UI, through which data management plans are managed and monitored, and a back-end service eestore that collects and provides information from a variety of data service registries. The eestore service has undergone a rearrangement of the API to make things easier to extend to include new registries. Throughout the year, new sources have been added to the service enabling the eestore to provide a more extensive set of information to openDMP.

EOSC Hub monitoring system is based on ARGO, which is a flexible and scalable framework for monitoring status, availability and reliability of services provided by infrastructures. Several enhancements have been performed in ARGO compute engine to enable flexible definition of SLA targets and customer defined thresholds related to service availability and reliability. The implementation of a service management interface for infrastructure managers to configure the monitoring service to their needs and to manage the monitoring probes has been started. A new Web UI has been developed and deployed, which retrieves data from multiple sources and provides unified EOSC-hub view for monitoring information.

The ARGO Messaging Service (AMS) enables reliable asynchronous messaging for the EOSC Hub infrastructure, provides the transport layer for multiple systems and facilitates scalable and uniform integration of various services which rely on it. The Argo Messaging Service (AMS) has been enhanced by the introduction of a number of new API calls based on user requirements. Another notable enhancement of the service, during the first year of the project, was adding support for translating x509 certificates to AMS tokens to allow/ease the on-boarding of the remainder users of the old Message Broker Network to the new AMS Service.

The Accounting Repository stores compute (serial and parallel jobs), storage, and cloud resource usage data collected from Resource Centres of the EGI and EUDAT infrastructures. Accounting information is gathered from distributed sensors into a central Accounting Repository where it is processed to generate summaries that are available through the Accounting Portal. The initial integration of EUDAT storage accounting system with Accounting repository has been successfully accomplished according to the integration plan. Several enhancements have been performed on Accounting Portal including initial integration with EOSC-hub AAI on the development instance, development of new EOSC-hub instance. Further work on mapping of the internal EUDAT storage metrics into storage accounting record format and uniform visualization of the accounting data of all EOSC Hub resources in Accounting Portal is in progress.

The Helpdesk service xGUS for EOSC Hub has been integrated with the helpdesk systems of the main infrastructures EGI and EUDAT under the umbrella of EOSC-hub project. This integration permits the movement of tickets between the infrastructures and keeping the history of the ticket in the xGUS front-end. In addition, service providers onboarded through the EOSC Portal can decide to adopt xGUS as their main helpdesk of integrating it with their own helpdesks. This allows the EOSC users to have a unique EOSC contact point, the xGUS helpdesk, but the ticket handling and response can be delivered by the specific provider helpdesk's support teams (EUDAT, EGI or any other providers that joined the EOSC).

The Applications Database (AppDB) is a central service that maintains information about software solutions and virtual appliances (VAs). In addition, AppDB is responsible for distributing the registered VM images to the resource providers and enabling users to deploy and manage Virtual Machines to the EGI Cloud infrastructure. The AppDB VMOps has been integrated with the GGUS ticketing service and provides a channel for users to communicate their issues to cloud provider administrators in order to resolve them. A graphical interface has been made available, which lets users create a ticket within the GGUS system, addressing a specific VM or a topology of VMs. AppDB's platform also got extended with support for persistent identifiers (PIDs) for registered digital objects (software & Virtual Appliances) using open standards, such as the HANDLE system.

In the context of collaboration with OpenAIRE-Advance, the Application Database was planned to exchange information about relevant registered products with OpenAIRE's catalogue. To this purpose, a new OAI-PMH server interface has been developed, which exposes metadata about registered software items and their releases, as well as virtual appliances and versions thereof, to the OpenAIRE harvesting service

1   Introduction

This report of work package 5 provides a summary of integration results for EOSC Hub federation and collaborative services, their enhancements and maintenance activities which have been accomplished during the first project year. The federation and collaborative services deliver the main functions of the EOSC Hub, enable access to EOSC Hub eco system and often referred to as access enabling or as a core EOSC-hub services. They are considered one of the key components of the EOSC Federating Core.

The structure of the document is defined as follows: after a brief description of the high level architecture of the EOSC federating services, each section provides a summary of enhancements, maintenance activities and future development plans for the services in the corresponding WP5 task, followed by section, which outlines the integration activities. For each integration activity the summary of achievements made during the first year of the project, identified deviations from the initial integration plan, identified integration gaps and future plans are given.

2 High-level architecture of the EOSC Hub federating services

The high-level architecture of the major EOSC Hub core services and their components is shown in Figure 2-1. The EOSC Hub AAI layer, shown as a blue box, enables uniform federated access to EOSC Hub core services, which is shown by the blue line, which depicts the integration of the core services with the AAI.
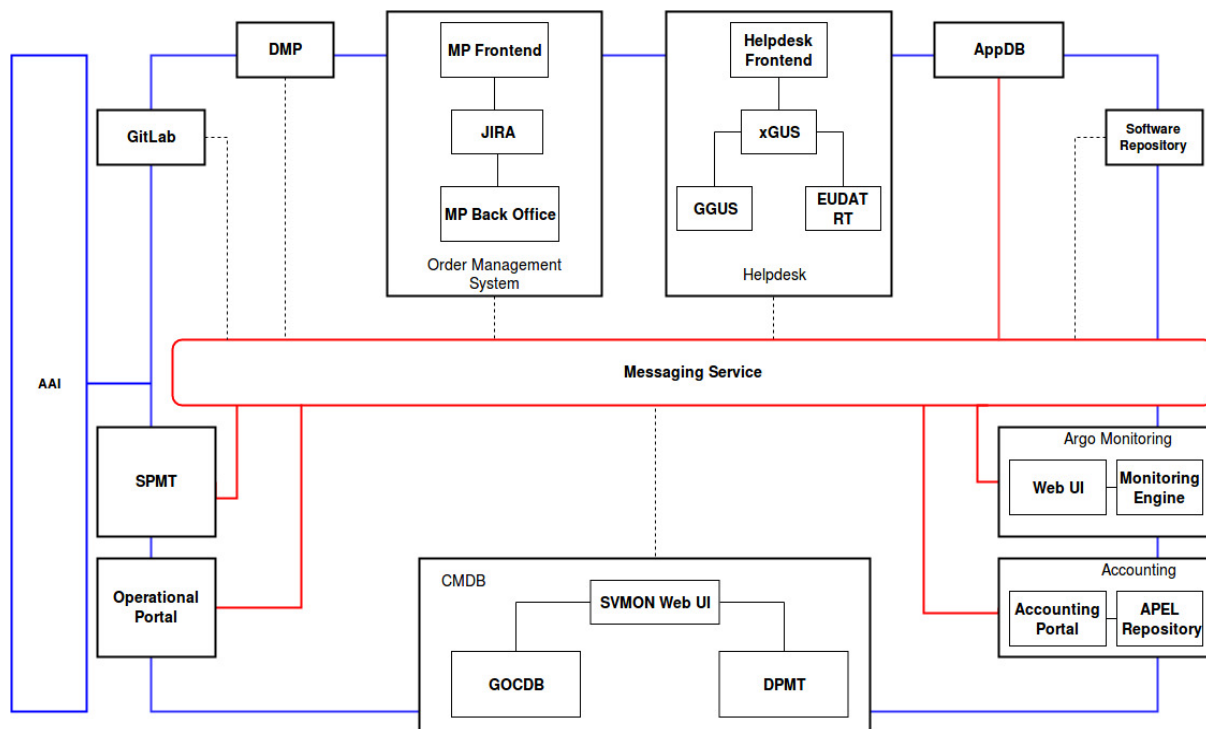

Figure 2-1. High-level architecture of EOSC Hub core services

The main objective of WP5 is to integrate many different services, portals, platforms provided by different e-infrastructures creating an ecosystem of tools able to support the daily operations of the EOSC Hub. There are many dependencies between these services, detailed later in this document, and some of them were already integrated following point-to-point integration patterns while others leveraged on the Messaging Service for asynchronous communication. The first approach has shown many limitations and drawbacks like building the tight dependencies between systems, complex integration patterns based on the unique protocols and interfaces. These limitations are eliminated by the second approach, the adoption of Messaging Service as a uniform transport layer across the core EOSC Hub infrastructure, which has been identified as the target for the tool-to-tool communication. A HTTP interface provided by the Messaging Service allows the robust, scalable integration of independent applications and services using the Publish/Subscribe Model. Further details about Messaging Service are given in Section 6.5 of this document. As shown in Figure 2-1, the Messaging Service interconnects main core services allowing loose coupling at the system/interface level. Red connectors on the picture depict already existing integrations of various services with Messaging Service, while the dotted ones represent missing interconnections that are planned to be implemented. It must be mentioned that the full implementation of the architecture

shown in Figure 2-1 is still in progress and requires further adoption and definition of common data model, which is being defined. The progress on implementation of this architecture will be reported in further deliverables.

3 Identification, Authentication, Authorisation and Attribute Management

3.1 Overview

The EOSC Hub AAI builds on existing AAI solutions that follow the architectural and policy recommendations defined in the AARC project [R1]. Solutions from EGI, EUDAT, GEANT and INDIGO that have successfully delivered a portfolio of operational services with Technology Readiness Levels (TRL) above TRL 7 in this field over the last years are the initial basis of the EOSC Hub AAI. These AAI solutions connect to eduGAIN as service providers, but act as identity providers from the services point of view, thereby allowing users to use their credentials from their home organisations. Complementary to this, users without an account on a federated institutional Identity Provider are still able to use social media or other external authentication providers for accessing services. Thus, access can be expanded outside the traditional user base, opening services to all user groups including researchers, people in higher-education, and members of business organisations.

Research communities can leverage the EOSC Hub AAI services for managing their users and their respective roles and other authorisation-related information. At the same time, the adoption of standards and open technologies, including SAML 2.0, OpenID Connect, OAuth 2.0 and X.509v3, facilitates interoperability and integration with the existing AAIs of other e-Infrastructures and research communities.

A high-level view of the EOSC Hub AAI is provided in Figure 3-1. The EOSC Hub AAI comprises different SP-IdP-Proxy services each of which acts as a bridge between the community-managed
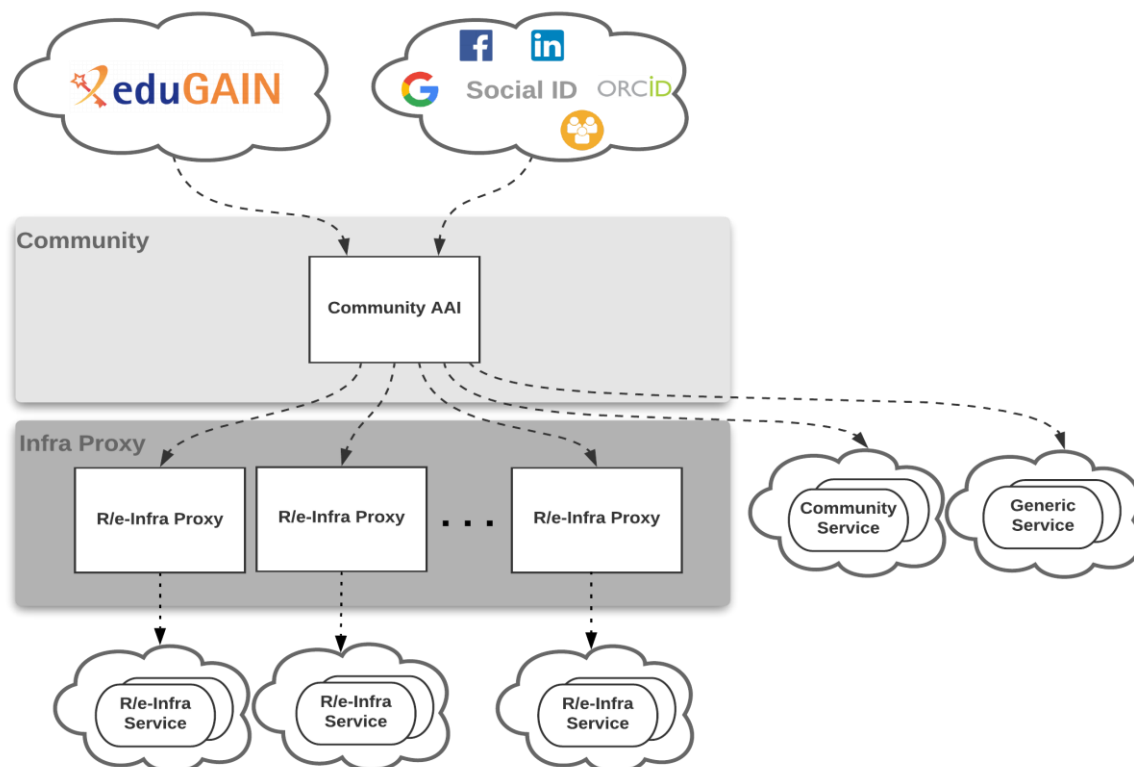

Figure 3-1 High-level view of EOSC-hub AAI architecture

proxies managing the researchers' identity and the generic EOSC-hub e-infra services. Each community proxy in turn serves as a bridge between external identity providers and the proxies to the e-infrastructure services. This is the "community-first" approach to the AARC Blueprint Architecture, which enables researchers to sign in with their community identity via their Research Community AAI, e.g. B2ACCESS, Check-in, IAM, as well as GÉANT eduTEAMS as a result of the collaboration agreement with the GEANT4-2 project [R2]. Community-specific services are connected to a single Research Community AAI while e-Infra services are connected to a single e-infra AAI service gateway. Lastly, generic services (e.g. RCauth.eu Online CA) may be connected to more than one Research Community AAI proxies.

3.2   B2ACCESS

A detailed description of the B2ACCESS service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

3.2.1 Maintenance activities

Within the last year several versions of Unity [R5], the underlying technology of B2ACCESS [R6], were tested. During the tests the software was investigated for issues in operation and issues in usability. An example for operational issues was the handling of multiple external identity providers (IdP) configured by the metadata of the IdPs, which caused problems, as not all IdPs were usable for the users. The release containing this issue was skipped and did not go into production. An example for a usability issue was the sorting of IdPs in the login screen by ASCII sort (first upper cases A-Z, then lower cases A-Z), which is not intuitive for users. This release was skipped too. Some other releases were skipped because new releases were available before all tests had been done. Beside the skipped releases of Unity, two updates to new versions were made, which passed all tests in the last year. In addition to updating the used software, we updated the configuration to disable insecure cipher suites. We removed obsolete service providers which were no longer in use from B2ACCESS. With this action, the authentication of users was no longer possible and potential security issues within B2ACCESS were closed.

Another aspect in addition to the maintenance of Unity itself is the maintenance of the operating system. New software versions were installed regularly, and security patches installed immediately. We also removed insecure cipher suites from other software components such as the web and SSH servers.

Within 2018, the server certificate and the SAML signing certificate reached their end of life. The certificates were renewed and all administrators of connected identity and service providers, using the SAML protocol, were informed about the new certificate and asked to update their local configuration. This action was scheduled on a Friday afternoon to avoid many errors for users due to certificate mismatch.

3.2.2 Summary of service enhancements

The organisation name is one of the mandatory attributes for users in B2ACCESS. This attribute can be released only by home organisation's identity providers which are part of eduGAIN [R7] (federation of home organisation IdPs), but even the most identity providers, who are part of eduGAIN, do not release this information. The social or community identity providers do not know or release the organisation name of their users. In order to reduce the number of attributes provided by the users themselves during the registration, we extract the organisation name from the scoped affiliation. In case the identity provider does release the organisation name, this information is stored and the organisation name is not extracted from the affiliation attribute. With this enhancement, we were able to reliably extract the organisation name for users who are authenticated by their home organisation's identity provider (home IdP). If the home IdP does

release the full set of Research & Scholarship (R&S) attributes, users do not need to provide any further information themselves.

In March, we started the integration between B2ACCESS and Check-In [R8]. Check-In was added as an external identity provider on the one hand, and as a service provider on the other hand. The same was done for B2ACCESS in Check-In. This integration allows users of services connected with B2ACCESS to authenticate with their Check-In account and users of services connected with Check-In to authenticate with their B2ACCESS account. In the following months we harmonised the exchange of user information between B2ACCESS and Check-In to a common set of attributes to reduce the user interaction in the login procedure. More information about the integration is available in Section 3.10.1.

3.2.3 Future plans

In the future, we plan several enhancements in usability. The enhancements are in the integration with Check-In, the UI and the group management.

We want to reduce the user interaction in the integration with Check-In. With a common AUP, the users do not need to accept an additional AUP, if they sign into B2ACCESS with their Check-In account. The account/email validation should be removed in this integration, too. We will trust that the account/the email address was verified during the external account creation at Check-In. Having these enhancements, there is almost no additional user interaction needed to use B2ACCESS and connected services with the Check-In account.

After the integration with Check-In in the SAML context, we plan to extend it to the OAuth/OIDC context. At the moment, B2ACCESS can only verify its own token, if a service requests the token verification. In future we plan that the token is verification against Check-In, if the token was not generated by B2ACCESS. If the validation against Check-In was successful, the services, who requested the validation, will get a positive response. If the check against Check-In fails, the service will get a negative response, too. The extension of token verification will allow services connected to one AAI to interact with services connected to the other AAI using the user tokens received during the authentication. The services remain connected to a single AAI, but are able to interoperate with services from other e-infrastructure. The users will be able to use multiple services with one login and do not need to create additional passwords or tokens to connect the different services manually.

A new user interface in the login screen will simplify the selection of external identity provider by the users. The Check-In service will be more highlighted and a user can select it with one click. Returning users will be offered the same external identity provider that was used for their last authentication.

A new group management endpoint will allow the group management in a simplified user interface. Group administrators will be able to create and remove subgroups, invite or remove collaborators, accept or reject group membership requests and grant or revoke group management privileges to other group members. This endpoint allows communities to do the membership management by themselves and enhances authorisation decisions for connected services, based on group membership of the users.

3.3   Check-in

A detailed description of the Check-in service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

3.3.1   Maintenance activities

The production operation of the EGI Check-in service involved technological upgrades of the underlying framework and libraries in order to take advantage of new features and robustness, as well as continuous optimisation of the architecture and automation of new tasks to ensure the uninterruptible and performant operation.

3.3.2   Summary of service enhancements

3.3.2.1   Enhancements to service provider interface

- The syntax for expressing group membership and role information adopted the URN scheme specified in AARC guidelines AARC-G002 [R9]. Depending on the federated identity protocol, this information is represented as follows:
  - in the case of SAML, using the multi-valued eduPersonEntitlement attribute, as defined in eduPerson [R10]
  - in the case of OpenID Connect/OAuth2, using the multi-valued eduperson_entitlement claim, as defined in eduPerson [R10] following the naming conventions specified in [R11].
- The OAuth 2.0 Service Provider (SP) interface added support for the Token Exchange protocol [R12]. This is an extension of OAuth 2.0 that enables clients to request and obtain security tokens from authorisation servers, including security tokens employing impersonation and delegation. These new tokens might be access tokens that are more narrowly scoped for the downstream service compared to the original tokens that they were exchanged for.
- The OpenID Connect/OAuth2 SP interface added support for the nonstandard OAuth2 scopes defined in [R13] for requesting eduPerson related claims.
- The issuer (iss) claim has been added to the token introspection response.
- The OpenID Connect/OAuth2 SP interface added support for configuring the default and max validity time of issued tokens.

3.3.2.2   *User interface improvements*

A new theme, based on Bootstrap, was created and applied to the following interfaces of Check-in:

- IdP Discovery (supports searching (multilingual) through eduGAIN IdPs and provides dedicated login buttons for selected IdPs (e.g. for Social Identity Providers, ORCID and connected Community AAIs such as B2ACCESS and ELIXIR)
- User information release page
- OpenID Connect/OAuth2 client management interface

The new theme includes a cookie banner and supports adding a corner ribbon with customisable text for allowing end-users to distinguish between the different environments of Check-in (e.g. production, staging, development).

Lastly, the "Manage Services" page was added to allow users to view and revoke active access/refresh tokens that have been issued for OpenID Connect/OAuth2 clients.

3.3.3   **Future plans**

3.3.3.1   *Support for (de-)provisioning and continuous update of user account information*

Many services require accounts to be provisioned before the users can access the service. Even for services, which can provision accounts at the time of the first user access, the account information needs to be kept up to date (e.g. VO/groups/roles) and the services needs to be notified to deprovision the accounts when they become inactive.

This activity will enable services that require it, to be notified for account provisioning, deprovisioning and updates using standards-based solutions:

- VOMS (de-)provisioning (for users without a personal certificate or users whose VO is not managed by VOMS): for those services that require VOMS proxy certificates, Check-in needs to be able to translate SAML assertions or OIDC claims to VOMS proxy extensions. Having this capability, users without a personal certificate or users whose VO is not managed by VOMS, will be able to use certificate-based services.
- Check-in needs to automatically (de-)provision users and groups to any application or identity store that is fronted by a web service with the interface defined in the System for Cross-Domain Identity Management (SCIM) 2.0 protocol specification.

3.3.3.2   *Uniform look and feel across all Check-in service component UIs*

All the user facing web interfaces of Check-in should be provided with a common, customisable look and feel. A new theme, based on Bootstrap, has already been created (see Section 3.3.2.2). The new theme will also be extended to the user-facing interfaces of the COmanage Registry, including the following:

- User enrolment
- User profile management
- VO/group management

3.3.3.3 *Proxy certificate retrieval through SSH key information managed in COmanage Registry*

Users can use SSH key authentication in order to retrieve proxy certificates from the MasterPortal (see Section 3.8). Currently, this requires users to upload the ssh public key via a dedicated self-service portal [R14]. This activity will enable proxy certificate retrieval from the MasterPortal using the SSH keys uploaded by users through the COmanage Registry user profile management page [R15]. It should be noted, that the MasterPortal will take into account the SSH keys from both the existing dedicated portal and the COmanage Registry.

3.3.3.4 *User-based active attribute value selection*

This enhancement will allow users to select which specific values of their attribute(s) will be released to a given service. For instance, users will be allowed to filter their affiliations or their entitlements expressing VO/group memberships and roles. Specifically, the following requirements have been identified:

- The attribute(s) whose active value(s) can be selected should be configured per relying party.
- For each attribute(s) whose values can be selected, it should be possible to specify a friendly description per relying party.
- The attribute value selection screen should support including an introductory text per relying party.
- In the case of single value selection, the first value should be selected by default.
- In the case of multi value selection, no value should be pre-selected.

3.3.3.5 *OAuth2 scope-based active attribute value selection*

This enhancement will allow selecting the values of specific claims returned in access tokens and results from requests to the *UserInfo* and the *Introspection* endpoint. Value selection will be triggered by requesting scopes in the following form:

<scope_name>[:<scope_value>]

For instance, when requesting "eduperson_entitlement:<value1> eduperson_entitlement:<value2>", the eduperson_entitlement claim will only include <value1> and <value2> (assuming these values are included in the user's entitlements).

3.4   eduTEAMS

The eduTEAMS service enables research and education communities to securely access and share common resources and services. Leveraging the ubiquitous presence of eduGAIN federated identities, eduTEAMS enables communities to securely authenticate and identify their users, organise them in groups, assign them roles and centrally manage access rights for use of community resources. GEANT operates and provides support for the following eduTEAMS offering: eduTEAMS service (delivery of the shared platform), eduTEAMS dedicated (setup and delivery of the single tenant platform for pilot) and eduTEAMS bespoke (setup and delivery of customly tailored single tenant platform for pilot).

The release notes for reporting period are provided in D5.2 [R4].

3.4.1 Maintenance activities

eduTEAMS [R16] is a complex service with a number of sub-components. In 2018, we have continuously upgraded individual sub-components to the latest stable versions, improving the performance and stability of the services. In parallel, the service was migrated to the Amazon AWS IaaS and it is now deployed in the EU-FRANKFURT and EU-PARIS regions. The Acceptable Usage Policy of eduTEAMS has been updated to follow the WISE AUP guidelines AARC-i044 [R17].

3.4.2 Summary of service enhancements

Proxy and Discovery Service

The following enhancements have been added:

- Improved support for entity categories in the SAML metadata, including support for Sirtfi [R18].
- Support for step-up authentication flows, where eduTEAMS can determine whether for a given service, a user has to use 2nd factor authentication and enforce it.
- Improved support for the AARC-G002 "Guidelines on expressing group membership and role information".
- Support for draft recommendations of the OIDCre-WP.
- The Discovery Service has been updated to match the latest recommendations from RA21.

Membership Management Service and other Backends

Users can link multiple account to a primary account. Account linking can be triggered by the user or can be triggered by the systems based on a number of heuristics. In all cases, the user has to successfully authenticate with both accounts before account linking is completed.

The backend integration with COmanage, HEXAA and Perun has been significantly improved. There has been a trend by many communities to use Keycloak in local deployments. eduTEAMS has been successfully integrated with IdP and SP instances of Keycloak .

Policies

The Acceptable Usage Policy was updated to the latest version of the WISE recommendations. eduTEAMS is GDPR compliant with support for the Research & Scholarship entity category and GEANT Data Protection Code of Conduction. In addition, now eduTEAMS supports also Sirtfi, both as an SP proxy and as an IdP proxy. The Privacy Policy was updated and a Data Processing Impact Assessment, was successfully completed as required by Article 35 of the GDPR.

Other

A number of UI and UX improvements have been implemented in COmanage, HEXAA and Perun.

3.4.3 Future plans

The following improvements are planned:

- Introduction of the step-up authentication service
- Improved registration flow for service providers
- Active role selection
- Support for AARC-G021 "Exchange of specific assurance information between Infrastructure"
- Support for AARC-G031 "Guidelines for evaluating the combined assurance of linked identities"
- Improved handling of user identifiers

3.5   INDIGO-IAM

The INDIGO IAM service [R19] provides an integrated solution for securing access to an organization's resources and services. It supports authentication via identity federations and social logins, a registration service providing moderated access to the organization, delegation and provisioning APIs and flexible account linking.

During 2018, the main focus of the work on IAM was to enhance its functionalities in order to fully support the requirements emerging from the WLCG Authorization Working Group, in support of the design of the future WLCG Authorization service.

The following sections summarize the main development and maintenance activities.

3.5.1   Maintenance activities

Significant work has been put in improving SAML support and integration with identity federations such as EduGAIN or SAML identity providers (e.g., the CERN Single Sign-On and the italian Sistema Pubblico per l'Identità digitale (SPID)). A bug that prevented the correct parsing of the *EduPersonTargetedID* attribute from SAML assertions has been fixed.

A blocking problem that prevented the deployment of IAM on MySQL 5.7 has also been fixed. The root cause was a backward incompatibility in MySQL 5.7.

3.5.2   Summary of service enhancements

3.5.2.1   Multiple OpenID Connect provider support

Up to IAM version 1.4.0, IAM supported a single OpenID Connect provider, Google. The support for authentication and account linking with an external OpenID Connect provider has been extended to allow multiple providers. Each provider can be listed in the IAM login page, and login buttons text and appearance can be customized with appropriate configuration. This enhancement has been showcased by enabling OpenID connect login from EGI CheckIn on the DODAS IAM and iam-test INDIGO IAM instances.

3.5.2.2   RCAuth.eu integration

RCAuth.eu [R20] has been integrated with INDIGO IAM in order to provide on-demand X.509 certificates to users without a certificate. The certificate is obtained using the OAuth4MyProxy OAuth-based protocol.

When the RCAuth.eu integration is enabled, IAM provides users with the ability to request a certificate on-demand from the IAM dashboard. What happens under the hood is that the user is redirected to the RCAuth.eu instance to be authenticated and give consent to the generation of an X.509 certificate and that such certificate is accessible by IAM. Once the user has given its consent, IAM fetches the generated certificate from RCAuth.eu and creates a proxy certificate out of it that is then stored in the IAM database and linked to the user membership.

3.5.2.3 Generic label and attribute APIs

A generic Labels API, inspired by the Kubernetes labels API, has been introduced in IAM that allows privileged users/agents to attach labels to groups and users.

These labels can be used internally by IAM (e.g., to provide additional metadata about users/groups status, to implement VOMS role semantics on top of IAM groups) or by external applications. A URI-based namespace mechanism is supported to avoid name clashes on attributes managed by different applications.

A generic Attribute API has been introduced to allow to link key-value pairs to users and groups. This information is meant to provide additional authentication/authorization information related to users and groups that can be included, if requested by the configuration, in tokens issued by IAM, providing a mechanism very similar to VOMS generic attributes.

3.5.2.4 VOMS provisioning

A VOMS Attribute Authority microservice has been developed to expose IAM VO membership attributes in the form of VOMS attribute certificates. The VOMS microservice talks to the IAM DB and leverages IAM support for x.509 authentication. The service is compatible with existing voms clients.

Since IAM does not provide a role abstraction, and that VOMS roles are equivalent to group membership asserted on request, a mechanism based on the labels API has been developed to flag some IAM groups as VOMS roles. These groups are not automatically included in generated VOMS Attribute Certificates (AC) but are instead returned using the VOMS role syntax only on explicit request from the client, preserving the original VOMS role semantics.

With this work, IAM can support a gradual and seamless migration from WLCG legacy AAI based on X.509 and VOMS to a token-based AAI.

3.5.2.5 Proxy certificate provisioning

A certificate provisioning API has also been implemented that allows users/agents with the appropriate privileges to obtain the proxy certificate stored in the IAM database. A simple bash client exploiting this interface has been developed to showcase token and certificate on-demand provisioning for command-line interfaces.

3.5.2.6 Group managers

IAM now supports group managers, which are privileged users that can approve group membership requests or add users to a managed group. Users can now request to join a group from the IAM dashboard home page.

3.5.3 Future plans

The main development planned for the future is to leverage Keycloak as the main IAM authentication engine. Keycloak is a flexible and popular open source solution by Redhat for centralized authentication and authorization. Integrating Keycloak in IAM will provide enhanced capabilities and improved sustainability.

We also plan to work on the technical alignment activities to achieve interoperability with other EOSC-Hub AAI solutions as described in Section 3.10 in this document.

3.6 Perun

A detailed description of the Perun service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

3.6.1 Maintenance activities

The service was operated without any substantial issues. During the last year there was one unscheduled outage caused by networking hardware failure, which was resolved within 3 hours. Planned maintenances were performed for deploying new version of the service and for updating underlying operating system.

Significant part of the operation routine is providing support to VO administrators and end users. VO managers often need help with configuring a new VO or with changes in registrations and user-flow in general. Most problematic part for end users is linking a new identity to an existing account.

3.6.2 Summary of service enhancements

Service is kept up to date with current release of Perun software. New versions are deployed within a few days from the official release for all major and minor releases.

One of the main improvements which was implemented was support for expiration of group membership. This functionality brings an improvement for manually managed groups, where administrators often forget to remove members who are no longer eligible to be in the group. The membership expiration together with notification system enables group managers to configure user-lifecycle consisting from enrolment, expiration and membership renewal. This can be further enhanced with existing Perun features like capturing user approval of AUP and access management for services. All these features combined together make powerful system for access management to the services, which can be fully or partially automated based on the requirements of the managed services.

Another significant change was in access management to resources available for VO. Initially the access could be managed only by the VO administrator. With implementation of new changes, the VO manager now can assign administrator for each resource, hereby delegating the responsibility of access management to the resource level. Alternatively, the resource can be marked as available for self-service for selected or all VO members, which enables them to self-register their groups for use of that resource. This change offers additional layer of delegation for responsibilities which is well aligned with other already available options.

There were a lot of other less significant changes, bug fixes and even performance upgrades. Full list is available on Perun GitHub pages in releases section [R21].

3.6.3   Future plans

The first item on the future plan is to redesign the process for account linking. The technical part works well, but the graphical interface seems to be complicated for some users as it is indicted by feedback obtained directly from users. Initial idea for this improvement is to clearly visualize all the steps in the process with indication what steps were already performed and what needs to be done to complete the linking of a new account. A prototype will be developed first and consulted with users to get the best user experience for the final version.

To improve the integration with other AAI components and possible some services, the new major version of Perun LDAP connector will be deployed. This connector exports data from Perun internal database to LDAP in nearly real-time speed. The main improvement of the new version of LDAP connector is an option to configure in detail which data will be exported to the LDAP server. LDAP server is primary used for integration with SP-IdP-proxies, therefore the ability to configure export of additional information to LDAP is crucial to support advanced features on SP-IdP-proxies.

3.7   WaTTS

A detailed description of the WaTTS service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

3.7.1   Maintenance activities

Software updates were done on a daily basis. No major downtimes apart from one, in which both key servers rebooted at the same time. Uptime was certainly above 99.9%

3.7.2   Summary of service enhancements

WaTTs continues to run in a demonstrator installation. This provides a detailed preview on the API and interfaces. The demonstrator installation of WaTTS has been integrated with a pilot instance of the RCauth - Online CA.

3.7.3 Future plans

In the short term we will install a dedicated WaTTS instance for EOSC-HUB. This instance will be connected to the RCAuth Online CA.

In the medium term we will develop that instance to be highly available.

3.8 MasterPortal

A detailed description of the MasterPortal service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

3.8.1 Maintenance activities

The code, which builds upon CIlogon code [R22], is in the process of being updated to the latest stable upstream code. At the same time the repositories are being reorganised to make maintenance easier. No known downtime.

3.8.2 Summary of service enhancements

A new client auto-registration endpoint (by default disabled) was added, which can be used for supporting the OpenID Connect federation specification.

3.8.3 Future plans

In the near future an update to the new code release is foreseen. Integration with the ssh-key upload capability of EGI's COmanage is planned. Investigation into the best scenarios for running in a HA setup. Service capability alignment with WaTTS is ongoing, and although at this point the two token translation services provide slightly different capabilities and a different focus, where the MasterPortal being primarily targeted at community proxy operators that will connect multiple science gateways to a single MasterPortal. It is feasible that both WaTTs and MasterPortal evolve in a way that allows merger of the services.

3.9 RCauth - Online CA

A detailed description of the RCauth - Online CA service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

3.9.1 Maintenance activities

Like the MasterPortal (see Section 3.8), the RCauth code, which also builds upon CIlogon code, is in the process of being updated to the latest stable upstream code. At the same time the code repositories are being reorganised to make maintenance easier. There was a short downtime in the attached 'WAYF'[2] service due to a full /var partition, so some additional production "hardening" may be necessary.

---

[2] WAYF=Where Are You From, a service for users to select a suitable identity provider.

3.9.2   Summary of service enhancements

At the beginning of the reporting period, the status was that only NIKHEF ran the RCauth service; the target is to have three instances running, the remaining two by STFC and GRNET. To this end, the focus was on preparation of plans for the development and deployment of the high availability service and on its operations.

3.9.2.1   Multi-site setup of RCauth

The goal of this task is to have three instances of the RCauth signing service, while not losing the IGTF accreditation. Thus, most of the effort in the task has gone into planning the key cloning and distribution process, as this needs to be done in a way that does not lose the IGTF accreditation of RCauth. Likewise, each hosting site has made plans for how to host the key securely, whether that requires using existing key management infrastructure (STFC) or procuring new hardware (GRNET) - the CA's private key must be hosted on Hardware Security Modules (HSMs) that have been validated according to FIPS140-2 [R23]. These plans - for cloning and hosting the RCauth private key - have been developed and agreed with the EUGridPMA, without whose approval RCauth would lose its IGTF accreditation. The outline plan was presented at the meeting in May 2018, and the detailed planning was then presented at the meeting in September 2018 and approved by the EUGridPMA.

At Nikhef a 'cold-standby' system was installed. The system software for the secure CA (i.e. the signing service) has been adapted to support independent distributed operation for up to 256 parallel issuances.

3.9.2.2   RCauth operations

Most of the efforts have gone into developing the governance framework for the future RCauth CA, specifically the policy (the Policy Management Authority, or PMA) and the governance board, with their initial memberships and terms of reference. Additional work has gone into identifying the main stakeholders for the service.

3.9.3   Future plans

The currently operational RCauth.eu instance is single-homed at Nikhef, where a local cold-standby system is available. To reach the desired service level, it is necessary to distribute the service geographically and move to a full active-active redundant set-up across the federated operators GRNET, STFC, and Nikhef. The software platform ("delegation service") will be re-engineered to allow for state consistency between a geographically distributed set of issance machines, assuming dedicated, secure, and low-latency virtual private circuits between the hosting sites. The plans for the development and deployment are quite extensive, particularly the plan for the cloning of the key, which now needs to be implemented. In particular, a minor software development activity is needed to support the execution of the plan.

As regards the production infrastructure, it may make sense to be able to containerise at least parts of it, in order to ease the deployment and improve monitoring, security and availability.

3.10 Integration activities

3.10.1 Summary of integration activities

During the reporting period, the main focus was put on the initial integration of AAI aimed at demonstrating technical ability for communities using either B2ACCESS or Check-in for their community AAI to use services behind the EGI and EUDAT e-Infrastructure SP Proxies. Specifically, a researcher whose community is managed by B2ACCESS should be able to access EGI services. Likewise, a researcher whose community is managed by Check-in should be able to access EUDAT services. The task's future plans include the improvement of this initial integration between Check-in and B2ACCESS as well as the interconnection of all EOSC-hub AAI services, i.e. GEANT eduTEAMS and INDIGO IAM, and the finalisation of the remaining harmonisation activities on the technical and policy level (see Section 3.10.3).

The remainder of this section elaborates on the activities required to achieve the initial integration between Check-in and B2ACCESS, including the adoption of guidelines for expressing user information.

3.10.1.1 Alignment of unique user identifiers

Both B2ACCESS and Check-in express user identifiers as *eduPersonUniqueId* values. The *eduPersonUniqueId* (ePUID) was introduced in version (201305) of the eduPerson Object Class Specification to meet the requirement for globally unique, non-targeted, persistent and non-reassignable user identifiers. This identifier is of the form uniqueID@scope. The "uniqueID" portion is unique within the context of the issuing identity system, while the "scope" is the administrative domain of the identity system where the identifier was created and assigned.

3.10.1.2 Alignment of group membership and role information

Both B2ACCESS and Check-in express group membership and role information as URN-formatted `eduPersontEntitlement` values with the following syntax (components enclosed in square brackets are optional): `<NAMESPACE>:group:<GROUP>[:<SUBGROUP>*][:role=<ROLE>]#<GROUP-AUTHORITY>`

The syntax above is based on AARC guidelines [R9]. To support the assignment of such URNs, B2ACCESS and Check-in have registered namespace `"urn:geant:eudat.eu"` and `"urn:mace:egi.eu"`, respectively. This change in the syntax of the group membership and role information required coordination with the relying parties that typically consume this information for authorisation purposes. It should be noted that Check-in is currently releasing group information in both the old and the new format to allow for an easy transition.

3.10.1.3 Alignment of name information

B2ACCESS switched the attributes containing the name from `cn` to `givenName` and `sn`. `givenName` and `sn` are part of the R&S attribute set [R24] which is released by a number of identity providers in eduGAIN. Within `cn`, the whole name was stored in one attribute, but with `givenName` and `sn`, the name is stored in two attributes, one for the given name and one for the surname. We

transferred the information for the majority of B2ACCESS users, the name information was transferred automatically; when this was not possible, the user had to supply the updated name attributes values. For connected service providers who still rely on the `cn` attribute, B2ACCESS translates this information during the user authentication.

3.10.1.4 Alignment of OpenID Connect/OAuth2 claims

Some additional user information like the affiliation or group membership are well known in the SAML context but not in the case of OpenID Connect/OAuth2. To provide a consistent naming of the claims, both B2ACCESS and Check-in implemented the mapping profile described in the REFEDS white paper of mappings between SAML and OIDC [R13]. After the implementation of the white paper all service provider can rely on a common vocabulary for information exchange.

3.10.1.5 Interconnection of EGI Check-in with EUDAT B2ACCESS

Check-in has been integrated as a SAML IdP (proxy) with the EUDAT SP proxy. Similarly, B2ACCESS has been integrated as a SAML IdP (proxy) with the EGI SP proxy.

3.10.2 Identified integration gaps

While the current integration between B2ACCESS and Check-in allows communities using either B2ACCESS or Check-in for their community AAI to use services behind the EGI and EUDAT e-Infrastructure SP Proxies, the following integration gaps have been identified:

- **Multiple user registrations:** Users still need to register in both B2ACCESS and Check-in due to differences in their Acceptable Use Policies (AUP).
- **Multiple IdP discovery steps:** As described in Section 3.1, the EOSC-hub AAI is based on the AARC BPA "community-first" approach, whereby users often need to go through multiple IdP discovery steps: (a) to select their Community AAI and (b) to select their Home Organisation. During this process, users don't need to re-enter their login credentials as long as their Single Sign-On session is active, however the IdP selection can be frustrating in some cases. The "IdP hinting" protocol proposed in AARC-G049 [R25] can greatly simplify the discovery process for the end-user, by either narrowing down the number of possible IdPs to choose from or by making the actual selection process fully transparent.
- **OAuth2 token validation:** Existing implementations of OAuth2 based Authorisation Servers do not support the validation of tokens issued by a different Authorisation server. As a workaround, services are required to connect to the Authorisation Server that issues these tokens, instead of relying on a single SP Proxy. This is illustrated in Figure 3-2, where *e-Infra B Service* has been connected to both *e-Infra B Proxy* (i.e. its infrastructure proxy) and *e-Infra A Proxy*. This additional connection allows *e-Infra B Service* to validate tokens issued by e-Infra A. However, this integration should only be considered a short-term solution, given that connecting all end-services with all e-Infra proxies cannot scale. As a long-term solution, we're investigating the extension of the OAuth2 Token Exchange flow for allowing OAuth2 Authorisation Servers to handle tokens issued by other trusted Authorisation Servers (see also Section 3.10.3.1.6).
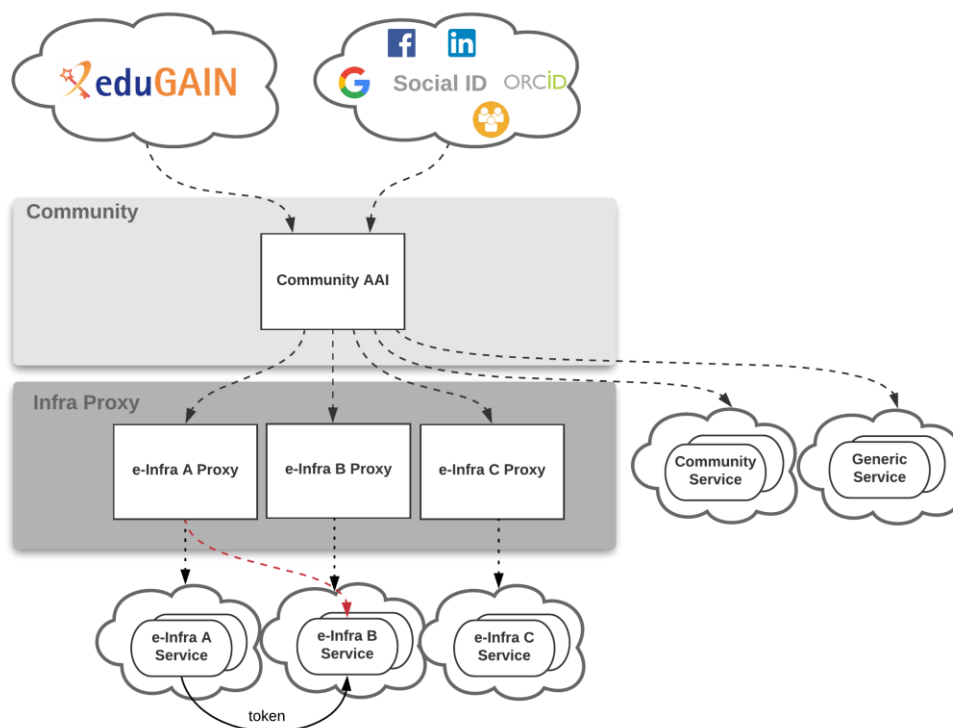
Figure 3-2 Current token validation scheme

3.10.3 Future plans

This section describes the future plans for the EOSC-hub AAI. These include alignment activities across the EOSC-hub AAI services which can be classified into technical and policy-related activities.

3.10.3.1 Technical alignment activities

Table 3-1 lists the identified technical alignment activities and their status. A green checkmark indicates a complete activity, otherwise the expected time of implementation is provided. The activities are detailed in the remainder of this section.

Table 3-1 Roadmap of EOSC-hub AAI technical alignment activities

Activity	B2ACCESS	Check-in	eduTEAMS	INDIGO-IAM
Alignment of user attributes	✓	✓	✓	M21
Alignment of VO/group membership and role information	✓	✓	✓	M21

Alignment of resource capabilities information	M18	✓	✓	M21
Alignment of affiliation information	M24	M24	M24	M24
Alignment of assurance information (including freshness of affiliation information)	PY3	PY3	PY3	PY3
Oauth2 token validation across multiple domains (initial implementation)	M24	M24	M24	M24
Oauth2 token validation across multiple domains	PY3	PY3	PY3	PY3

3.10.3.1.1 Alignment of user attributes

The attributes used to express user information should follow the REFEDS R&S attribute bundle, as defined in REFEDS-R&S [R24].

3.10.3.2 Alignment of VO/group membership and role information

VO/group membership and role information is typically used by relying parties for authorisation purposes. This information should be expressed according to the following syntax:

```
<NAMESPACE>:group:<GROUP>[:<SUBGROUP>]...[:role=<ROLE>]#<GROUP-AUTHORITY>
```

This syntax is detailed in the guidelines specified in AARC-G002 [R9].

3.10.3.3 Alignment of resource capabilities information

A capability defines the resource or child-resource a user is allowed to access, optionally specifying certain actions the user is entitled to perform. Capabilities can be used to convey - in a compact form - authorisation information. Capabilities should be expressed according to AARC-G027 [R27]:

```
<NAMESPACE>:res:<RESOURCE>[:<CHILD-
RESOURCE>]...[:act:<ACTION>[,<ACTION>]...]#<AUTHORITY>
```

3.10.3.4 Alignment of affiliation information

There are service providers that rely on affiliation information in order to control access to resources. Two different types of affiliation have been identified, namely *Affiliation within the Home Organisation*, such as a university, a research institution or private company; and *Affiliation within the Community*, such as cross-organisation collaborations. Affiliation information should be expressed according to AARC-G025 [R26] (still under review by AEGIS)] as depicted in Table 3-2:

Table 3-2 Alignment of affiliation information

Affiliation type	SAML attribute	OIDC claim
Affiliation within Community	`eduPersonScopedAffiliation`	`eduperson_scoped_affiliation`
Affiliation within Home Organisation	`voPersonExternalAffiliation`	`voperson_external_affiliation`

Note that AARC-G025 also provides guidelines for expressing the freshness of affiliation information. However, the expression of affiliation freshness will be covered as part of the alignment of assurance information activity (see Section 3.10.3.1.5).

3.10.3.5 Alignment of assurance information

Some service providers need to make authorisation decisions based on assurance information, which provides a means to express how much they can trust the attribute assertions about the authenticating user. The REFEDS Assurance framework (RAF) RAF-version-1.0 [R28] splits assurance into the following orthogonal components:

- the identifier uniqueness
- the identity assurance
- the attribute assurance

The components above are mounted into two profiles, namely cappuccino and espresso. Based on the use-cases, these two profiles can be used stand-alone or in combination with the REFEDS Authentication Profiles Single Factor Authentication (SFA) REFEDS-SFA [R29] and Multifactor Authentication (MFA) REFEDS-MFA [R30]. RAF also specifies how to represent the assurance component and profile values using existing federated identity protocols, currently SAML 2.0 and OpenID Connect.

AARC has developed additional guidance that extend RAF:

- Guideline on the exchange of specific assurance information AARC-G021 [R31], which defines RAF-based profiles equivalent to IGTF-BIRCH and IGTF-DOGWOOD, and introduces AARC-Assam, a new specific profile addressing assurance partially derived from social-identity sources.

- Guideline on stepping up the authentication component in AAIs implementing the AARC BPA AARC-G029 [R32], which supports use-cases that require a stronger authentication mechanism when accessing sensitive resources in BPA compliant environments.
- Guideline for evaluating the combined assurance of linked identities AARC-G031 [R33], which describes how to evaluate the combined assurance when linking different identities and defines compensatory controls that allow BPA Proxies to obtain assurance information even when the IdP of the Home Organisation is lacking support for RAF.
- Guideline Expression of REFEDS RAF assurance components for identities derived from social media accounts AARC-GO41 [R34], which describes how REFEDS RAF assurance components should be expressed by the BPA Proxies and how these may be combined on 'outbound' assertions, if the exchange involves authentications with credentials based on social media accounts (like Google, LinkedIn, Facebook, etc).
- Guidelines for expressing the freshness of affiliation information, as defined in AARC-G025 [R26] (see also Section 3.10.3.1.4).

3.10.3.5.1 Oauth2 token validation across multiple domains

OAuth2 Authorisation servers should be able to validate tokens issued by other trusted Authorisaton servers. Extending existing flows, such as the "OAuth2 Token Exchange flow" described in the document [R35], will be considered for enabling the validation of such externally issued tokens.

3.10.3.6 Policy-related integration activities

Table 3-3 lists the identified policy-related activities and their status. A green checkmark indicates a complete activity, otherwise the expected time of implementation is provided. The activities are detailed in the remainder of this section.

Table 3-3 Roadmap of EOSC-hub AAI policy-related alignment activities

Activity	B2ACCESS	Check-in	eduTEAMS	INDIGO-IAM
Alignment of privacy statements	✓	M21	✓	✓
Alignment of operational security and incident response policies	✓	✓	✓	✓
Alignment of Acceptable Use Policies (AUPs)	M21	M21	✓	M21

3.10.3.6.1 Alignment of privacy statements

For the EOSC-hub AAI and for virtually all of the SPs, compliance with the GÉANT Data Protection Code of Conduct version 1 (DPCoCo-v1) [R36] is implicit, since it reflects the Data Protection Directive and means compliance with applicable European rules (see AARC-G040 [R37]). To explicitly declare compliance with DPCoCo-v1, the privacy notice of each EOSC-hub AAI service should include a reference to DPCoCo-v1.

3.10.3.6.2 Alignment of operational security and incident response policies

The Security Incident Response Trust Framework for Federated Identity (Sirtfi) V.1.0 [R38] provides a mechanism to identify trusted, operationally secure eduGAIN participants and facilitate effective incident response collaboration. The entities of the EOSC-hub AAI registered with eduGAIN should meet the Sirtfi requirements and express Sirtfi compliance in their metadata in order to facilitate coordinated response to security incidents across organisational boundaries.

3.10.3.6.3 Alignment of Acceptable Use Policies (AUPs)

The acceptable use policy (AUP) and terms and conditions of an infrastructure bind the user to the purpose for which the services and resources are provided. The AUPs of different organisations, service providers, and infrastructures can vary significantly. To reduce the burden on the users and increase the likelihood that they will read the AUP as they access resources from multiple service and resource providers, the EOSC-hub AAI services should adopt the WISE Baseline AUP model WISE-AUP [R39]. This model includes a common set of criteria for acceptable use and terms and conditions which can be augmented with community- and infrastructure-specific terms and conditions.

3.10.3.7 Integration of EOSC-hub AAI services

This section presents the integration roadmap of the EOSC-hub AAI services. The status of each of the required integrations or the expected time of implementation is described in Table 3-4. Integrations which have already been established are marked with a check mark. Note that in the case of EUDAT B2ACCESS and EGI Check-in, integration is not considered complete (see identified gaps presented in Section 3.10.2), thus an amber checkmark has been used to indicate the status.

Table 3-4 Roadmap of EOSC-hub AAI integration

	EUDAT	EGI	GEANT	INDIGO-IAM
B2ACCESS		✓		
Check-in	✓			✓

eduTEAMS	M18	M18		PY3
INDIGO-IAM	PY3	PY3	PY3	

4 Marketplace and Order Management tools

4.1 Overview

The EOSC-hub Order Management System (OMS) as shown in Figure 4-1 comprises several central components including the Service Catalogue and Marketplace (MP), Service Portfolio Management Tool (SPMT), Service Order Management Back Office (SOMBO) and integrates many other systems to facilitate promotion, discovery, access and ordering of the productional EOSC-hub services.

The EOSC-hub OMS allow customers to access central EOSC-hub Marketplace, discover services and resources using the intelligent search and filtering mechanism, place an order and track it until its fulfillment and service or resource delivery. The EOSC-hub OMS integrate Order Management Systems, which are operated by other e-infrastructures.

Service Portfolio Management Tool and Marketplace are the elements of EOSC-hub environment with a direct relation to the EOSC services. SPMT being a platform where Service Organisations can manage service information relevant in the EOSC-hub environment, necessary to run operations around service delivery in the EOSC-hub scope. Marketplace is a dedicated platform there those services are presented to the users and made available to get access to. Is a place where the Service Organisations can define and present to the users dedicated service offers, users can issue an order for those offers and handle different phases of the ordering process. Those two mentioned tools are meant to support Service Order and Customer Relationship Management (SOCRM) process in EOSC-hub and in order to do so, development of features to implement established in SOCRM order management procedures is needed as well as integration with other tools involved in the activities.
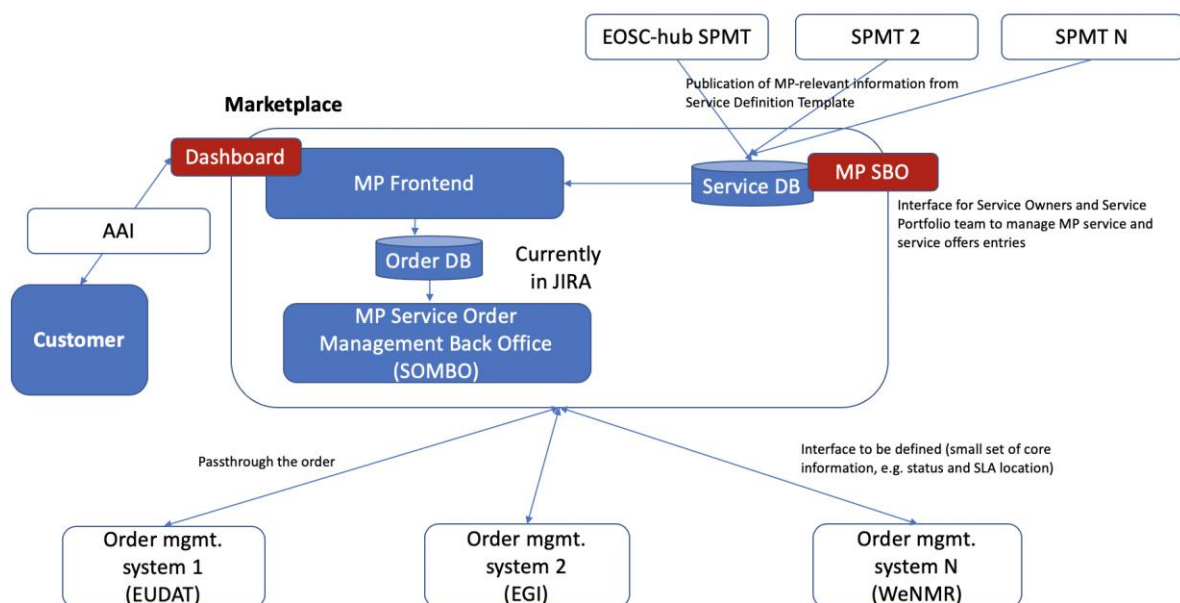
Figure 4-1.EOSC-hub Order Management System high-level architecture

4.2   Marketplace

A detailed description of the initial version of Marketplace service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

4.2.1   Maintenance activities

In order to ensure high availability and platform's smooth operation, regular software updates including gems' updates are being performed.

4.2.2   Summary of service enhancements

Marketplace during the last months has been evolving in various dimensions. As it is the central platform in user-facing service delivery chain it must provide:

- Features to ensure proper service presentation and creation of service offers for the Service Organisations.
- User friendly environment for the service discovery and service ordering for the EOSC users.
- Integration with other EOSC-hub tools involved in service delivery to support surrounding processes, such as Service Portfolio Management or Service Order and Customer Relationship Management processes.

At the very beginning of MP evolution the main emphasis was put on the EOSC user-facing functionalities, so the platform can facilitate basic service discovery and, most importantly, service ordering. User was able to crawl the database and filter the results to depict the services of interest and issue an order for them. Each of these features was rather in its early form and required deeper business and operational analysis. Being for several months in production, MP users started to provide requirements and insights based on their user experience in the MP. As a result an adequate development has been undertaken and completed. Main features delivered for the EOSC users are:

- Enhancements in user interface based on users' feedback.
- Evolved search engine and improved filtering mechanism.
- Enhancements in rating functionality.
- Implementation of features enabling voucher-based orders (Helix Nebula use case). This development is crucial as a potential direction to facilitate money-based procurement in the MP.
- Service tags functionality in user interface. Services are findable with their tags, and the tags are also presented in the GUI on a service entry.
- Adjustments in the Marketplace ordering procedure to align with new requirements from the Service Order Management Team and other stakeholders.
- Proof of Concept for implementation of a new service composition model - bundles. Concept aimed at clear guidance for the users to show which services are integrated on a level allowing their joint delivery.

Analysis on a shared vision for the different types of service access types. Together with other stakeholders it was established that services can be divided in terms of service access into 3 categories:

- Open access (services open to everyone) - no orders are placed, but access requests coming from EOSC-hub need to be tracked and stored.
- Orderable services with External ordering / catalogue type in the MP - orders are handled externally, but access requests coming from EOSC-hub need to be tracked and reflected in the Management Back Office.
- Orderable services with Internal MP ordering. Within this group we have a distinction of ordering models like:
  - Lightway integration (e.g. communication to support the orders via dedicated email registered with jira/Ops Portal)
  - Medium integration
  - Tight integration (as automated as possible access to the service after the 'order' button has been clicked)

After this finding has been agreed upon, appropriate development will take place in the next few months.

To support Service Organisation and allow proper service and service offer presentation the following enhancements were made:

- First version of the Marketplace Service Back Office - frontend for the Marketplace Service Database, to manage MP service and service offers entries has been released. It allows additional control over the service presentation in the MP and introduction of additional MP-relevant only fields to enhance service discoverability in the MP. At the moment the Service Back Office is open to the SPM team only. It will be given to the Service Organisations after a clear procedure to support service definition update control mechanism is defined.
- GUI to support management of the Marketplace service categorization has been made available. It allows maintenance of a standardised list of service organisations (internal to the MP only) and related infrastructures and platforms - elements to support better presentation of the services in the MP.
- Possibility to choose the order management integration method, both for services and service offers.

To support many requirements coming from WP4 and better support Service Order Management (SOM) procedures many improvements and extensions in the tool chain MP - JIRA - SOMBO has been introduced, such as:

- Inclusion of a new fields relevant for the procedure. For some of them the source of information is user input (those needed to be included in the GUI), for others, service entry (information at the moment) filled in the MP Back Office by the SPM Team.

- Notification mechanisms about relevant events (ticket creation) for the roles involved in the SOM Procedure such as JIRA shifters.

- Facilitation of communication between user and shifter, who manages the order after its submission in MP: use of JIRA ticket comment field to propagate the communication to the MP.

- Preparations to introduce Project-Orders hierarchy in JIRA which will be reflected with Epic-Task hierarchy model.

It is a continuously ongoing process as the procedure is still under development and it influences all of the tools like MP, JIRA, Ops Portal involved in the service order management.

4.2.3 Future plans

After achieving the goals connected with features aimed at Service Organisations and service discovery, next months will be focused on improvements around overall user experience, further development supporting service ordering procedure on both user-facing and integrational side and introduction of a new user-facing concept: Marketplace Project.

Marketplace Projects is a light way approach to allow the user to organise his services and service orders into a logical block to reflect a common purpose and gain support in the scope of the created project. It's ultimate goal is to provide a user-friendly and helpful UI where EOSC services of a user interest can be gathered and managed and the user is carried out step by step through the service order management process at the same time being separated from the complex operational side of this process. A prototype of this concept has been already created, discussed and agreed on and can be found under: [R40] (pass: eosc123).
Rest of the foreseen development in the scope of other user-facing features and integrational aspects can be found beneath:
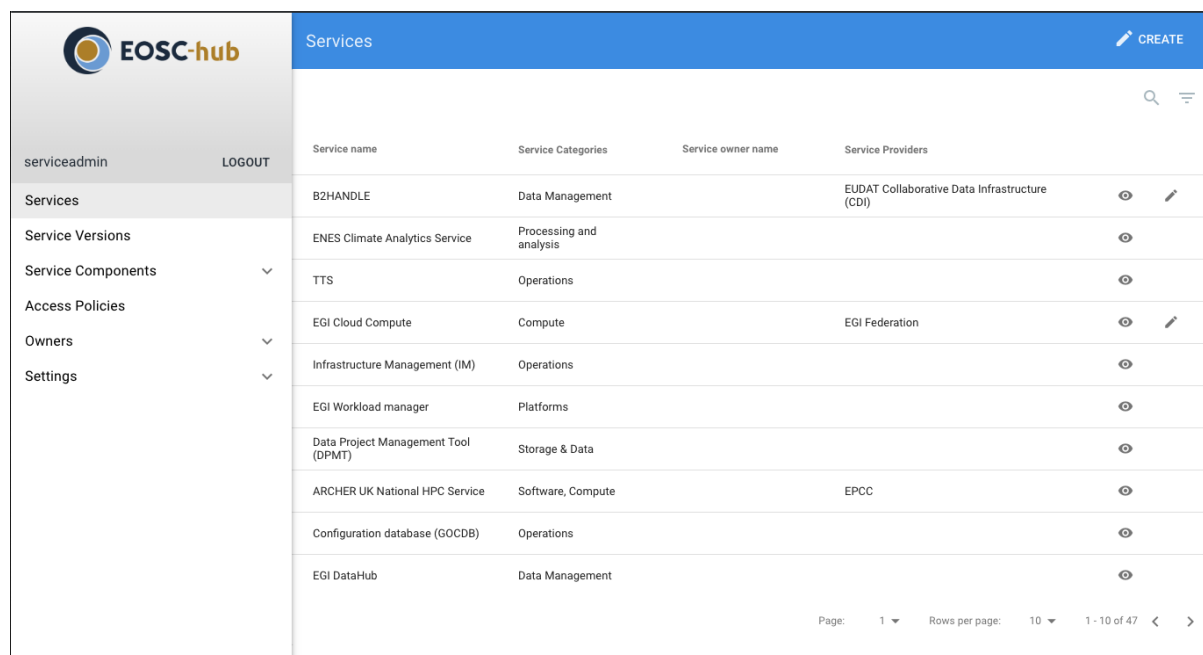
- Introduction of the Project - Order hierarchy in the MP - JIRA integration. Support for new user use cases based on this feature. Better support for the operations behind order management.

- Implementation of an early user support in the MP Project - after a project is created, the user can start a chat with an EOSC-hub expert to ask for guidance.

- Evolution in other features supporting the Customer Relationship Management

- Enhancements in the GUI.

- Integration with SPMT, to be able to retrieve services (marked as ready to be released in the Marketplace) and their basic information.

4.3 Service Portfolio Management Tool (AGORA)

The Service Portfolio Management Tool is a tool aimed at facilitating service management in IT service provision, including federated scenarios. SPMT represents a complete list of the services managed by a service provider; some of these services are visible to the customers, while others are internal. The service management system has been designed to be compatible with the FitSM service portfolio management [R41]. Figure 4-2 shows the top Web UI of the SPMT.

A detailed description of the SPMT service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

Figure 4-2 EOSC-hub SPMT Web UI

4.3.1 Maintenance activities

The AGORA team performs regular maintenance and security updates in the 1st week of each month on all SPMT/AGORA.

4.3.2 Summary of service enhancements

In the last period we added support for the following functionality in SPMT/AGORA:

- Added Support for multiple and customisable feeds that allows user to create multiple service catalogues (internal / external) and to choose which fields to show in the catalogue.
- Added support to publish Service info updates to Argo Messaging Service as a 1st step to integrate with the Marketplace.
- Added new API call to get the list of Service_Types listed in catalogue that will be used to integrate SPMT with the CMDBs.

4.3.3 Future plans

Our plan for the next period is to Implement the new Service Definition Template in SPMT/AGORA and to proceed with the integration of SPMT with the following tools

- **Marketplace**: In order to integrate with the Marketplace Front end, AGORA will be sending service update notifications to marketplace including a timestamp of the last update via the Argo Messaging Service to upload the service info.
- **Service Order Management Back Office** (**SOMDO**)**:** During the management workshop that was held in Munich in March 2019 it has become clear that there is a need for Service Order Management Back Office to be populated with contact info for each service provider and / or federation member, the Agora team plans to investigate the best solution to provide this either directly to SOMDO and/or by becoming an attribute authority in Check-in.
- **CMDBs:** The plan is SPMT/Agora to become the authority for the service types that are used by the CMDBs GOCDB and DPMT to classify the services, this will also help the marketplace to correlate the feeds from CMDBs with the Services listed by SPMT by using the Service_Type as key.

4.4 Integration activities

4.4.1 Integration of Marketplace with SPMT

4.4.1.1 Summary of integration activities

At the moment there is no productional integration between Marketplace (MP) and SPMT. After a series of meetings, it has been decided to divide this activity into 2 phases:

1) First integration that focuses on the integration protocol - developing a connection between the MP and SPMT that allows the MP to register changes newly introduced in SPMT, relevant from the MP's point of view. At this stage MP will retrieve only the most important fields of Service Description Template, such as: Service Unique ID, Service Name, Service Description etc.

2) Extensions in MP - SPMT integration: after the completion of tasks included in the first phase, the integration will be extended with the rest of MP-relevant SDT fields. It is possible that some of them, due to their nature (not being a simple text field), will conclude in new requirements for the SPMT. However, this analysis will take place after the first phase is completed and the first productional version of MP-SPMT integration is in place.

4.4.1.2 Identified integration gaps

To achieve a synchronisation between MP and SPMT (synchronisation that is sustainable and replicable) SPMT and MP teams came up with a workflow, which comprises a list of tasks that need to be completed to support it.

The workflow with tasks needed to achieve an intended integration:

1. Start of an instance not populated with services and that has not been synchronized with SPMT yet.

2. First call of the SPMT endpoint [R42] in order to get an initial state of services registered in SPMT.
   - To sustain workflow's continuity a timestamp for the taken snapshot is needed.
3. Requests for updates from SPMT via ARGO messaging system [R43]. Updates should start at the same timestamp that was received at #2.
4. Pull updates.
   - MP consumes messages and acknowledge successful delivery
   - when the MP goes down unacknowledged messages reside in the topic queue
   - once the MP is up again it starts to normally consume messages
5. Receive notifications about updates.
   - for this push notifications on the SPMT side are requested

To properly distinguish which services are willing and ready to appear in the MP a dedicated flag manageable by the service owner has to be present.

By implementing this workflow there is an automatic way to create a new instance of the MP, pull service information from SPMT and process SPMT updates. It supports several use cases:

- To setup MP - SPMT integration for the first time in production, the whole workflow, starting at #1, should be performed.
- When a new version of the Marketplace service is deployed or marketplace goes down for any other reason, we start at #4.
- When in need to setup a test/demo instance workflow should start at #1.

4.4.1.3 Future plans

As mentioned before, after completing the first phase and achieving the first productional SPMT - MP integration, SPMT and MP teams will work on passing the information about the rest of MP-relevant SDT fields.

4.4.2 Integration of Marketplace with Service Order Management Back Office

4.4.2.1 Summary of integration activities

Service Order Management Procedure is hosted in 3 tools: Marketplace, EOSC-hub JIRA instance and Operation Portal which implements COMBO, whereas the latter two are considered as a Marketplace Order Backoffice. Marketplace as a tool where the user interaction takes place is a tool that triggers the whole procedure and interacts every time when user input or notification is needed.

MP is responsible for the communication of all relevant data and information in the SOM Procedure on the User - Operations Team line. The major activities where the MP is involved in the SOM Procedure scope are listed below:

- Ticket creation: passing information about the order relevant in SOM Procedure to the the tool supporting order handling. Information should be organised in a manner suitable for order handling.

- Fostering communication: allowing to exchange information between the Customer and Operational Teams (First Line, Providers etc) relevant in the scope of the order.
- Passing information about the order status.
- SLA presentation to the user after it is in place

To facilitate these activities, integration between the MP-JIRA is in place, no direct MP-Ops Portal integration is existant. If an exchange of information between the MP and Ops Portal is needed, JIRA is the middleman.

JIRA integration is based on two side communication - from marketplace to JIRA via JIRA's API, and from JIRA to the Marketplace via JIRA's webhooks. JIRA integration requires configuring mapping of several key properties, such as:

- MP_JIRA_USERNAME - username of the service user as which application is accessing JIRA
- MP_JIRA_PASSWORD - password of the service user as which application is accessing JIRA
- MP_JIRA_PROJECT - project key in which marketplace will create / update issues
- MP_JIRA_URL - url to jira instance

Which allows integration with various JIRA instances/JIRA projects, manageable on ENV variables level. JIRA instance itself has a dedicated workflow and custom fields which correspond to what MP itself expects and to reflect the Service Order Management Procedure. A dedicated workflow is used in status synchronisation, custom fields are added to a JIRA ticket template to forward order information relevant in the SOM Procedure. Some of the custom fields:

  - CI-Name

  - CI-Surname

  - CI-Email

  - CI-DisplayName

  - CI-EOSC-UniqueID

  - CI-Institution

To foster the communication between the user and operational teams a privilege to read and add comments for the MP user (integrated with JIRA) has to be added and the communication is enabled.

SLA presentation to the user is not yet in place and will be a subject to next discussions in the MP-Backoffice subject.

4.4.2.2   Identified integration gaps

In order to more efficiently manage and sustain MP-JIRA integration it is crucial to introduce versioning to JIRA ticket workflows and templates. In this way it will be possible to smoothly move between the versions without any disruption to the MP users. To orderly manage this task, a JIRA schema versioning procedure has been proposed:

In case of introducing changes to any project related schema (workflow/ticket etc):

1. A dedicated JIRA project should be in place where new schemas are created and VERSIONED (via agreed naming pattern) - e.g. EOSCSOWORKFLOWS.
2. In case of a new schema, the info about the new schema should be propagated via CHANGELOG (for the beginning a confluence page would be sufficient), where all schema versions are listed with names, basic information about the changes and any other relevant comments).
3. A new version is connected to EOSCSODEV and EOSCSOMASTER projects (JIRA projects connected with MP and Ops Portal dev instances), where appropriate development takes place.
4. After integration with a new schema, moving changes up to EOSCSOSTAGING (common JIRA project for MP and COMBO integration).
5. Testing phase.
6. After positive testing results, moving to production to EOSCSO, which is production project in JIRA.

Additional gap, for the sake of introduction of Project-Order hierarchy in JIRA, is the creation of a new ticket type in the JIRA ticket schema (currently it includes only Task type): Epic. The hierarchy model could not be reflected in JIRA without it.

4.4.2.3   Future plans

Next steps will focus on introducing the ticket hierarchy in JIRA and retrieving info about the SLA from the Ops Portal COMBO through JIRA. This functionality has to be yet discussed though.

5 Integrated Business and Operations Support Systems

5.1 Overview

This chapter reports on the maintenance and integration activities performed during the first year, for the Operations Portal and its component SOMBO, DMP, GOCDB, DPMT, and SVMON. A summary of enhancements these services received is also provided, along with action plans for the future.

It should be stressed that although the tools and services managed in this task undergone lots of changes and were partially integrated, following the operational requirements, still the concept of joint distributed configuration management and business layer was not agreed. The development and agreement on joint Data Model, which should provide a baseline for further integration of the support systems mentioned above, is an essential task for the next project year.

5.2 Operations Portal

A detailed description of the Operations Portal service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

5.2.1 Maintenance activities

We perform regular upgrades on the software part to improve the performance of the applications, including third parties libraries (javascript libraries, css framework or php framework) . We work actively on regular improvements on the data aggregation framework.

And we also put in place code reviews with the help of dedicated tools (SonarQube):

- to ensure the code quality
- to improve the efficiency of the code
- to ensure a good maintainability of the application

This last year we have migrated from bootstrap v3 (css framework) to v4 which brings lot of flexibilities and new components for the frontend part. We have made also an important upgrade of our data aggregation framework.

5.2.2 Summary of service enhancements

All the enhancements described below have been developed in parallel during the first year of the project. All the features are available currently on the test instance and will be delivered during Q2 within an important release.

Operations Dashboards

One historical module of the Operations Portal is the dashboard module in which a security dashboard and different operations dashboards provide information about failing monitoring probes and allow opening of tickets to the affected resource centres. The dashboard module also supports the central grid oversight activities. It is fully interfaced with the EGI Helpdesk, RTIR

(helpdesk used by security team) and the monitoring system through messaging service. This module is based on the old version of the Symfony framework and has been developed following the procedures and workflows set up during the EGEE project. The procedures have evolved, and it should be reflected on dashboards. We have worked on a new version of the module, which is more flexible with the use of a recent version of Symfony. This new dashboard is also completely customisable by users through the use of settings.

This dashboard module is integrated with EGI Checkin and allows a central authentication .

Service Order Management Back Office

The Service Order Management Back Office (SOMBO) is a new tool which has been designed to complete Marketplace features specially to facilitate the service order management.

First of all, the tool will provide facilities for shifters, so they can easily browse and modify the service orders registered into Marketplace. They can also communicate with resource providers or customers. Then they can interact with the infrastructure to allow customers to access/ configure/order the requested service or resources. This phase to be organized through a negotiation process between shifters and resource providers.

Then after the use of the service the tool should facilitate the collection of statistics for reporting. More globally SOMBO should provide resource usage consumption and reports about the quality of service and eventually survey about user satisfaction.

Currently the interface implemented in the Operations Portal is split into 4 parts:

- Service order list
- Authorization page
- JIRA details page
- SLA page

Figure 5-1 Service Order List page of COMBO

This first page, as shown in Figure 5-1, provides a list of the service orders sorted in 3 sections: on-going, accepted and rejected service orders with a brief description. Depending on your access rights you can act on this service order: display details registered in Jira, go directly to the issue in JIRA , go to the service level agreement page

As shown in Figure 5-2 authorization page allows the administrators of the tool to add/remove users to a group of services. The service order is associated with a service and only user authorized within this service will be able to treat the request.

Figure 5-2 Authorisation page of COMBO

The JIRA details page as shown in Figure 5-3 summarises all information registered in JIRA and allows authorized users to modify it, they can also change the status of the request in agreement with the workflow. The authorization is based on the declarations done in the authorization page.

New JIRA Issue #EOSCSO-329

information about user

Name *

Frank

Surname *

Michaelis

Email *

frank.michaelis@uni-bielef

Display name

Frank Michaelis

information about user affilations

Institution

Universität Bielefeld

Department

BITS

Departmental web page

http://www.uni-bielefeld.de

Supervisor name

information about the project

Customer typology

single_user ▾

Reason for access

We, the datacenter of the U

User group name

Project information

information about service

Category *

Compute

Service *

EGI Cloud compute

Offer *

General purpose

Service order target *

support@egi.eu

Creation Date

13/06/2019   13:57

Last Update

13/06/2019   15:53

Save   🔍 Manage SO

Change Status   In progress ⟳   Approved ✓

Comments

Add Comment

Add internal comment

Figure 5-3 JIRA details page of SOMBO

Start date : 02/04/2019		End date : 02/04/2019

Service Area : Sharing & Discovery

Service : B2HANDLE

ServiceOption : B2HANDLE For Researchers

Request	Value	Total
Access type	reserved	
Start of service	01/16/2019	
Number of days	365	

➕ Add provider   ✉ Contact provider

Figure 5-4 Service Level Agreement page of COMBO

The Service Level Agreement (SLA) page as shown in Figure 5-4 allows the users to contact different service providers . Users can also generate a pdf document which could be used as a template for SLA.

AAI integration

The AAI integration is now complete. Users need to use EGI Checkin to be authenticated into the Operations Portal. We manage EGI roles and EUDAT roles within the attributes forwarded by AAI system.

5.2.3   Future plans

The following plans are agreed and added to the service roadmap:

- Provide all these features in production and thereby implement feedback from users
- Manage SLA into the Service Order Management Back Office
  - For a given service order generate a document (or several documents) which will correspond to an agreement between the resource provider(s) and the customer.
  - This document will describe the level of service (availability, reliability , quality of support, resource usage) offered by service provider(s).
  - The SLA could be global to an offer (group of services) or manage per service.
  - The interface will provide different templates of documents depending on the type of resources.
- Add usage reports into the Service Order Management Back Office.

5.3   GOCDB

A detailed description of the GOCDB service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

5.3.1   Maintenance activities

We fixed various bugs including ensuring all the tables on Site and Service pages were sortable and accepting parenthesis in the HostDN field as per the OGF standards. We also improved our documentation and the depth at which knowledge is shared among the GOCDB team.

5.3.2   Summary of service enhancements

During the first 12 months of the project, we have introduced new features and fixed issues. First, we have made the production instance of GOCDB accessible over IPv6. Second, we added a new 'notify' feature, allowing users to receive notifications about Sites and Services. We also developed a privacy and acceptable use policy.

We have also been working with DPMT to work out how GOCDB and DPMT can best be developed to meet the needs of the EOSC community, reviewing the architecture of the GOCDB service (including failover) and planning changes to underlying GOCDB systems to upgrade the OS and improve configuration management to ensure the long term stability of the service.

5.3.3   Future plans

- Update the version of the GOCDB software in production.
- Improvements will be made to how "Reserved Scopes" are handled by the GOCDB software.
- Creating new EOSC-Hub specific ServiceTypes automatically when they are added to the EOSC-Hub SPMT API.
- A second, configuration managed, production instance of GOCDB will be deployed behind our load balancer. This will not only increase the resilience and reliability of the GOCDB service, but also the long-term stability of the service, by allowing production instances of GOCDB to be brought online faster and with more guarantee that the configuration is correct.
- Over the course of the project, the failover instance will need to be moved to new infrastructure and place under configuration management.
- The functionality of the Write API will be expanded to meet evolving use cases.
- A EOSC-Hub view will be developed and deployed on under its own URL. The EOSC-hub view will provide an EOSC-Hub centric look at the data already within GOCDB. Most views will only show resources with the 'EOSC-Hub' scope tag applied, and we will use the current ServiceGroup functionality to represent EOSC-Hub's federated services, which will be in the same views as non-federated services. Accessing the GOCDB PI via this URL will also, by default, only return entities with the 'EOSC-Hub' scope tag.

5.4 Data Project Management Tool

A detailed description of the DPMT service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

5.4.1 Maintenance activities

The DPMT is hosted by MPCDF and accessible at [R44]. It has been running smoothly over the reporting period with only one brief downtime due to maintenance of MPCDF's virtual hosting environment. Minor bug fixes and visual enhancements have been realized whenever necessary or appropriate.

5.4.2 Summary of service enhancements

Specific enhancements worth mentioning are:

- Introduction of the Plone add-on `plone.restapi` [R45]. Now it is possible for machine agents to invoke any action that a human can perform through a browser via REST calls (provided the agent has the appropriate privileges).
- Introduction of a scope attribute for projects. This makes it possible to explicitly define the context in which a certain project is performed (EUDAT CDI, EOSC-hub, SeaDataCloud, …). It is possible to assign multiple scopes to one project. Other information related to the project such as registered services or service components "inherit" the project scope.
- The accounting records that DPMT provides in StAR format [R46] have been enhanced by the addition of context information. More specifically, references to the related project and customer are now given. This will make it possible for clients such as the accounting portal to provide more detailed and informative views on ESOC service usage.

A changelog for DPMT's most important component is given at [R47].

5.4.3 Future plans

The most significant change planned for the DPMT at the moment is the introduction of a more powerful, detailed and flexible way to describe service endpoints of registered service components. This becomes necessary as practice has shown that some providers use the same component instances for multiple services and projects. In those cases, these instances are configured such that they offer multiple service endpoints. DPMT's data model so far does not allow for a proper description of such complex deployments. This will become possible with the planned improvements.

To further enhance interoperability with other operational and user facing tools of the European Open Science Cloud the service types used by DPMT to categorize individual service deployments will be brought in line with the service type descriptions elsewhere (Marketplace, Operations Portal, SPMT, GOCDB, ...).

DPMT's underlying web framework - the Plone content management system [R48] - has finally been ported to Python 3, but at the price of some significant changes. For DPMT to become able to benefit

from this development major adjustments might become necessary. As a first step we will explore where and to what extent changes and adjustments will be required.

5.5 Data Management Planning Tool

A detailed description of the DMP service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

5.5.1 Maintenance activities

The Data Management Planning Tool consists of a front-end component with a Web UI (OpenDMP [R49]) through which data management plans are managed and monitored and a back-end service (eestore) that collects and provides information from a variety of data service registries such as re3data.org [R50], through a uniform interface to OpenDMP shown in Figure 5-5. OpenDMP is developed and maintained by OpenAIRE and the eestore is developed and maintained by EOSC-hub.
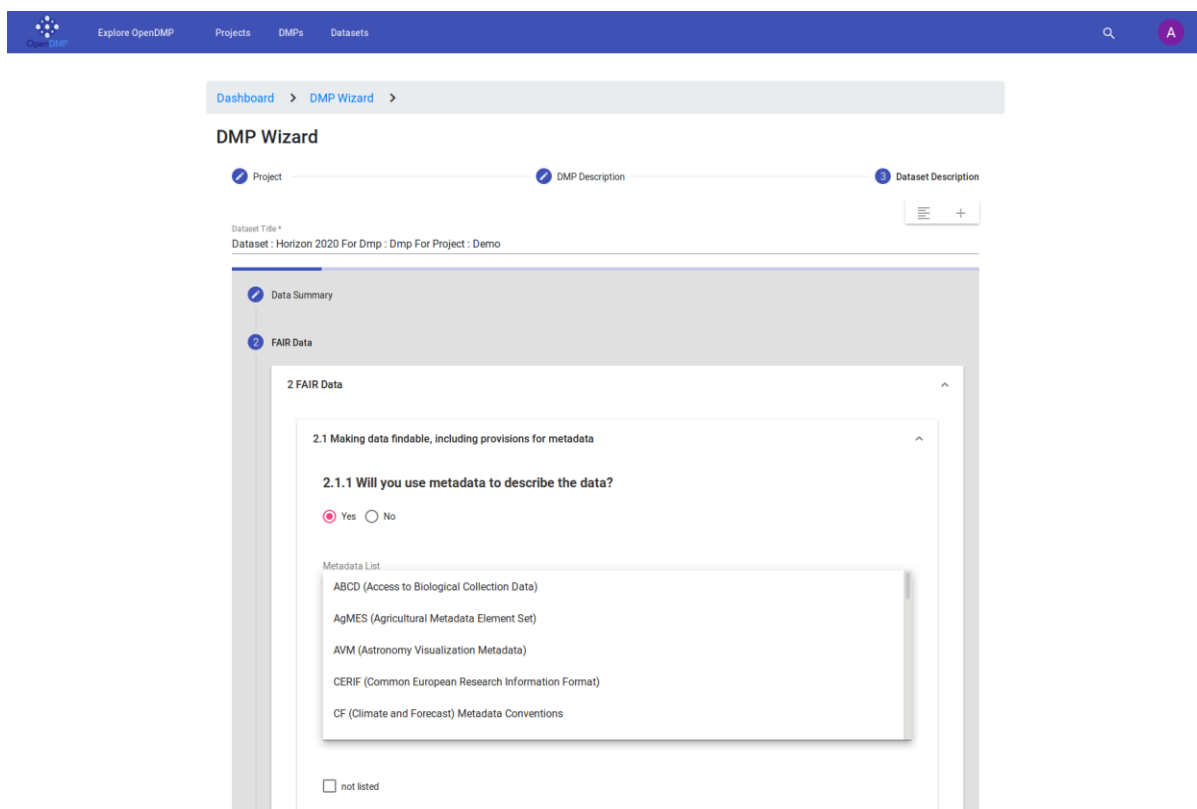

Figure 5-5 OpenDMP Web UI with a drop-down menu of metadata schemas supplied by the eestore

The eestore component of the Data Management Planning Tool is running on the Norwegian National e-Infrastructure for Research Data (NIRD) platform [R51]. The service runs in a Docker container on the Kubernetes service platform that is part of NIRD. The service has been running reliably although the platform has some significant downtime due to problems with the storage and

upgrades of Kubernetes. At the moment the information eestore presents is public and there is currently no need to provide access control to the information.

5.5.2   Summary of service enhancements

The API of the eestore component was rearranged to make it easier to include new information provider. Throughout the year new sources have been added to the service enabling the eestore to provide a more extensive set of information to openDMP. This work continues and is driven by the needs of openDMP. Modifications have been made to allow the eestore to be deployed on any platform. However, this needs further testing. OpenDMP has undergone and will continue to undergo updates following feedback from users as well as support for more templates.

5.5.3   Future plans

The eestore currently provides information from public registries. However, we anticipate the need for private information that will require integration with B2Access to provide access control to the information. Integration of the eestore with the DPMT and MarketPlace is in the early stages of investigation. A first version of integration with the DPMT that enables the eestore to fetch information on the types of services offered by EUDAT has been implemented. We have outlined an approach to automatically update a data management plan with information on services requested through the MarketPlace by accessing information available in the DPMT. We will continue to work on refining the architecture with stakeholders to ensure the integration is achieved. This work is targeted for the JAM1.4 milestone. We also expect to continue to extend the number of registries consumed by the eestore and expect to replace some of the existing registries that require manual processing of the data before ingestion into the eestore.

5.6   Service Versions Monitoring Tool

A detailed description of the SVMON service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4]. As SVMON provides harmonic views on the configuration information obtained from GOCDB and DPMT in can be seen as a part of distributed Configuration Management Database in addition to the role of version monitoring tool.
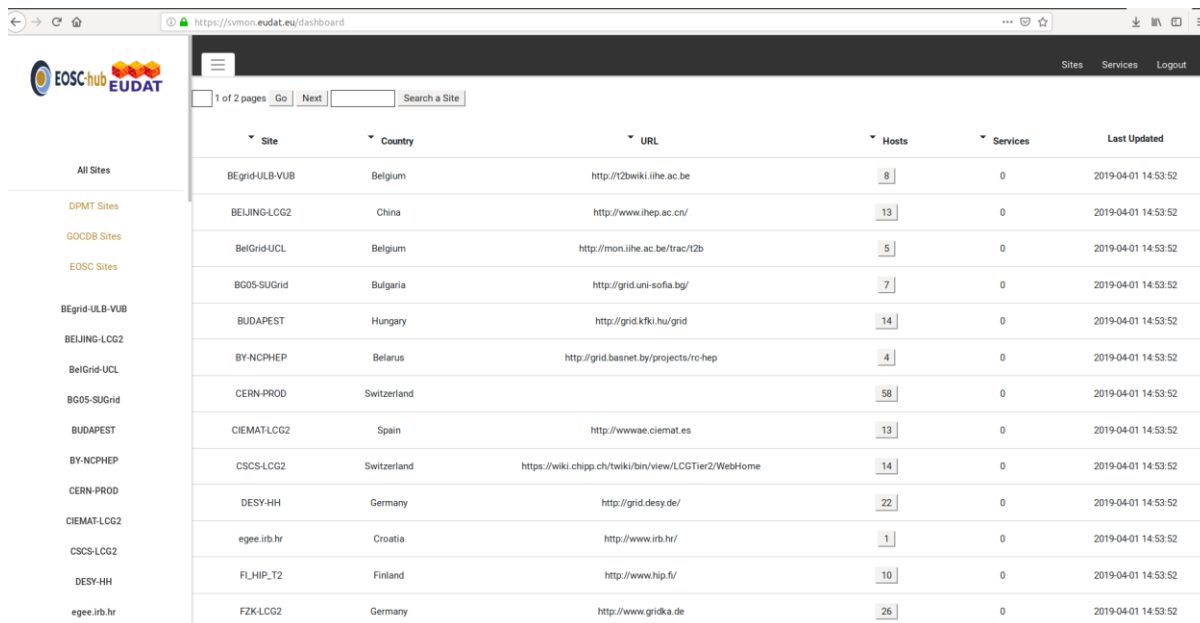
5.6.1   Maintenance activities

SVMON service is hosted at KIT. The production instance of the SVMON is running in highly available virtual cluster, which provides full backup and efficient recovery procedure for the service. The core backend based on Java Spring Framework provides REST APIs to execute HTTP requests. The underlying database is integrated with KIT MySQL server [R52].   The service is backed up using IBM TSM system.

5.6.2   Summary of service enhancements

During the first year, the service has undergone significant enhancements. Several views in Web UI have been implemented to distinguish site information from multiple resources, DPMT and GOCDB. We have rewritten the SVMON client [R53], to enable sites its easy installation for service providers.

We have implemented a third view for EOSC-hub specific sites based on the scoping information from GOCDB or DPMT. Figure 5-6 shows the current SVMON Web UI. The paging functionality, search functionality have been added to the Web UI. We have also provided alternatives to post data via SVMON client, and published it in PyPI [R54].

We have implemented a general OAuth2.0 interface and connected it with EUDAT B2ACCESS.

Figure 5-6 **SVMON Web UI**

5.6.3 Future plans

The distribution of SVMON client will be extended to cover more sites contributing with their services to EOSC-hub.

As SVMON is connected to GOCDB and DPMT it provides a combined view on EOSC-hub configuration information stored in EGI and EUDAT configuration databases. Further web views and filtering shall be implemented to facilitate access and lookup of the configuration information. Further harmonization of heterogeneous data sources and modification of the SVMON database is planned.

5.7   Integration activities

5.7.1   Integration of Operations Portal with Marketplace

5.7.1.1   Summary of integration activities

Currently there is no direct integration between Operations Portal and Marketplace.

All information are exchanged through JIRA issues. The Operations Portal is highly integrated with JIRA API and is able to browse and modify JIRA issues.

The back office SOMBO is connected with JIRA and allows users to modify issues without declaring any authorization into JIRA. The authentication and authorization part is ensured by the Operations Portal - via the AAI integration and the EGI Checkin attributes.

5.7.1.2   Identified integration gaps

The main difficulties until now are related to the fact that the Marketplace is still under development.

Under development means that the structure of the JIRA issues and the content of these issues are changing a lot. It means that we have to adapt the code on our side very frequently.   Another integration issue is the localisation of the information - this is a global issue not necessarily related to the Marketplace itself.

Some information needs to be centralized: list of shifters, list of resource provides, list of contacts. It's not clear currently where this information needs to be stored and in the meantime this information is stored in configuration files in different places. This is not easy to provide stable interfaces and coherent information in such conditions. A proper integration with SPMT and EOSC-hub CMBD together with further development of these tools with the goal to include missing information should resolve this issue. The plans for this integration are still in the initial stage and should be further evolved.

5.7.1.3   Future plans

The immediate plan is to provide a prototype of the back office and makes it available for shifters.

This step requires to have a stable version of the JIRA issue structure.

Then we will work step by step on missing features: a complete SLA management system with the integration of the resource usage report and the related monitoring information .

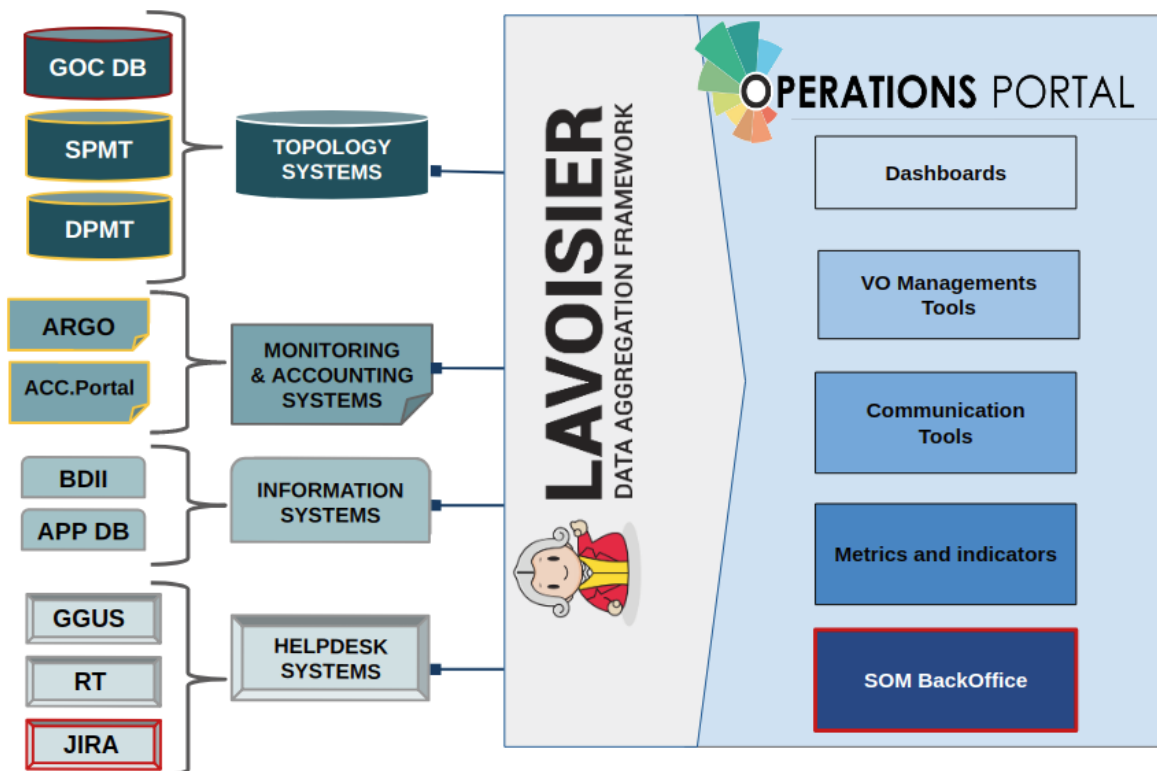### 5.7.2   Integration of Operations Portal with other systems

Figure 5-7 Operations Portal integration with other systems

5.7.2.1 Summary of integration activities

Figure 5-7 shows the integration which have been made for the Operations Portal through the use of our data aggregation framework "Lavoisier". You can see in red and yellow all the integrations needed for the final version of the SOMBO.

The completed integrations are shown in red (GOCDB, JIRA). The ongoing integrations are depicted in yellow: ARGO and Accounting Portal are integrated in Lavoisier framework, but the information is currently not used in the SOMBO, DPMT is also integrated in Lavoisier framework, but in read only mode (we don't use currently the write API of DPMT). The SPMT test instances has also been integrated, but currently the SPMT information cannot be used in Operations Portal.

5.7.2.1   Identified integration gaps

- DPMT: We are currently retrieving information from DPMT, but we are not fully using the DPMT API (read mode only). We need to understand now how to use this tool efficiently with SOMBO to directly query the resources or forward order request with the use of the write mode of the API.
- ARGO/Accounting portal: We have to identify how we can link the service order and the accounting/monitoring data related to the order. The mechanism for automatic check of SLA targets has to be implemented.
- SPMT: the tool should be evaluated to understand which information can be used, but we need the final structure of data objects in SPMT to have a proper evaluation.

5.7.2.2   Future plans

As described in the previous sections some of the integrations are incomplete. For a short-term plan, we have to achieve the DPMT integration to fully integrate EUDAT order management workflows.

Then we have to investigate with accounting team how we can discriminate the resources used by the service requestor and then how to capture this information. The same work should be done with the monitoring information (ARGO).

And in the meantime, we have to follow the evolution of the SPMT information structure and capture eventually central information for service endpoints.

5.7.3   Integration of SVMON with GOCDB and DPMT

5.7.3.1   Summary of integration activities

We have implemented the integration of SVMON with EOSC-hub CMDB provided by DPMT and GOCDB, we have successfully collected actual configuration information provided by DPMT and GOCDB and mapped this information to the actual service versions installed at endpoints. The SVMON front provides compact views of the configuration information separately for GOCDB and DPMT.

5.7.3.2   Identified integration gaps

A list of sites is recorded in both DPMT and GOCDB, we need further modification of SVMON database to harmonize the configuration information. At the same time, we need to adopt an additional scope tag "EOSC-hub" to filter out EOSC-hub specific resources from both GOCDB and DPMT.

5.7.3.3   Future plans

Further enhancements of the SVMON planned related to the SVMON Web UI with multiple views of EOSC-hub services and their components. The extension of supported software types in SVMON client is another task, which is required to cover more instances of various EOSC-hub services.

5.7.4   Integration of SVMON with Pakiti

5.7.4.1   Summary of integration activities

We have integrated SVMON with Pakiti client, and we provided unified installation and monitoring of software versions from service instances. Pakiti client requests information from SVMON client, it posts the report generated from the collected data to the SVMON server.

5.7.4.2   Identified integration gaps

A new version of Pakiti client is unsupported via yum repository, while the old one has no SVMON related options.

5.7.4.3   Future plans

The integration of SVMON with Pakiti is complete. No further activities planned, apart from maintenance of SVMON client and regular updates.

Some service providers prefer to use SVMON client directly, without integrated solution with Pakiti. Both versions of SVMON client integrated with Pakiti and standalone one will be supported and continuously improved.

5.7.5   Integration of SVMON with B2ACCESS

5.7.5.1   Summary of integration activities

SVMON has been successfully integrated with EUDAT B2ACCESS using OAuth2.0 protocol, so that users can gain access to SVMON using B2ACCESS.

5.7.5.2   Identified integration gaps

SVMON doesn't use any group membership or role information for user authorization. It's also not yet clear if SVMON as a simple lookup service needs this information to manage the user rights.

5.7.5.3   Future plans

Investigate role-based filters and user management based on group membership information.

6 Monitoring, Accounting, Messaging and Security Tools

6.1 Overview

This chapter provides the maintenance and integration activities performed for the ARGO Availability and Reliability Monitoring Service, Argo Messaging, Accounting Repository, Accounting Portal and Security Tools. We also describe what are the next steps towards materialising the integration plan for each tool/service.

6.2 Accounting Repository

The Accounting Repository is implemented using a software collection known as APEL. APEL is a computer resource usage accounting tool that collects accounting data from sites participating in the EGI and WLCG infrastructures as well as from sites belonging to other Grid organisations that are collaborating with EGI, including OSG and NorduGrid. The accounting information is gathered from different collectors into a central accounting repository where it is processed to generate statistical summaries that are available through the EGI Accounting Portal.

A detailed description of the Accounting Repository service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

6.2.1 Maintenance activities

There have been a number of small bug fixes released for the APEL software this year as well as enhancements that were designed to reduce the effort required to run the service. A number of these were aimed at giving users more information in their APEL logs to make it easier for the user to diagnose any issues they have with publishing accounting information as well as giving support staff more detail in case the user can't resolve the issue themselves.

All the systems used to run the service have been kept patched and the latest IGTF Trust Anchor Distributions have been applied in a timely manner. Work to reduce the number of hosts used to run the service is ongoing, but progress has been made in ensuring that the oldest systems can soon be decommissioned.

6.2.2 Summary of service enhancements

The biggest enhancements to the service have been an addition to the cloud virtual machine (VM) accounting of support for long running VMs, and the addition to the grid accounting of support for a scaling factor for HTCondor-CEs. The first enhancement means that cloud VMs that run over month boundaries will now have their usage in each month assigned to the correct individual month, significantly increasing the utility of the cloud accounting data. The second enhancement was a prerequisite for adding support for HTCondor-CEs to APEL and will be of benefit to the increasing number of resource centres that are looking to move from CREAM-CE to HTcondor-CE.

The software used for transferring accounting records (SSM) now has Debian builds available and these are being included in the Cloud Middleware Distribution (CMD). Additionally, release candidates of SSM are currently in testing that support the new ARGO Messaging Service.

6.2.3 Future plans

This coming year there are two tasks that need to be completed to allow the old message broker system to be decommissioned as APEL is currently the only user of this system. The new version of the APEL SSM messaging software needs to be included in the Universal Middleware Distribution (UMD) and CMD that supports the new ARGO Messaging Service. As mentioned in the previous section, this is currently at the testing stage. Also, a new publishing monitoring system is needed to replace the current one that pushes the results to ARGO monitoring via the message brokers. The new system will provide RESTful endpoints that ARGO monitoring can query to check the publishing status of resource centres.

A number of other enhancements are planned: as SAML authentication is becoming more widespread, research is needed to ensure that the APEL record formats can support users who use this authentication method; a number of improvements to storage accounting are planned, including creating a summary record format that will produce consistent aggregations and that will reduce the volume of data that needs transferring, and increasing the completeness and utility of the EUDAT storage accounting data; work is needed to ensure that the APEL software, of which the vast majority is written in Python, is compatible with the latest versions of Python 3 as Python 2 will no longer be supported in 2020.

There is ongoing work to support the DODAS thematic service by providing a way to deploy on the fly accounting probes to report usage metrics from automatically deployed clusters. Once operational this service can be offered to other use cases.

6.3 Accounting Portal

A detailed description of the Accounting Portal service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

6.3.1 Maintenance activities

There were many maintenance activities, beginning with monitoring the SSM queues used to communicate with APEL and receive accounting data, and the loaders which put these data in the database, since those failed several times, without leaving any error logs, so manual checking was needed.

Also, database indexes needed to be created and maintained depending on the data present in the database, to make expensive queries faster. Since the Accounting Portal doesn't have to write in the database unless there is a daily update from APEL, these indexes can be made very aggressively, meaning very frequently, without penalties.

Security was improved by making some directories off limits in a way that does not expose the existence of those directories. Also a fallback was implemented that denies access to these directories with a "Forbidden" response.

The WLCG Tier1 report codebase that was previously imported from REBUS wholesale was improved and became more responsive to partial experiment (VO) selections. Also pledge selection on multi year periods was improved.

6.3.2   Summary of service enhancements

There is a fully working release of the accounting portal that support SAML and is integrated with EGI Check-in waiting for review before to be deployed into production. This will allow users without X.509 credentials to access privileged views in the portal after being logged-in through Check-in.

The main page graphs were enhanced with a cloud-based graph, and the selection and order of metrics in the cloud accounting views were changed to make the elapsed time cores metric the default one. In general, unit handling was improved, including the use of multipliers and dividers for units especially in the case of storage and memory-based metrics.

A special instance was created to facilitate EOSC-hub integration at eosc-accounting.egi.eu, since this integration is expected to remove some basic functionality of the portal to streamline it (e.g. VO handling), a separate instance allows making those changes without impacting existing requirements in the vanilla instance.

6.3.3   Future plans

The EOSC-hub integration will be further enhanced in the new EOSC instance when improved EUDAT storage data is available.

The REBUS Tier1 view will be improved to be more responsive on partial experiment selections.

A new WLCG view to integrate pledge data with WLCG accounting will be done as a first step to later integrate Tier 1 and Tier 2 data in the same view.

Further security improvements will be implemented, including a new credential mechanism.

6.4   Argo Monitoring

ARGO is a flexible and scalable framework for monitoring status, availability and reliability. It provides monitoring of services, visualization of their status, dashboard interfacing, notification system and generation of availability and reliability reports. The dashboard design enables easy access and visualisation of data for end-users. Third parties can gather monitoring data from the system through a complete API. A central deployment of the ARGO monitoring engine can serve a large infrastructure reducing the maintenance costs. A high-level architecture of ARGO is shown in Figure 6-1.

A detailed description of the ARGO monitoring service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].
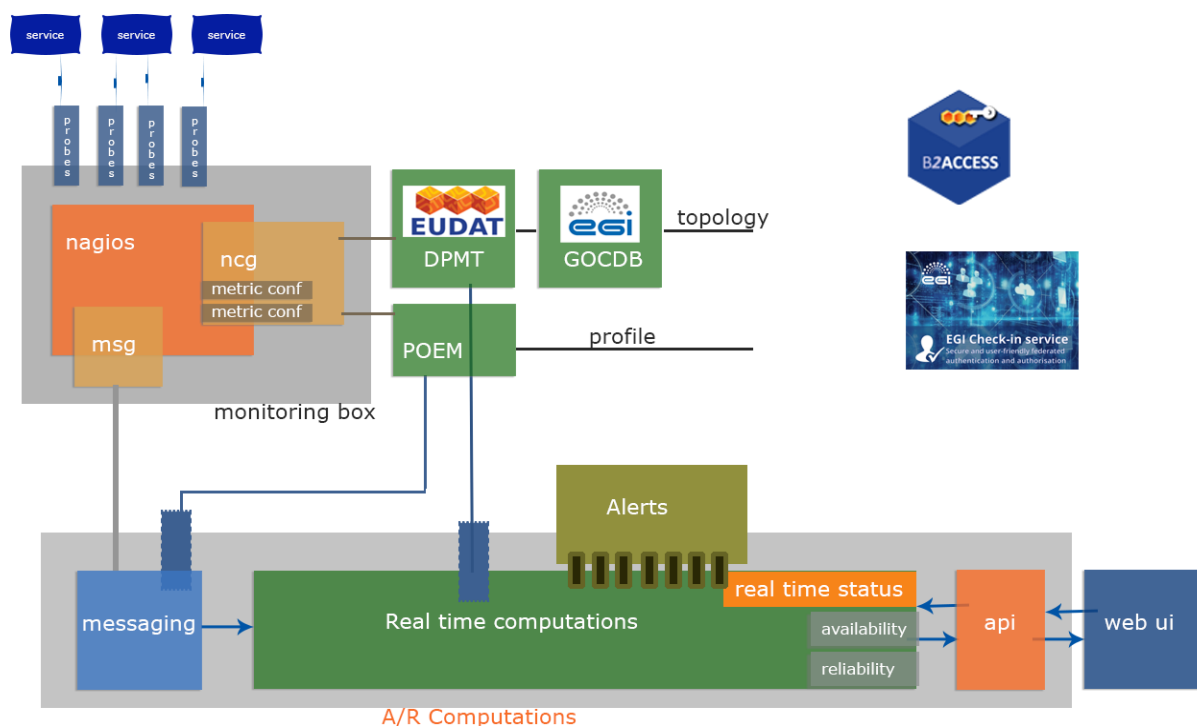
Figure 6-1 High level architecture of the ARGO Monitoring system

6.4.1 Maintenance activities

There is a standardized maintenance window every first Wednesday and Thursday of each month. These maintenance windows are used to apply regular operating system upgrades and stable releases. All necessary precautions (backing the data) are taken care of beforehand by the monitoring team.

ARGO follows a development process that includes mandatory tests for checking the functionality and the quality, correctness of the software. This process consists of automated unit tests and code quality checks, running via a CI tool (jenkins). The argo team maintains a full replica of the production deployment (devel, staging instance) that is used for development, testing and integration validation of all the components that ARGO comprises of (mon, poem, compute engine, UI).

6.4.2 Summary of service enhancements

6.4.2.1 Customer Defined Thresholds

The ARGO Monitoring Service is generating Status and A/R reports based on the metric results that it gathers from the execution of the monitoring probes. Each metric result includes a status and

performance data that typically contain values related to the provided status. Previously the ARGO Monitoring Service relied solely on the statuses returned by the probes in order to generate the Status and A/R reports. Each probe has a hard-coded built-in static logic in order to compute the probe status. Although this has been proven sufficient for the purposes of infrastructure monitoring up to now, it does not give us any flexibility in providing different SLA targets to customers. Threshold definitions allow the infrastructure owners to easily generate a multitude of reports with stricter or looser criteria tweaked exactly on their needs and use cases. Those reports are generated from the same set of original monitoring data and provide a group of different views on the same infrastructure allowing to compare numbers between different scenarios or current and possible future requirements.

By adding support for Customer Defined Thresholds we introduced a new profile type **Thresholds Profile** that is used to move the metric status computation to the ARGO Compute Engine, so the monitoring probes executed return the actual data (e.g. the average response time) and then on the ARGO Compute Engine can have multiple profiles, which will be used in order to generate reports based on customer defined thresholds.

The main format of the threshold that is used follows the following format: `'label'=value[UOM];[warn];[crit];[min];[max].` As mentioned earlier, by default its metric contains a predefined format of data (threshold). By introducing the **Thresholds Profile,** each service or service endpoint could introduce its own thresholds. The Thresholds Profile is stored as a new resource to the web-api. This profile is used in the ARGO Compute Engine to generate reports based on the new thresholds. For each metric received at ARGO Compute Engine if a performance data exist in the web api we use to run the computations otherwise Nagios status is used. At this moment we support: a) just one threshold per metric M1=T1; M2=T2; metrics are assigned to service type b) multiple thresholds per service based on additional data

This feature is based on changes from the following components of ARGO Monitoring:
- **Monitoring engine:** pass additional data to messaging and finally to other components
- **AMS publisher**: introduce the optional field with string containing additional data
- **Compute engine**: Update the computations by deciding the thresholds to use
- **WEB API**: Introduce the Thresholds Profile resource (add, update, get, delete action).

6.4.2.2   One Stop Shop

For this task we started to implement a service management web interface through which customers (e.g. VO managers, Infrastructure Managers etc) will be able to configure the monitoring service to their liking. We have already taken the first steps towards this direction with the addition

of full life-cycle management of the probes by the product team through the POEM web service and the autoconfiguration of the mon-box and configuration engine from the profiles given. The ARGO monitoring box now reads data from multiple sources a) poem, b) GOCDB or DPMT and automatically creates the needed configuration. The configuration is updated every hour to be sure that all changes from different sources are used in the configuration file. At the same time the configuration of probes and or metrics is redesigned in POEM so as to include all the necessary parameters to run the probes and metrics.

6.4.2.3 Harmonization of the user facing web interfaces

With this task our aim is to harmonise and update the look and feel of all the web interfaces that ARGO users. Apart from the Web UI all internal components are using the same interface .We have already updated the Web portal for both EGI and EUDAT views and added support for multi tenancy in the web portal that allows us to easily add a new view or report.

The Web UI is retrieving data - with the help of Lavoisier - from different sources :

- ARGO API
- GOC DB
- DPMT

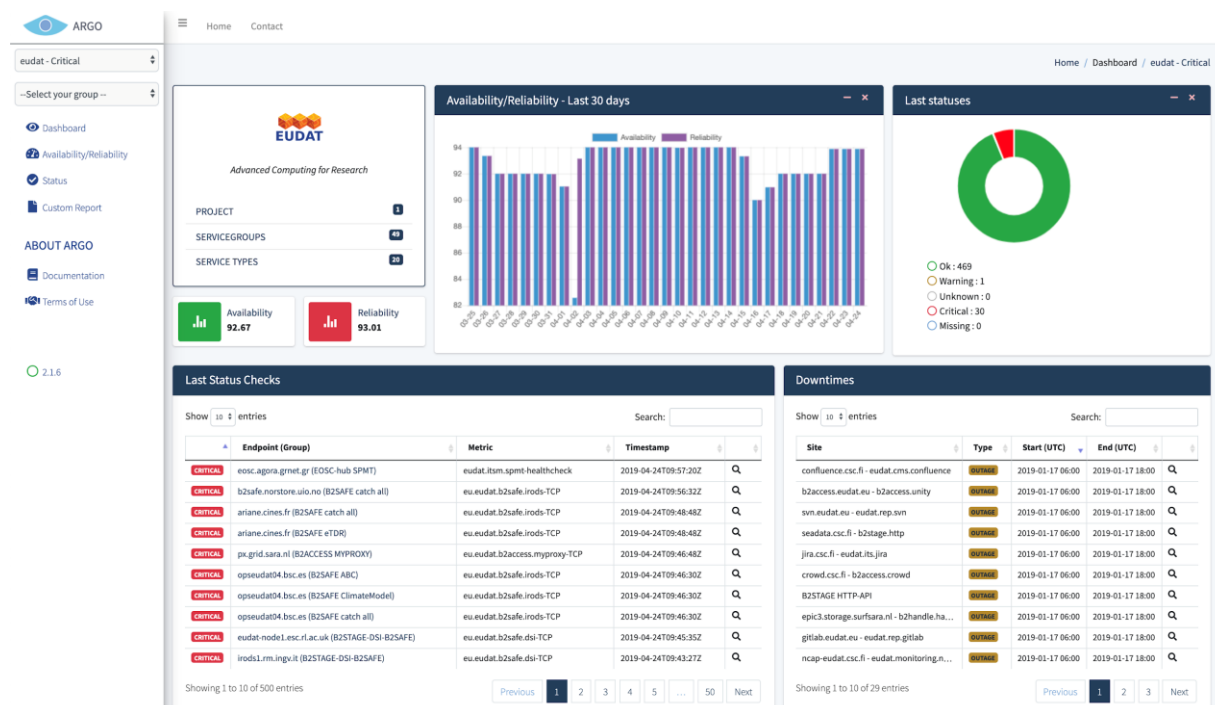Figure 6-2 shows the dashboard of ARGO Web UI.

Figure 6-2 Dashboard of ARGO Monitoring Web Portal

This page is a synoptic view for the tenant and a given report (ex. Critical.) with:

- the description of the topology - structure (project, sites,) and list of the related entries
- the results of availabilities/reliabilities for the last 30 days
- the 500 last status changes with the distribution and the details of these changes
- the downtimes affecting the the sites/service-groups

On the left side of the interface a user can change the report with the 1st list. At the same time a user can access the details of a site / service group with the 2nd list.

Figure 6-3 shows Availability/Reliability page. This kind of pages are providing Availability/Reliability reports for the given tenant and report for the last 5 months.

- Details of a given month can be accessed by clicking on the value in the corresponding month.
- Details of a sub-group can be accessed by clicking the name of the project.

A new feature is that a user may also see the A/R for endpoints

Different export formats are available (excel, csv, pdf).

Figure 6-3 ARGO Availability/Reliability page

A/R page functionalities:

- A/R values in table
- A/R values in charts
- Filter A/R report
- Search
- Export (csv, pdf, excel)
- Copy
- Jump to current Status

Figure 6-4 shows status page. Status of the sites, services, endpoints for the last 7 days .

Figure 6-4 ARGO status page

By clicking the little coloured bars, you can access the details until you reach the output details.

A custom report page is shown in Figure 6-5. A user may create his own reports - Availability/Reliability or Status - with a choice of site/service group and dates.
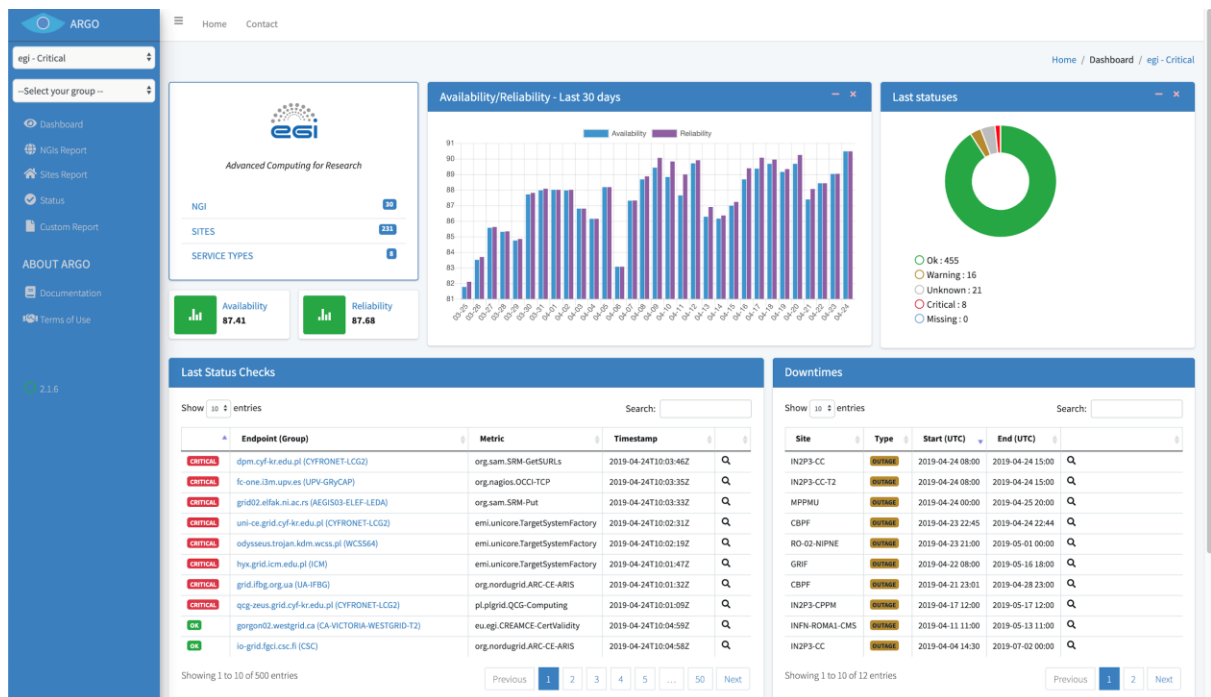

Figure 6-5 Custom report page

6.4.3   Future plans

Our aim is to continue our regular operation of Argo and to work on the 3 main goals

Customer Defined Thresholds: The next steps for this task is to extend the Argo Admin Interface to be able to handle the Thresholds profile that is already implemented and used by the ARGO Web-API and Compute engine. At the same time the WebUI will also visualize the thresholds used for each service.

One Stop Shop: Our Aim is to extend the ARGO Web-API to become the source of truth for all the components of ARGO this will allow the POEM to become the Admin interface for all ARGO Components and to be able to:

- Create a new tenant
- Modify the Profiles of an existing tenant
- Deploy or Update a Probe

Harmonization of the user facing web interfaces: The web-ui component of ARGO was updated to support multiple tenants the next step is to update the Poem interface to have the same look and feel as well. This will also allow us to decouple POEM from its backend and to integrate it with the WEB-API.

6.5   ARGO Messaging Service

The Messaging service enables reliable asynchronous messaging for the EGI infrastructure. The current implementation of the Messaging service relies on a Message Broker Network of ActiveMQ
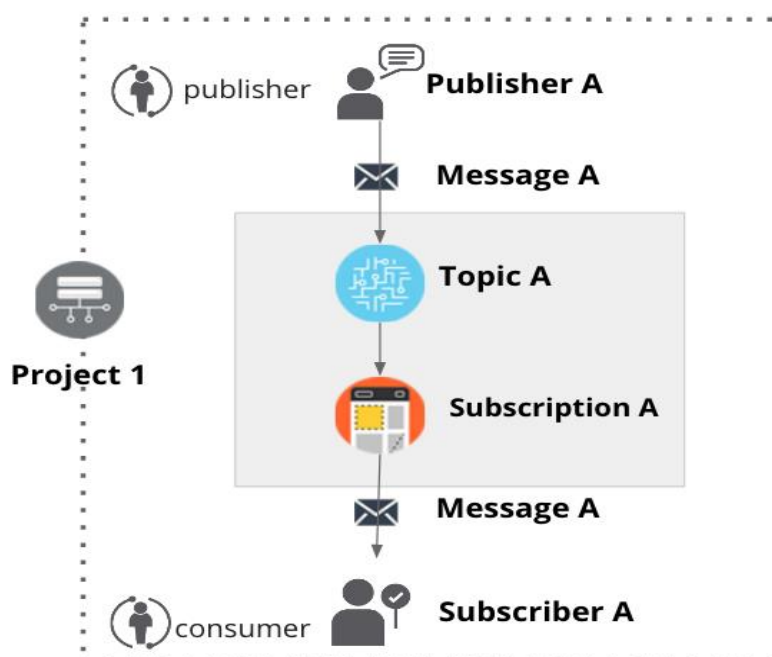

Figure 6-6 Publish/Subscribe Messaging Service

services and uses the STOMP protocol for the publication and consumption of messages.

The new version of the Messaging service is going to replace the STOMP interface with an HTTP interface which makes the implementation of new clients easier and the implementation more robust. The new ARGO Messaging Service is a real-time messaging service that allows you to send and receive messages between independent applications.

It is a Publish/Subscribe Service (Fig. 6-6), which implements the Google PubSub protocol. It provides an HTTP API that enables Users/Systems to implement message-oriented service using the Publish/Subscribe Model over plain HTTP. Publishers are users/systems that can send messages to named channels called Topics. Subscribers are users/systems that create Subscriptions to specific topics and receive messages.
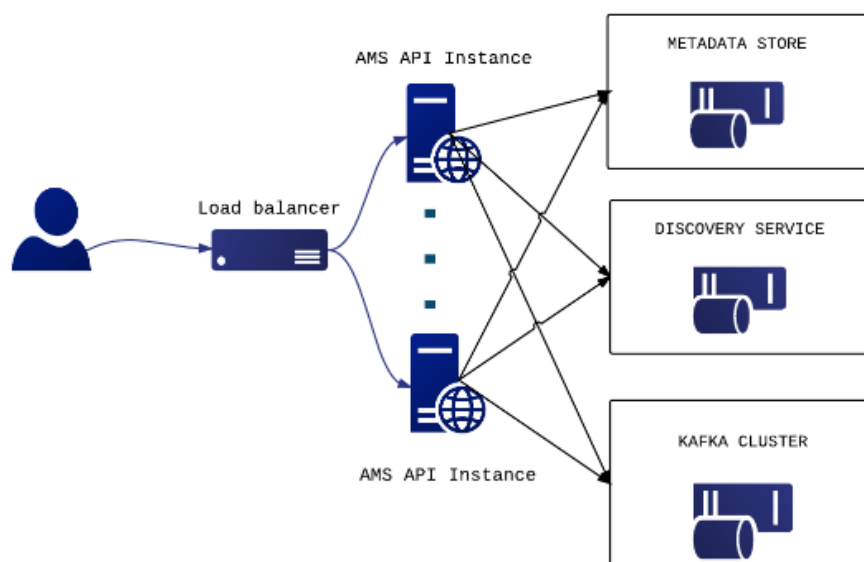
Figure 6-7 ARGO messaging service

It supports both push and pull message delivery. In push delivery, the Messaging Service initiates requests to your subscriber application to deliver messages. In pull delivery, your subscription application initiates requests to the server to retrieve messages.

Apart from all these the Messaging Service supports:

- Argo-ams-library: A simple library to interact with the ARGO Messaging Service.

- Argo-AuthN: Argo-authn is a new Authentication Service.

- AMS Metrics: Metrics about the service

As shown in Figure 6-7 current deployment of messaging service comprises a haproxy server, which acts as a load balancer for the 3 AMS servers running in the backend.

An initial description of the ARGO messaging service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

6.5.1 Maintenance activities

There is a standardized maintenance window every first Wednesday and Thursday of each month. These maintenance windows are used for applying regular OS upgrades and stable releases. All necessary precautions (backing the data) are taken care of beforehand by the monitoring team.

One major part of maintenance activities is the updates / upgrades of the software / library dependencies the AMS has. This follows a specific process where performance, features, and service stability are taken into consideration. When reliable version of a software dependency is available, the development team deploys a new stand-alone instance to test the validity of all main features and decide on a list of changes required. When a stable version is implemented, it is deployed on the development instance for at least one month until it is deployed in the production service.

AMS follows a development process that includes mandatory tests for checking the functionality and the quality, correctness of the software. This process consists of automated unit tests and code quality checks, running via a CI tool (jenkins). Unit tests that test crud and domain logic functionality on all resource objects supported by the api, using mock interfaces on the datastore and broker layers. (golang testify) At the same time AMS endpoints are tested as postman collections via newman. Newman is a command-line collection runner for Postman [R55]. This allows to effortlessly run and test a Postman Collections [R56] directly from the command-line. It is built with extensibility in mind and it can be easily integrated with ARGO's continuous integration server and build systems.

6.5.2 Summary of service enhancements

During the first year the AMS Services introduced a number of new API calls. These API calls are mainly requirements from external users or systems/clients. a) New api call get user by UUID, b) New api call status, that is used as a health check for the service, c) New api call modify a subscription's ack deadline, d) Internal mechanisms for aggregating project:metrics and topic:metrics. Until now the metrics where per topic. Instead of parsing all the topics to get the project metrics we add this mechanism while at the same time we added the support of daily, monthly metrics.

Another useful new feature is pagination support. Listable collections should support pagination, even if the results are typically small. So AMS now supports pagination for its main collections like (users,topics and subscriptions).

Finally, a new implementation for the push server started this year. An initial prototype is ready that allows to decouple the push functionality from AMS api nodes as it is in the current deployment. The push server(s) are an optional set of worker-machines that are needed when the AMS wants to support push enabled subscriptions. They perform the push functionality for the messages of a push enabled subscription (consume->deliver->ack). The new implementation provides a gRPC interface in order to communicate with AMS api [R57].

As already mentioned AMS Service also supports: a) Argo-ams-library: A simple library written in python to interact with the ARGO Messaging Service. b) Argo-AuthN: Argo-authn is a new Authentication Service. c) AMS Metrics: Metrics about the service.

The Argo-ams-library was extended to support more functionalities of the AMS Service. One of the main features was the offset manipulation that allows the client to replay or discard messages. The user can seek to an offset in the future to discard messages. To replay and reprocess previously acknowledged messages, the user may seek to a prior offset. At the same time the use of new libraries and a number of updates (libraries), fixes and tests were introduced and applied to the library.

ARGO-authN is the new service implemented and deployed in the first year. It is a mapping service between different authentication protocols. This service provides the ability to different services to use various authentication mechanisms without having to store additional user info or implement new functionalities. It allows the user to just map one authentication method (x509 certificates) to another, for example, API access tokens for AMS as shown in Figure 6-8.
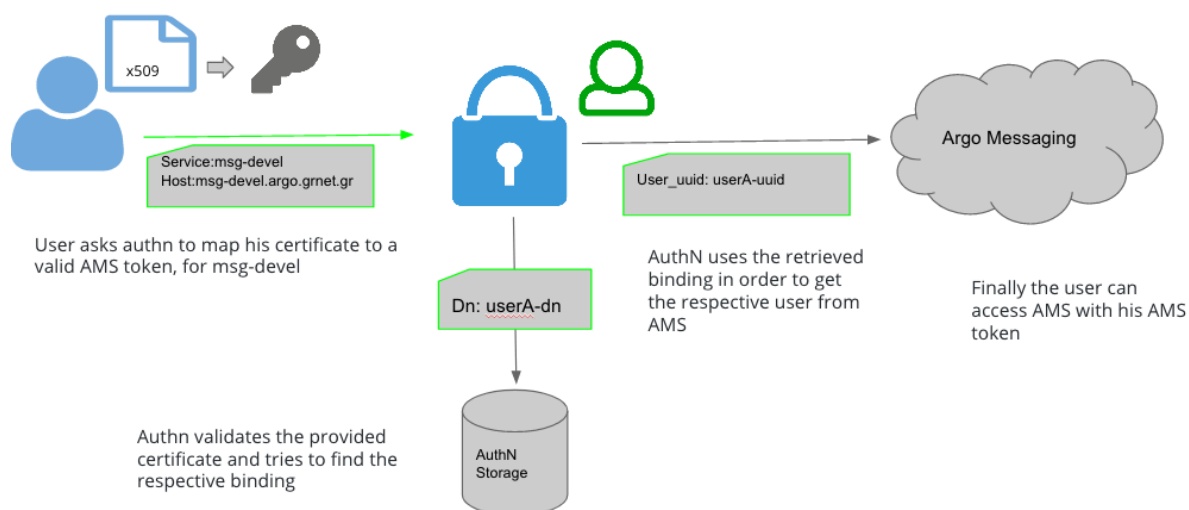

Figure 6-8 ARGO AuthN service: mapping workflow between authentication methods

It is important to mention that AuthN doesn't store other service's user credentials in order to perform the mapping. The service holds various information about a service's users, hosts, API urls, etc, and leverages them to provide its functionality. AuthN utilises three basic internal entities in order to provide this functionality: a) Service types(ams,web-api,etc) represent the different services that want to use the argo-api-authn as an external authentication mechanism, b) Bindings hold information related to how we should map specific authentication credentials, c) Auth methods hold information on how AuthN should communicate with the respective registered service types [R58].

AMS API exports usage metrics that can be monitored programmatically. Apart from Usage metrics it supports Operational Metrics that include mainly metrics related to the CPU or memory usage of the ams nodes. During the last year more metrics were added in the list and a UI was implemented to support and display these metrics.

6.5.3 Future plans

ARGO Messaging future plans are mainly to support the users the services that want to start or continue using the service such as:

- Support, maintain, extend the AMS Service
- Support, maintain, extend the AuthN Service
- Support FedCloud Information System
- Support EGI Information System
- Support AppDB

At the same time, we are also investigating new features (google pub/sub new features)[3] that will facilitate the use of the service from new services.

6.5.4 Integrations

The ARGO Messaging Service [R59] [R60], consists of 3 main resource types a) A topic is a named resource to which messages are sent by publishers (publish messages), b) A subscription is a named resource representing the stream of messages from a single, specific topic (consume messages) and c) the actual messages which is the combination of data and attributes that a publisher sends to a topic and is delivered to subscribers. The following Services rely on the AMS Service:

- **Operations Portal**: Reads the alarms from predefined topics, store them in a database and displays them in the operations portal. Whenever a service / endpoint changes status (warning, unknown, critical) the ARGO Monitoring engine raises an alarm. These alarms, which are used by the operations portal. ARGO Monitoring engine sends these alarms in predefined forma as a message to alarms topic of EGI Project in the ARGO Messaging Service. These alarms are consumed by the Operations Portal via alarm subscription as shown in Figure 6-9.

- **Accounting**: Use of AMS as a transport layer for collecting accounting data from the Sites. The accounting information is gathered from different collectors into a central accounting repository where it is processed to generate statistical summaries that are available through the EGI Accounting Portal. The software used for transferring accounting records (SSM) is using the ARGO Messaging System (in testing mode). The SSM publishes messages to predefined topics as shown in Figure 6-9.

- **FedCloud**: Use of AMS as a transport layer of the cloud information system. It makes use of the ams-authN. The entry point for users, topics and subscriptions is GOCDB. A utility python script reads the xml feed from GOCDB, creates the respective ams users under the specified project, assigns to the correct project's topic, creates a binding for each user, using the dn from GOCDB and finally creates topics with the schema SITE_sitename_ENDPOINT_id_in_gocdb. Each site publishes the necessary information to

---

[3] https://cloud.google.com/pubsub/docs/release-notes

the predefined topic and periodically this information is consumed by the corresponding subscription by AppDB info consumer.

- **ARGO Availability and Reliability Monitoring Service**: It uses the AMS service to send the messages from the monitoring engine to other components. The Monitoring Engine is using the AMS and sends all the raw metric data to the metrics topic. These messages are consumed by different components of the ARGO framework but mainly by the compute/analytics engine. The analytics engine - apart from the metrics raw data - is using data from different sources of truth. All these data are published in the AMS by different connectors and are consumed by the analytics engine so as to produce status, availability and reliability reports.
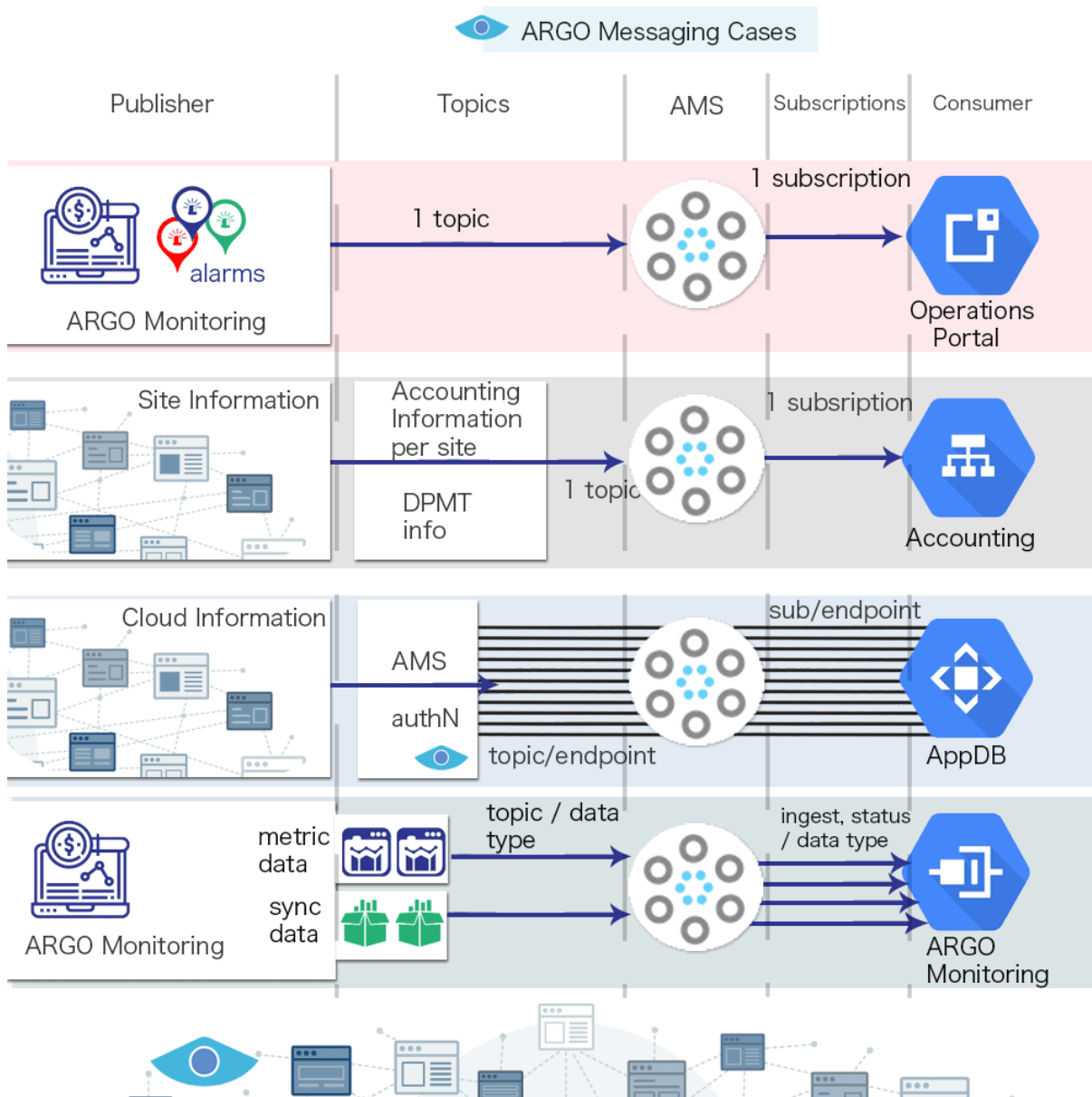
Figure 6-9 ARGO messaging service integrations

These services were or are still using the EGI Message Broker network today. The ARGO Monitoring Service is already using a connector for the new Messaging Service. Operations portal was working on a solution and testing it in their devel instance. Operations portal, Accounting, FedCloud information system and AppDB are expected to also complete the implementation of their own interfaces to the new Messaging Service, within the timeframe of the project.

The Messaging Service does not have any dependencies to other services at the moment.

6.6   Security Tools: Pakiti

Pakiti provides a monitoring mechanism to check the patching status of Linux systems. Pakiti uses the client/server model, with clients running on monitored machines and sending reports to the Pakiti server for evaluation. The report contains a list of packages installed on the client system,

which is subject to analysis done by the server. The Pakiti server compares versions against other versions which are obtained from various distribution vendors. Detected vulnerabilities identified using CVE identifiers are reported as the outcome, together with affected packages that need to be updated.

A detailed description of the Pakiti messaging service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

6.6.1   Maintenance activities

The service has been operated without notable issues for the end user. The system was regularly maintained and kept up to date with security patches.

In addition to the production instance, a pilot Pakiti service was established a while ago and security monitoring of EGI adapted to feed the instance with Pakiti records. The new service is based on the new Pakiti code that has been finalized recently. Results produced by both the instances have been correlated for the past months to detect any discrepancies or irregularities in information produced. Both services have been gradually adapted based on the findings so that the outcomes are comparable. Results produced at the moment are very similar and the new instance is being prepared for production utilization.

6.6.2   Summary of service enhancements

Development activities focused mainly of tasks related to planning transition of the production EGI Pakiti instance to new codebase. Synchronization with GOC DB roles and site/NGI information was implemented, which allows the Pakiti access control mechanisms to reflect the GOC DB settings. In order to ease the transition phase, the format of exported data was changed to be the same as what is used by the current production. The processing of CVE information provided by Linux vendors was made more resilient and open to changes in the format. Packaging rules for .deb packages were added to Pakiti client.

6.6.3   Future plans

We will continue to maintain the Pakiti service and support its users, which is mainly the EGI CSIRT and site/NGI security administrators. We will finish the evaluation of the new Pakiti version and switch it to production when it supports all current functions. We will restart communication with the maintainers of the Pakiti client package in EPEL about how to update the client package.

6.7   Security Tools: Secant

Secant is a security cloud assessment framework that is used to check the security characteristics of virtual machines and their images. The framework instantiates the machine in a contained environment and runs a set of security probes against it. The probes combine external and internal checks and aim at typical configuration error or vulnerabilities commonly misused by Internet attackers.

A detailed description of the Pakiti messaging service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

6.7.1 Maintenance activities

The service is still operated in a pilot deployment that makes it possible to evaluate results and adapt the functions accordingly. The service was regularly updated to keep synchronized with the latest development. A number of errors and minor improvements have been applied to the service in the course of the operations and its evaluation. As part of maintenance activities, utilization of the Argo messaging channels and cloudkeeper setup was handled and made up to date with the settings of other services.

The service is operated by CESNET on their cloud facilities that are based on OpenNebula. Since CESNET decided to move from OpenNebula to OpenStack, it was necessary to prepare the transition of Secant to using the new technology.

6.7.2 Summary of service enhancements

A great deal of development activities were focused on the transition to another cloud framework, which required changes to the corresponding code base. Additional changes were related to the integration with AppDB, which had to be rescheduled, though in order to finish the adaptation to OpenStack.

6.7.3 Future plans

We will finish the OpenStack support, after which the interaction with AppDB will be revisited and properly tested.

6.8   Integration activities

6.8.1   Integration of Accounting Repository with EUDAT Accounting Service

6.8.1.1   Summary of integration activities

APEL uses the Storage Accounting Record (StAR)[4] format for exchanging data on storage space accounting and so work was required to ensure that DPMT could provide data in this format.

During the course of this year, the plan for integration presented in deliverable 5.1, section 5.2.1, was modified so that instead of EUDAT pushing storage accounting data via the message brokers, the APEL service would instead regularly query a REST endpoint that DPMT already provided using a modified version of SSM to gather the accounting metrics as shown in Figure 6-10. While sites normally push storage data to the Accounting Repository using the message broker system, DPMT already provided an HTTP REST API that can be queried to extract certain information, so it was decided to make use of this interface and to also add a method for the Accounting Repository to query this interface to retrieve EUDAT storage accounting data.

---

[4] http://cds.cern.ch/record/1452920/files/GFD.201.pdf

To prevent clashes between EGI and EUDAT data, the site names provided by DPMT are now prefixed by "EUDAT-". This is necessary as the work to integrate the configuration management systems of EGI and EUDAT is still ongoing and so there is currently no single source of de-conflicted site names.
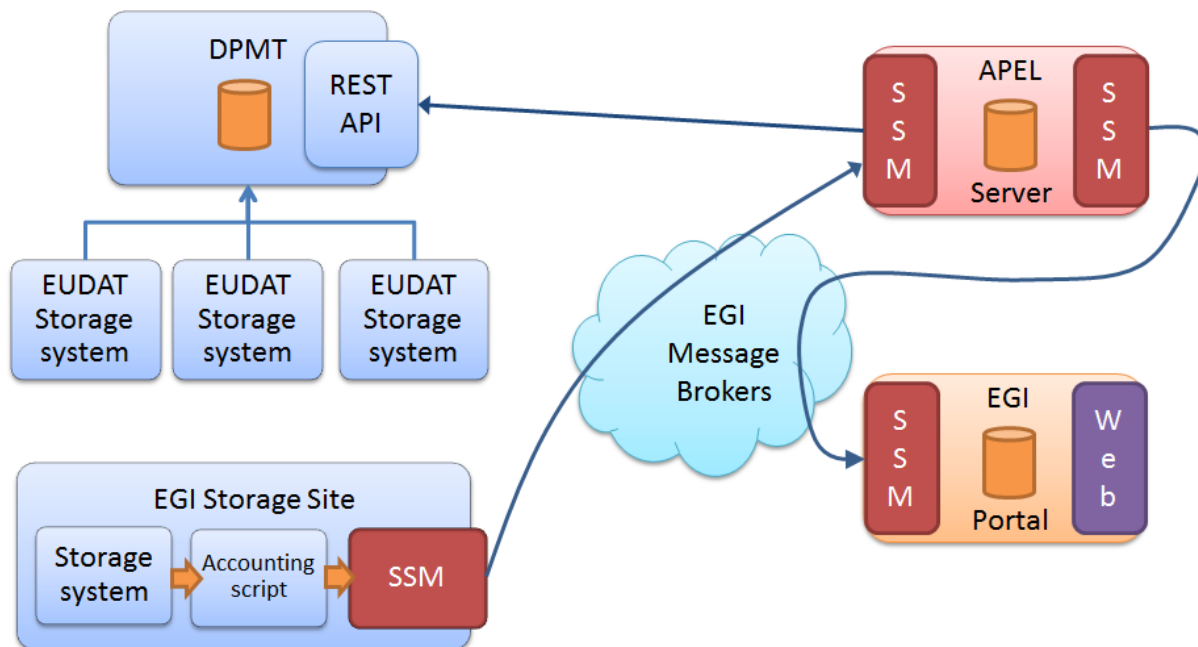

Figure 6-10 DPMT integration with the APEL central Repository and Accounting Portal

6.8.1.2   Identified integration gaps

The metrics that APEL is able to retrieve from DPMT are currently quite minimal. Work is needed to provide further metrics via the REST endpoint.

6.8.1.3   Future plans

Work is ongoing, and will be continued this year, to map internal DPMT metrics to the metrics that the APEL storage accounting record format supports.

6.8.2   Integration of Pakiti with SVMON

6.8.2.1   Summary of integration activities

The Pakiti client has been extended to support the SVMON protocol, which makes it possible to send reports to the SVMON service using the Pakiti client. Additional details are provided in Section 5.7.3 of this document

6.8.2.2   Identified integration gaps

No significant gaps were identified.

6.8.2.3   Future plans

The integration is complete.

7 Helpdesk Services and Tools

7.1 Overview

The helpdesk service provided by EOSC-hub is using the ticketing system xGUS as front-end and it is integrated with EUDAT helpdesk system (EUDAT-RT) and the EGI helpdesk system (GGUS). The description of the current maintenance activities, service enhancements and future plans for these three helpdesk services are explained in this section.

7.2 GGUS

The GGUS system is the main support access point for the EGI infrastructure. It creates a trouble ticket to record the request and tracks the ticket from creation through to solve. Users can submit a ticket via the web interface.

A detailed description of the GGUS messaging service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

In addition, information on change, release and deployment are available in the EGI wiki at [R61] [R62].

7.2.1 Maintenance activities

No major changes during the covered period have been performed. Maintenance activities of GGUS are documented in the GGUS release notes available at [R63].

7.2.2 Summary of service enhancements

Requested enhancements are collected in EGI RT system in the GGUS queue.

7.2.3 Future plans

GGUS releases take place in a bi-monthly release schedule. Releases are usually done on the last Wednesday of the release month. They are recorded and announced via GOC DB maintenance feature. All release dates are listed in EGI wiki.

7.3 EUDAT-RT

EUDAT-RT is the main contact point for all the users of EUDAT infrastructure and for the internal management or requests or incidents within the infrastructure providers. The EUDAT-RT is open to all the users through the web form [R64] or can be used directly via the web interface available at helpdesk.eudat.eu. In the second option, the users should request an account in the B2ACCESS service to be able to submit a ticket or check the status of their previous tickets.

7.3.1 Maintenance activities

The service has been in full production during this year, the only maintenance activities performed has been related to the infrastructure running the service. They were done on 7th August 2018 and implied the full stop and restart of the service.

7.3.2 Summary of service enhancements

During this year the main enhancements has been related to the integration between EUDAT-RT and xGUS, modifying the RT action scripts to permit the synchronization of the tickets between the EUDAT-RT and the xGUS ticketing systems. The integration also implied the creation of a new field in the EUDAT-RT database to keep information of the ticket-id from xGUS in order to be able to synchronize any change in the EUDAT-RT to the right xGUS ticket.

All these changes were done in the test environment and replicated in the production environment without affecting the normal production of the service.

7.3.3 Future plans

EUDAT-RT service is fully operational. The implementation of changes is done in the development version of the service and moved to production following the change management procedure of EUDAT-CDI.

7.4 xGUS

The helpdesk service for EOSC-hub project is managed by xGUS. The service is available at heldpesk.eosc-hub.eu and it is open to any user with an account on B2ACCESS or Check-in. The users can submit to xGUS any request related to services from EOSC-hub or any infrastructure under the umbrella of EOSC-hub as EUDAT or EGI.

7.4.1 Maintenance activities

Maintenance activities are coupled with GGUS maintenance activities. About xGUS releases, they take place in a bi-monthly release schedule. Releases are usually done on the last Wednesday of the release month. They are recorded and announced via GOCDB.

7.4.2 Summary of service enhancements

Implemented XML/CSV export for search results.

7.4.3 Future plans

For the xGUS instance used at EOSC-hub project the future tasks will be focused in the improvement of the usability for the end user, and the adaptation of the graphical interface with the current EOSC-hub public image. Also in the connectivity of xGUS with other ticketing systems used at EOSC-hub level, we are working on the possibility to move requests from xGUS to the JIRA used for tracking

requests in the internal services (this point is in study from the xGUS developers at KIT to understand its feasibility)

7.5 Integration activities

7.5.1 Integration of the GGUS and EUDAT-RT helpdesk tools using xGUS

The Helpdesk service for EOSC-hub (xGUS) has been integrated with the helpdesk systems of the main infrastructures under the umbrella of EOSC-hub project, EGI and EUDAT. This integration permits the movement of tickets between the infrastructures and keeping the history of the ticket in the xGUS front-end. It allows the users to have a unique contact point, the xGUS helpdesk, but the response can be delivered by the EUDAT or the EGI helpdesk's teams.

7.5.1.1 Summary of integration activities

The main integration activity has been the integration of EUDAT-RT under xGUS, as xGUS is a sub-system of GGUS (the EGI helpdesk system) and it implied the automatic integration between xGUS and EGI helpdesk.

The integration has been based on an interface between EUDAT-RT and xGUS mounted using the xGUS API and the RT action scripts, also it was needed to include an extra field in the RT database to link the ticket-id from xGUS with the ticket it from EUDAT-RT as they keep they own numbering and the link needs to be done for the synchronization. The current integration permit modifies a ticket in EUDAT and propagate this modification (state, status, comments and priority changes) to the xGUS ticket. Figure 7-1 shows current architecture of EOSC-hub helpdesk which integrates EUDAT-RT and EGI GGUS and provides a possibility to integrate other helpdesk systems of any service provider or community who is willing to join EOSC-hub and use EOSC-hub helpdesk system.
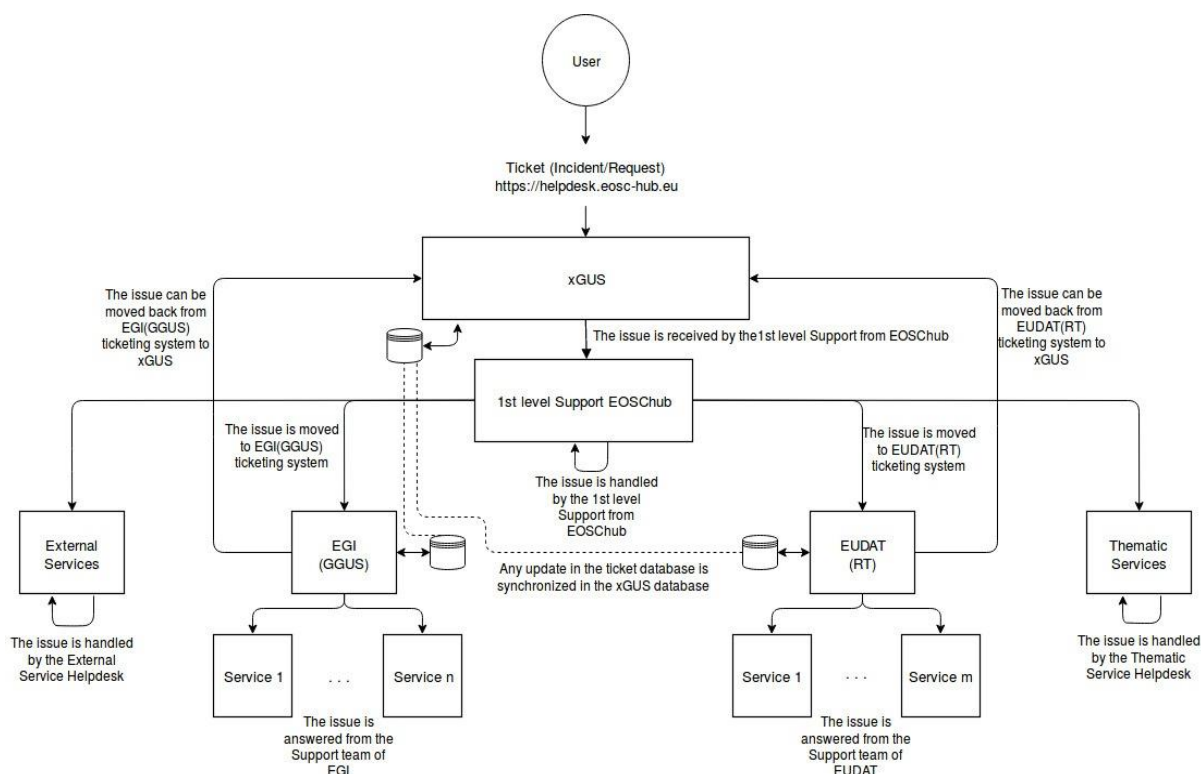
Figure 7-1 EOSC-hub Helpdesk: High-level architecture

7.5.1.2   Identified integration gaps

The main features to synchronize the tickets between xGUS, GGUS and EUDAT-RT have been achieved. The only point not fully synchronized is the modification of specific custom fields in EUDAT-RT as they are not available in xGUS, however they are mainly used for the categorization of the tickets inside EUDAT, so under an analysis, it was decided that up to now they are not required for the xGUS interface.

7.5.1.3   Future plans

The future plans are to include the helpdesk from xGUS within the EOSC-hub public web page and under the marketplace space, as the users will be able to open requests to the service when they have any problem with the EOSC-hub services. Also, to facilitate the usage of the service, we are investigating the option to permit non-registered users to create/generate requests. Up to now, the previous registration in one of the authentication systems integrated in EOSC-hub(EUDAT B2ACCESS or EGI Check-in) is required before the submission of a request, the idea is to avoid this step and permit the non-registered users to create requests.

8 Application store, Software Repositories and other Collaboration Tools

8.1 Overview

This chapter reports on the maintenance and integration activities performed during the last year, for the application store, software repositories, and other collaboration tools. A summary of enhancements these services received is also provided, along with action plans for the future.

8.2 Application Database

A detailed description of the AppDB service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

8.2.1 Maintenance activities

During the first 12 months of the project, the AppDB portal received one minor and twelve revision releases, starting with 6.1.0, up to 6.1.12. Apart from service enhancements addressed in the following sections, these releases addressed a variety of fixes pertaining to security and usability of the service, as well as other routine bug fixes.

8.2.2 Summary of service enhancements

AppDB portal
 - Integration with secant for automated VA image security assessment
 - Integration with B2HANDLE for assigning PID's to registered items via HANDLE
 - Two-way integration with OpenAIRE for OAI-PMH harvesting
 - Continuous delivery support for VAs
 - Preliminary support for GLUE 2.1

VMOps
 - Support for adding block storage to running VMs
 - Support for adding/removing public IPs to/from running VMs
 - Integration with GGUS for opening tickets related to VM deployment

AppDB-IS
 - Replaced ldap queries to top-BDII's with the AMS (Argo Messaging Service) message queue as a transport mechanism, in order to retrieve latest infrastructure (cloud) information (still in beta)
 - Preliminary support for GLUE 2.1

8.2.3 Future plans

The main plans for the AppDB portal and related services are focused on the following tasks:

 - Support OpenID connect to authenticate user actions
 - Support for per-VO user quota in VMOps

- Finalize support for GLUE 2.1 by public APIs (REST, GraphQL)
- Move AMS support in AppDB-IS to production

8.3   GitLab

GitLab is a single application which can facilitate an entire DevOps lifecycle. GitLab provides an integrated environment for software development and continuous integration. GitLab is used as an integrated solution for full software development cycle and provides rich APIs for integration with other services. Fully automated workflows for software testing and deployment implemented in GitLab can be used for efficient release and deployment management for EOSC-hub distributed services.

GitLab instance deployed at KIT is integrated with Container Registry, which allows to store Docker images. Federated access to the GItLab is provided by the EUDAT AAI solution B2ACCESS, thus GitLab resources and Git-repositories are available for many research communities and scientific organisations.

A detailed description of the AppDB service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

8.3.1   Maintenance activities

We performed regular updates according to the official GitLab release schedule. Gitlab production instance [R65] is backed up daily.

8.3.2   Summary of service enhancements

Following service enhancements were performed for GitLab service:

- Integration with B2ACCESS
- Container registry integration

8.3.3   Future plans

We will provide continuous integration tools (GitLab runners) and help the developers and service owners to test and deploy new releases automatically.   We will add more cloud resources for future usage and increase the data backup frequencies. We will keep track on GitLab official releases and provide new good features into our service. Integration plan with AppDB is still under investigation, this will provide source code repository access for any registered software item in AppDB.

8.4   EGI Software Repository

A detailed description of the EGI Software Repository service is given in D5.1 [R3]. The release notes for reporting period are provided in D5.2 [R4].

8.4.1   Maintenance activities

The administration instance of the EGI Repository frontend was migrated to a new virtualized infrastructure provided by IASA. Moreover, the RPM-based agent, the responsible unit for building

the YUM repositories got updated in order to properly support newer packages, and several fixes were made to the RSS/REST API that provides data from the backend to the frontend.

8.4.2   Summary of service enhancements

The EGI Software repository service got enhanced with a new tool which provides statistical information from the collected logs of all components. Moreover, work on employing Jenkins as a CI (Continuous Integration) solution for the specific case of UMD is in progress.

8.4.3   Future plans

Future plans for the EGI Software repository, apart from the regular maintenance tasks, including the finalization of integration with Jenkins and the possibility of moving the current frontend of UMD to the AppDB portal.

8.5   Integration activities

8.5.1   Integration of the AppDB VMops with the GGUS

8.5.1.1   Summary of integration activities

The AppDB VMOps has been integrated with the GGUS ticketing service and provides a channel for users to communicate their issues to cloud provider administrators in order to resolve them. A graphical interface has been made available, which lets users create a ticket within the GGUS system, addressing a specific VM or a topology of VMs.

First, as shown in Figure 8-1 the user locates the VM in question and clicks on the "Open GGUS ticket" and then in the pop-up dialog the "Open ticket in GGUS".
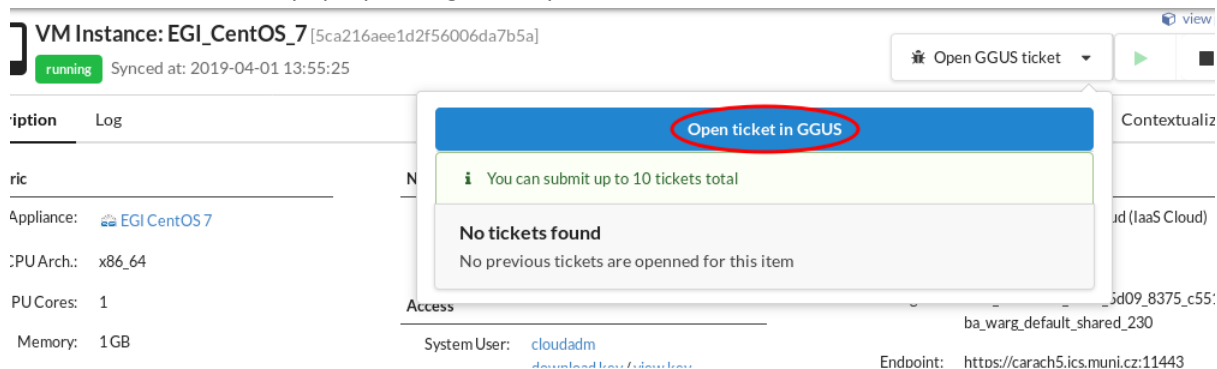

Figure 8-1 VMOps ticket creation flow

The user will be prompted to provide a description about the issue and, optionally, set its priority and extra recipients for the ticket. By clicking the "proceed" button, a ticket request gets sent to the VMOps dashboard backend.

Figure 8-2 VMOps ticket creation dialog

The AppDB VMOps portal enriches the body of the ticket (Fig. 8-2) with all the necessary information related to the specific VM, in order to help site administrators, locate and resolve the issue. The ticket submission workflow is shown in Figure 8-3. After the ticket is created, the VMOps portal properly formats it and submits it to the GGUS service by using the provided GGUS SOAP API [R66]. If no error occurs during the submission, the VMOps dashboard retrieves the ticket id and stores it within the VMOps' backend DB, associating it with the related VM.
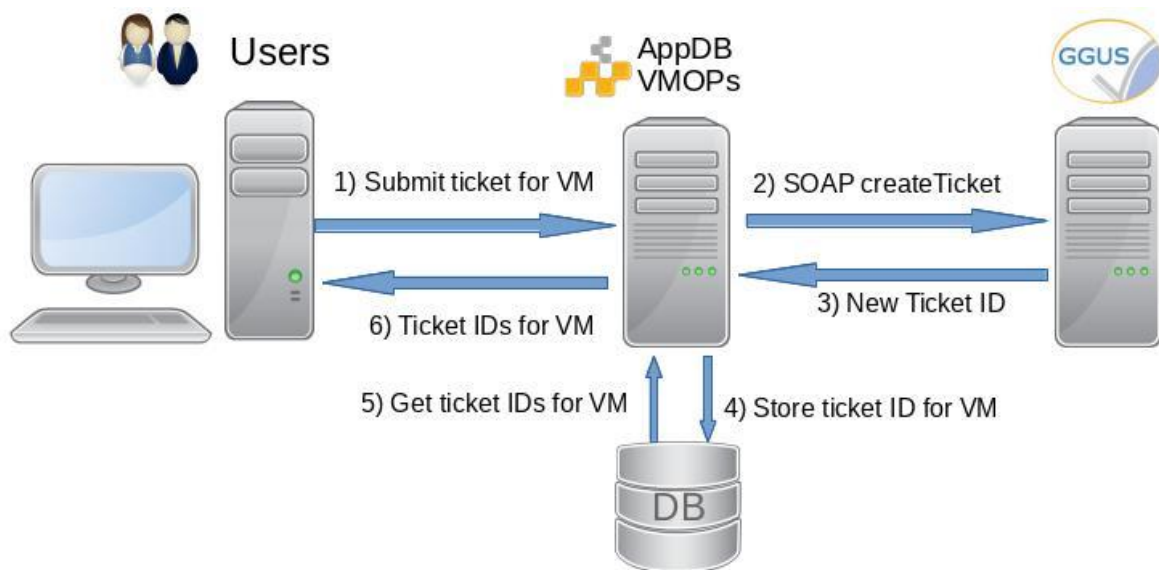
Figure 8-3 **VMOps / GGUS ticket submission work flow**

Users are able to visit and review the progress of all tickets they have created for each VM by means of the same interface as shown in Figure 8-4, at any given time. For further information, a link is provided for each ticket which refers users to the GGUS portal, where the details of the ticket may be displayed.
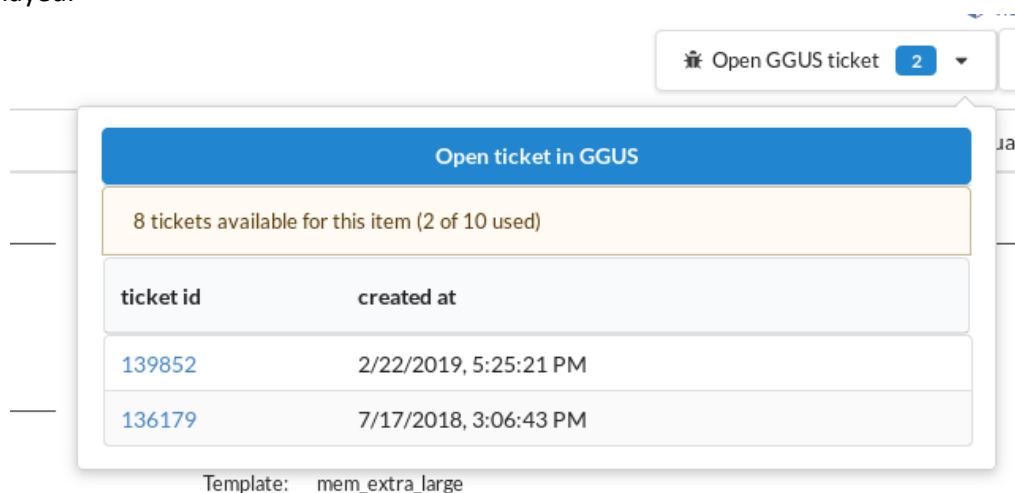
Figure 8-4 **VMOps ticket list dialog**

Currently, a user may create up to 10 GGUS tickets per VM instance. In case of failed deployment, where no VMs are available to select for submitting a ticket to, users can create a ticket at the VM topology level instead, in order to communicate to the site administrators with regard to the related deployment failure.

8.5.1.2   Identified integration gaps

Currently, the VMOps dashboard does not retrieve nor display the status of each ticket. However, this does not seem to pose a major issue since the GGUS ticketing system communicates with users via email throughout the whole ticket lifetime.

8.5.1.3   Future plans

This integration task has been completed as per the use case that was set forth in deliverable 5.1, and no related feedback has been given regarding improvements or issues with the current implementation. Addressing the gap mentioned in the previous section, pertaining to the display of ticket status from within the dashboard itself, will be evaluated based on the community's feedback.

8.5.2   OpenAIRE Integration

8.5.2.1   Summary of integration activities

In the context of collaboration with OpenAIRE-Advance, the Application Database was planned to exchange information about relevant registered products with OpenAIRE's catalogue. To this purpose, a new OAI-PMH server interface was developed [R67], which exposes metadata about registered software items and their releases, as well as virtual appliances and versions thereof, to the OpenAIRE harvesting service. The set of metadata for each type of registered product exposed was based on guidelines [R68], [R69] defined by OpenAIRE with the contribution of WP5 and which follow the datacite metadata schema. The OAI-PMH service supports the aforementioned schemata in three formats: plain datacite, oai-encapsulated datacite (oai_datacite), and the default Dublin Core (oai_dc) mandatory format, for all products. It also defines a multitude of hierarchical sets; two top-level sets, one for software items and one for virtual appliances as shown in Figure 8-5, and the rest based on their category classification. These sets make selective harvesting easier and are complementary to datestamp-based selective harvesting, which is also supported.
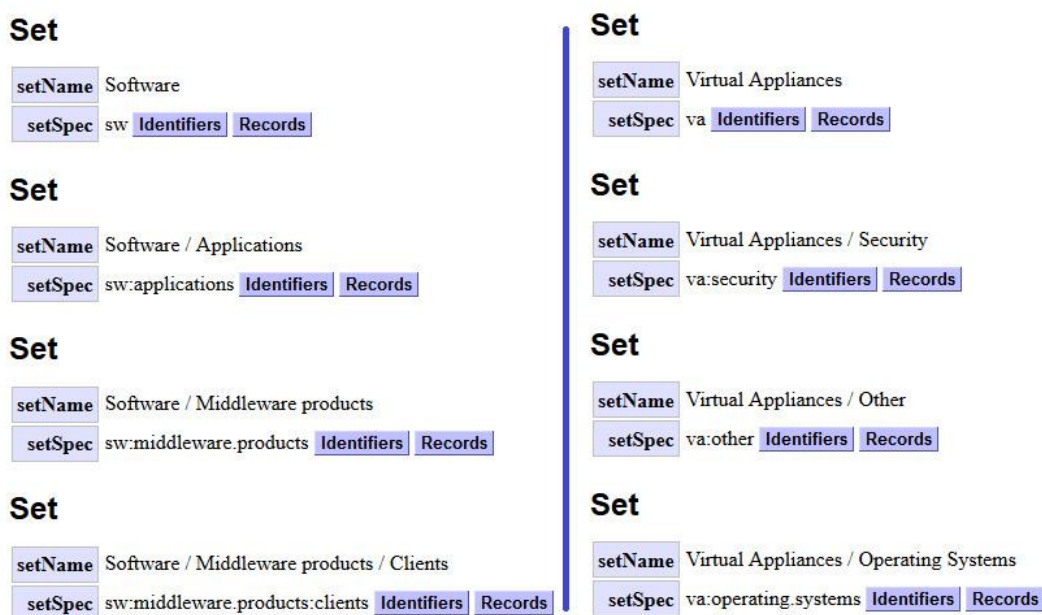
Figure 8-5 **Software and Virtual Appliance sets in AppDB's OAI-PMH service**

Data flow between OpenAIRE and AppDB is two-way as shown in Figure 8-6. On one hand, AppDB periodically fetches organization, project, funding information, etc from OpenAIRE's catalogue and adapts it to its internal schema, where it's made available for linking to software and virtual appliance records by the users. Then, these enriched records are exposed by the OAI-PMH service and harvested by OpenAIRE, which in turn stores them in its catalogue. It is worth mentioning that the identification and matching of records between the two systems is made possible through the use of HANDLE persistent identifiers (PID's). This has been accomplished by integrating with EUDAT's B2HANDLE service, using the provided PYHANDLE [R70] python client library.
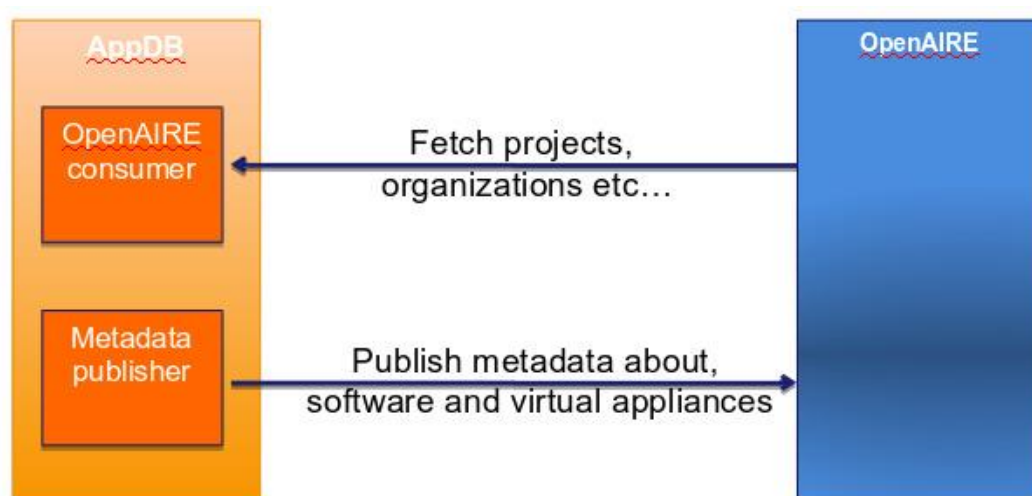
Figure 8-6 **Two-way integration between AppDB and OpenAIRE**

8.5.2.2 *Identified integration gaps*

AppDB interfaces with OpenAIRE through it bulk access OAI-PMH publisher in order to fetch organization and project data. However, the publisher does not support datestamp-based incremental harvesting, which results in time-consuming operations whenever data are to be updated.

8.5.2.3 *Future plans*

The results of the integration activities have been published in OpenAIRE's beta version (Fig. 8-7) of the explore dashboard. The next steps in completing the integration would be to identify any pending actions in order to move the harvested data into production and to investigate possible remedies for overcoming the selective harvesting issues.

Figure 8-7 **AppDB as a data repository in the beta version of OpenAIRE explore**

9 Summary

The deliverable provides a detailed overview of the progress and achievements during the first reporting period of the project. It underlines the current status of integration work for all federation services, which has been being started according to the initial plan prepared during the first half of the project year.

One of the important results of the first year is the establishment of the AAI strategy and definition of the EOSC hub AAI architecture. In addition, the initial integration of major AAI services of EGI and EUDAT has been accomplished.

A significant enhancement of Marketplace platform, its launch and started integration with other services is another achievement. It's important to mention that to provide an efficient support for Service Order and Customer Relationship Management the integration of Marketplace with other tools of Order Management System has to be finalized. This goal has been set for the next project year and will be reported in the next deliverables.

Many enhancements and integrations have been done in service monitoring and resource accounting areas. Still more effort is required for aggregation of the monitoring and accounting information from different infrastructures and its structured presentation for EOSC customers.

The initial integration of EGI and EUDAT helpdesks has been accomplished and unified EOSC hub helpdesk system is in production and ready to be operated by EOSC hub L1 support team.

Dynamic development has been performed for Application Database and its components like AppDB VMops and AppDB-IS, many accomplished integration activities with OpenAIRE have been reported.

During the first year some point-to-point initial integration scenarios and plans between services have been replaced by adoption of Messaging Service as a uniform transport layer across the core EOSC Hub infrastructure, which allows the robust, scalable integration of independent applications and services. The progress on implementation of this integration approach will be reported in further deliverables.

The WP5 roadmap is provided in the next chapter

10 Roadmap

10.1 Identification, Authentication, Authorisation and Attribute Management

The roadmap of technical and policy-related alignment activities, which have been identified across the EOSC-hub AAI services, is maintained in the project wiki [R71]. The wiki page also includes the roadmap of integration activities among B2ACCESS, Check-in, and eduTEAMS. The subsections that follow provide the roadmaps of service enhancements which are specific to each EOSC-hub AAI service.

10.1.1 B2ACCESS

2019

- New UI at login screen [Q4 2019]
- Reducing/removing steps in user login workflow between B2ACCESS and Check-in [Q2 2019]

2020

- Update of underlying software stack [Q2 2020]
- Release of group/Project management endpoint [Q2 2020]

10.1.2 Check-in

2019
- Provide uniform look and feel across all Check-in service component UIs [Q2 2019]
- Add support for (de-)provisioning and continuous update of user account information:
    - VOMS [Q1 2019]
- Add support for the OAuth 2.0 device authorization grant [R22]: This grant is designed for devices that lack a browser to perform a user-agent-based authorization [Q3 2019]
- Improve integration with Perun [Q4 2019]

2020
- Improve integration with EUDAT B2ACCESS [Q1 2020]
- Add support for retrieving Proxy certificates through SSH key information managed by the EGI Check-in COmanage Registry [Q1 2020]
- Scope-based active attribute value selection: This enhancement will allow OAuth2 clients to limit the values of specific claims based on the requested scopes [Q1 2020]
- Improve the identity linking user experience and interface [Q2 2020]
    - Improve linked identities panel by including localised friendly name & logo of user's IdP
    - Enable implicit identity linking
- Add support for (de-)provisioning and continuous update of user account information:
    - SCIM [Q4 2020]

10.1.3 eduTEAMS

2019

- Enable Sirtfi and CoCo on the eduTEAMS frontend (IdP) [Q1 2019]
- Add fail-over for Perun, COmanage and HEXAA MMS services [Q1 2019]
- Enable new generator for user identifiers [Q2 2019]
- Add new functionality to check whether IdPs release the expected attributes [Q3 2019]
- Add new functionality to whitelist SPs connected to eduTEAMS [Q3 2019]
- Enable the release of the new subject-id attribute [Q3 2019]
- Add extra scopes following the REFEDS whitepaper document on SAML to OIDC mapping: https://wiki.refeds.org/display/CON/Consultation%3A+SAML2+and+OIDC+Mappings [Q3 2019]
- Add support for multiple user identifiers in Perun [Q3 2019]
- Add support for Bitbucket as an authentication option [Q3 2019]

2020

- Upgrade to eduTEAMS v3 [Q1 2020]
- Step-up authentication service [Q1 2020]
- Enable Active Attribute Selection feature [Q2 2020]
- Integration with the Step-up Authentication Pilot Service [Q2 2020]
- New OpenID Connect Frontend [Q2 2020]
- Support for OIDC Federation specification [Q2 2020]

10.1.4 Perun

2019

- Tight integration with Check-in service [Q4 2019]
- Support user life-cycle within groups [Q4 2019]

2020

- New GUI [Q2 2020]
- Improved UX for the account linking [Q3 2020]
- Redesign workflow for account linking [Q3 2020]
- Performance optimization [Q4 2020]
- New user profile interface [Q3 2020]
- Improved integration with authentication proxies [Q4 2020]
- Pentesting [Q1 2021]
- Improved user documentation [Q3 2020]
- Automatization of operation process [Q4 2020]
- New roles and authorization module [Q2 2020]
- Improved process for the synchronization of users, attributes and group from external systems [Q3 2020]

10.1.5 WaTTS

2020

- Add fault tolerance, so that operation will not be interrupted, if once instance goes down [Q2 2020]

10.1.6 MasterPortal

2020

- Support high availability deployment [Q3 2020]

10.1.7 RCauth - Online CA

2020

- Support high availability deployment to allow distributing the service geographically across the federated operators, i.e. GRNET, STFC, and Nikhef [Q1 2020]

10.2 Marketplace and Order Management Tools

10.2.1 Marketplace

2019

- First prototype of the Marketplace Back Office to manage service entries for Service Portfolio Management Team [Q1 2019]
- Initial implementation of the functionalities supporting vouchers in Helix Nebula use case [Q1 2019]
- Improvements in MP -JIRA-Operation Portal integrations for service order management process [Q3 2019]
- Initial version of MP - SPMT integration [Q3 2019]
- Proof of Concept for 'Bundle' functionality - representation of tightly integrated services within the Marketplace [Q1 2019]
- Adjustments in the Marketplace ordering procedure to align with new requirements from the Service Order Management Team and other stakeholders [Q3 2019]
- Enhancements in user interface [2019]
- Adjusted search and filtering capabilities [Q2 2019]
- Two steps hierarchy for Projects and Issues for the Marketplace [Q2 2019]
- Marketplace Metrics for EC (Number of providers in the Marketplace, Thematic disciplines of users ordering services, Countries of users ordering services) [Q4 2019]
- Division of tickets in JIRA (master and sub tickets) according to the project scheme [Q3 2019]
- Implementation of recognised service access models (open access, external ordering, internal ordering) [Q3 2019]
- Implementation of dedicated user space to express requests for service integrations - Marketplace Projects [Q4 2019]
- Various integrations with service ordering management tool (reflecting project structure, passing service order target, allowing for ops. team hidden conversations) [Q4 2019]

2020

- Implementation of the helpdesk interface in Marketplace as a communication channel for general support of federated services [Q1 2020]

- Preparations for the MP content and graphical customizability [Q2 2020]
- Enhancements in service offer attributes - structured schema, relational architecture reflected in filtering and search capabilities [Q3 2020]
- Analysis and implementation of OCRE use cases and requirements (accounting, ordering, vouchers) [Q4 2020]

EOSC Portal developments implemented in scope of JA3:

2019

- Definition of EOSC Portal technical implementation architecture. Delivery of Portal architecture documentation
- Definition and further development of a single portal (WUI) and entry point (API interface) for Service Providers to register and maintain service descriptions
- Definition and/or further development of a single Service Database and the appropriate data APIs in which the service descriptions are maintained
- Interconnection between the Service Database and the Portal User Interface
- Optimisation of the integration of the current EOSC Portal, Service Catalogue and Marketplace and the APIs to improve the user experience from a user point of view

10.2.2 Service Portfolio Management Tool

2019

- Provide an api for the CMDBs (GOCDB and DMPT) to get list of approved the service_types [Q4 2019]
- Add In_Marketplace flag to Service Versions [Q1 2019]
- Adapt service model to support SDT v1.2 [Q3 2019]

2020

- Integrate with Marketplace [Q2 2020]
- Adapt service model to support SDT v2.0 [Q2 2020]

10.3 Integrated Business and Operations Support Systems

10.3.1 Operations Portal

2019

- Develop the Operations Dashboard to replace the different existing dashboards [Q2, 2019] - DONE
- Integration with AAI [Q2,2019]
- integrate new authorization rules - especially for users from EUDAT B2ACCESS, VO Membership
  - Use the unique identifier as reference
- Develop Service Order Management Back Office (SOMBO) to manage the Marketplace orders [Q3,2019] - DONE initial version existing

2020

- Integration with EOSC AAI [Q1, 2020]
- SLA Management Module in SOMBO [Q2, 2020]
  - For a given service order generate a document (or several documents) which will correspond to an agreement between the resource provider(s) and the customer.
  - The interface will provide different templates of documents depending on the type of resources.
- Add usage reports into the Service Order Management Back Office. [Q3,2020]
- Metrics module for EC [Q4,2020]

10.3.2 GOCDB

2019

- Development of an EOSC-hub specific view on the data in GOCDB. [Q4,2019]
- Change in the underlying infrastructure of GOCDB to improve reliability. [Q2,2020]
- IPv6 accessibility [Q2,2019]
- Implement notification mechanism, to allow users to receive notifications about Sites and Services [Q2,2019]
- Update to site map on main page. [Q2,2019]

2020

- Improve configuration management to ensure the long term stability of the service [Q2,2020]
- Improvements and modification of "Reserved Scopes" handling in GOCDB [Q2,2020]
- Creating new EOSC-Hub specific ServiceTypes automatically when they are added to the EOSC-Hub SPMT API. [Q2,2020]
- A second, configuration managed, production instance of GOCDB will be deployed behind our load balancer. [Q2,2020]
- The functionality of the Write API will be expanded to meet evolving use cases. [Q2,2020]
- EOSC-hub separate view under its own URL to show resources with the 'EOSC-Hub' scope tag applied, and we will use the current ServiceGroup functionality to represent EOSC-Hub's federated services [Q1,2020]

10.3.3 Data Project Management Tool

2019

- Implement add-on to expose REST API for integration with EOSC operational infrastructure and invocation of any action by the machine agents [Q2,2019]
- Introduction of a scope attribute for projects. This makes it possible to explicitly define the context in which a certain project is performed (EUDAT CDI, EOSC-hub, SeaDataCloud, …). It is possible to assign multiple scopes to one project. [Q2, 2019]
- Enhancement of accounting records in StAR format [Q3, 2019]

2020

- Migration to Plone 5.2 and Python 3 [Q2,2020]

- Connect DPMTs request handling to the EOSC order management via a message bus to be in line with the general switch to the AMS message bus for communication between services [Q1,2020]
- When collecting service information from SPMT: switch to SPMT API version 2 [Q2,2020]

10.3.4 Data Management Planning Tool

2019

- Rearrange API of eestore to include new information providers [Q2, 2019]
- Add new sources to the eestore to provide more extensive set of information to openDMP [Q2, 2019] - DONE
- Implement cross-platform deployment for eestore [Q2,2019]

2020

- Align the schema of EasyDMP with that of the RDA common DMP standard to demonstrate the interoperability between EasyDMP and openDMP by exporting plan from one tool to another. [Q2, 2020]
- Further services that make the data management plans machine actionable and verifiable will be developed as part of EOSC-HUB and interfaced to easyDMP and openDMP. [2020]
- Support local deployments of the eestore and extend the resources harvested (to include the SPMT/DPMT). [Q2,2020]
- To onboard the OpenDMP tool into the EOSC portfolio of services. [Q1,2020]
- Integration with DMPT and automatic procedure for update of data management plan based on the information provided in Marketplace order. [Q3, 2020]

10.3.5 SVMON

2019

- Distribution of the SVMON client among service providers [2019]
- Improvement of the codebase of SVMON client [Q1,2019]
- Integration of SVMON with DPMT and GOCDB [Q1,2019]
- Implementation of multiple views [Q2,2019]

2020

- Distribution of the SVMON client among service providers [2020]
- Implementation of token-based authentication for reporting agents [Q1,2020]
- Improvement of search and filtering inn SVMON dashboard [Q2, 2020]

10.4 Monitoring, Accounting, Messaging, Security Tools

10.4.1 Accounting Repository

2019

- Roll out support for the Argo Messaging Service (AMS) [Q2,2019]
- Ensure SAML authentication is supported in the accounting records [Q2,2019]

- Accounting for long running virtual machines [Q2,2019]

2020

- Fixes to SLUM parser [Q1,2020]
- New interface and API for publishing and synchronisation tests [Q2,2020]
- Summary storage accounting record format to produce consistent aggregations to reduce the volume of data to transfer [Q2,2020]
- Improvement of AMS integration including making certificates optional [Q3,2020]
- Enhancements to storage accounting [Q3,2020]
- Support of DODAS thematic service for deployment of accounting probes to report usage metrics from automatically deployed clusters [Q2,2020]

10.4.2 Accounting Portal

2019

- Finalise AAI integration, which is still waiting for testing [Q2, 2019]

2020

- Move Storage accounting to production after APEL update [Q1, 2020]
- Improve WLCG dedicated sections in Accounting Portal [Q1, 2020]
- Transition from AMS to SSM [Q2, 2020]
- Further Improvement of EUDAT dedicated sections in Accounting Portal [Q2, 2020]

10.4.3 ARGO Monitoring

2019

- Support, maintain, extend the Argo Monitoring service according to the workplan defined, including: One Stop Shop Customer Defined Thresholds Unified View [2019]

2020

- **Harmonization of the user facing web interface**: The new versions of the ARGO A/R and Status web interface and the POEM web interface will soon have similar look and feel [Q1,2020]
- **Single stop shop for service enablement and configuration**: This activity is designing a service management web interface through which customers (e.g. VO managers, Infrastructure Managers etc) will be able to configure the monitoring service to their liking. [Q3, 2020]
- **Customer defined thresholds**: This activity will allow ARGO customers (e.g. VO managers, Infrastructure Managers etc) to set multiple threshold profiles for each individual metric or specific service endpoint to generate reports. [Q2,2020]

10.4.4 ARGO Messaging Service

2019

- Support, maintain, extend the AMS Service [2019]

- Extension of AMS API calls [Q2,2019]
- Pagination support [Q2,2019]
- Push server to support push enabled subscriptions [Q2,2019]
- Support, maintain, extend the AuthN Service [2019]
- Support FedCloud Information System [2019]
- Support AppDB [2019]

2020

- Support, maintain, extend the AMS Service [2020]
- Support, maintain, extend the AuhN Service [2020]
- Support FedCloud information System [2020]
- Support EGI Information System [2020]
- Support AppDB [2020]

10.4.5 Security Tools

10.4.5.1.1   10.4.5.1 Pakiti
2019

- Synchronization with GOCDB roles and site/NGI information [Q2,2019]
- Integration with SVMON [Q1,2019]

2020

- Evaluation of new Pakiti service version [Q1,2020]
- Support and maintenance [2020]

10.4.5.1.2   10.4.5.2 Secant
2019

- Transition to another cloud framework [Q2,2019]

2020

- OpenStack support [2020]
- Integration with AppDB [Q3,2020]

10.5 Helpdesk Services and Tools

10.5.1 GGUS

2019

- Regular bi-monthly release schedule, security updates, implementation requested features [2019]

2020

- Regular bi-monthly release schedule, security updates, implementation requested features [2020]

10.5.2 EUDAT-RT

2019

- Finalize integration between xGUS and EUDAT-RT using the SOAP interface available on xGUS [Q2 2019]

2020

- Maintenance and regular updates, no further development or integration is planned [2020]

10.5.3 xGUS

2019

- Use registered email instead of Cert DN email during ticket submit [Q2,2019]
- Implement XML/CSV export for search results [Q2,2019]
- Full Integration between EUDAT RT and EOSC-hub helpdesk [Q2, 2019]
- Integration with EOSC Portal AAI [Q4,2019]

2020

- Integration with multiple web interfaces for request submission provided by EOSC portal, Marketplace [Q1, 2020]
- Maintenance and security patches [2020]

10.6 Application store, Software Repositories

10.6.1 Application Database

2019

- Integration with GGUS for opening tickets related to VM deployment [2018]
- AppDB integration with B2HANDLE for enriching registered digital objects with PIDs [2018]
- Two-way integration with OpenAIRE for OAI-PMH harvesting [2018]
- Support for adding block storage to running VMs [Q4,2018]

- Maintenance and support [2019]
- Allow site administrators of sites with org.openstack.nova endpoints to retrieve VO wide image lists [Q2, 2019]
- Populate VM images from site endpoints exposing native APIs (e.g. openstack) along with OCCI enabled ones [Q3,2019]
- Support predefined VO quotas per user in AppDB VMOps dashboard [Q1, 2019]

2020

- Replaced ldap queries to top-BDII's with the AMS (Argo Messaging Service) message queue as a transport mechanism, in order to retrieve latest infrastructure (cloud) information [Q1,2020]
- Provide support for OpenID Connect [Q1, 2020]
- Extend the AppDB IS to support GLUE 2.1 schema [Q1, 2020]

- Provide support for native APIs in VMOps dashboards [Q1, 2020]
- Drop OCCI support from VMOps dashboard [Q1, 2020]
- Development of the Endorser Dashboard [Q2, 2020]
- Development of the Security Dashboard [Q4, 2020]

10.6.2 GitLab

2019

- Maintenance, user support and regular updates [2019]

2020

- Support and maintenance of entire DevOps cycle [2020]
- Increase resources for the instance from 6GB to 16GB [Q1, 2020]

10.6.3 EGI software repository

2019

- Integration of Jenkins Continuous Integration (CI) with the EGI software repository [Q3, 2019] - DONE
- Maintenance and support [2019]
- Restored proper RSS syndication for internal product releases (CAs, SAM) [Q4, 2019] - DONE

2020

- Move from EGI SSO to EGI AAI authentication for admin instance [Q1, 2020]
- Provide a new front-end for the UMD / CMD and internal production software repositories, based on the AppDB portal codebase. [Q4, 2020]
- Move from RT to some other system, such as JIRA or AMS, for the software provisioning process. [Q4, 2020]

11 References

No	Description/Link
R1	https://aarc-project.eu/
R2	https://www.geant.org/Projects/GEANT_Project_GN4
R3	https://documents.egi.eu/public/RetrieveFile?docid=3344&version=2&filename=EOSC-hub%20D5.1%20final.pdf
R4	https://documents.egi.eu/public/RetrieveFile?docid=3418&version=1&filename=EOSC-hub%20D5.2%20FINAL.pdf
R5	http://www.unity-idm.eu/
R6	https://b2access.eudat.eu/home/
R7	https://edugain.org/
R8	https://aai.egi.eu/registry/
R9	https://aarc-project.eu/guidelines/aarc-g002/
R10	https://software.internet2.edu/eduperson/internet2-mace-dir-eduperson-201602.html
R11	https://wiki.refeds.org/download/attachments/38895621/20181011-OIDC-WP.pdf?version=2&modificationDate=1539611807924&api=v2
R12	https://tools.ietf.org/html/draft-ietf-oauth-token-exchange-16
R13	https://wiki.refeds.org/download/attachments/38895621/20181011-OIDC-WP.pdf?version=2&modificationDate=1539611807924&api=v2
R14	https://aai.egi.eu/sshkeys
R15	https://aai.egi.eu/registry/ssh_keys/add/
R16	https://www.geant.org/Services/Trust_identity_and_security/Pages/eduTEAMS.aspx
R17	https://aarc-project.eu/guidelines/aarc-i044/
R18	https://aarc-project.eu/policies/sirtfi/
R19	https://www.indigo-datacloud.eu/identity-and-access-management
R20	https://rcauth.eu/
R21	https://github.com/CESNET/perun/releases
R22	https://www.cilogon.org/
R23	https://www.eugridpma.org/guidelines/online-cas/
R24	https://refeds.org/category/research-and-scholarship

R25	https://aarc-project.eu/guidelines/aarc-g049/
R26	https://aarc-project.eu/guidelines/aarc-g025/
R27	https://aarc-project.eu/guidelines/aarc-g027/
R28	https://wiki.refeds.org/display/ASS/REFEDS+Assurance+Framework+ver+1.0
R29	https://refeds.org/profile/sfa
R30	https://refeds.org/profile/mfa
R31	https://aarc-project.eu/guidelines/aarc-g021/
R32	https://aarc-project.eu/guidelines/aarc-g029/
R33	https://aarc-project.eu/guidelines/aarc-g031/
R34	https://aarc-project.eu/guidelines/aarc-g041/
R35	https://tools.ietf.org/html/draft-ietf-oauth-token-exchange-16
R36	https://wiki.refeds.org/download/attachments/1606087/GEANT_DP_CoCo_ver1.0.pdf?version=1&modificationDate=1450367740260&api=v2
R37	https://aarc-project.eu/guidelines/aarc-g040/
R38	https://refeds.org/wp-content/uploads/2016/01/Sirtfi-1.0.pdf
R39	https://wiki.geant.org/download/attachments/123766285/WISE-SCI-Baseline-AUP-V1.0.1-draft.pdf?version=1&modificationDate=1557297275149&api=v2
R40	https://kcjh3g.axshare.com/#g=1&p=my-services-list
R41	https://fitsm.itemo.org/
R42	https://eosc-hub-demo.agora.grnet.gr/api/v2/services/
R43	https://messaging-devel.argo.grnet.gr/
R44	https://dp.eudat.eu
R45	https://plonerestapi.readthedocs.io/
R46	http://cds.cern.ch/record/1452920/files/GFD.201.pdf
R47	https://github.com/EUDAT-DPMT/pcp.contenttypes/commits/master
R48	https://plone.org
R49	https://opendmp.eu
R50	https://www.re3data.org/
R51	https://eestore.paas2.uninett.no/api/
R52	https://www.scc.kit.edu/dienste/5926.php
R53	https://gitlab.eudat.eu/jie.yuan/pysvmon

R54	https://pypi.org/project/svmon-client/
R55	https://getpostman.com
R56	https://www.getpostman.com/docs/collections
R57	https://api-doc.argo.grnet.gr/argo-messaging/
R58	https://api-doc.argo.grnet.gr/argo-api-authn/
R59	http://argoeu.github.io/messaging/v1/
R60	https://api-doc.argo.grnet.gr/
R61	https://wiki.egi.eu/wiki/GGUS
R62	https://wiki.egi.eu/wiki/GGUS:Main_Page
R63	https://ggus.eu/index.php?mode=release_notes
R64	https://www.eudat.eu/contact-support-request
R65	https://gitlab.eudat.eu
R66	https://wiki.egi.eu/wiki/GGUS:SOAP_Interface_FAQ
R67	http://oai.appdb.egi.eu/oai/?verb=Identify
R68	https://software-guidelines.readthedocs.io/en/latest/application_profile.html
R69	https://guidelines-other-products.readthedocs.io/en/latest/application_profile.html
R70	https://github.com/EUDAT-B2SAFE/PYHANDLE
R71	https://confluence.egi.eu/display/EOSC/AAI+Roadmap