An EOSC-hub proposal for EOSC Technical Architecture

This document is an extract of the D10.4 EOSC Hub Technical Architecture and standards roadmap v2

Introduction

The aim of the work presented in this document is increasing the added value provided by EOSC and fostering its uptake through the definition of a reference Technical Architecture for EOSC that facilitates access to services, lower barriers to integrate and composes services and promotes the usage of services between adjacent communities. This is achieved identifying key technical functions, named building blocks in the rest of the document, for each of the EOSC service category (Federation, Access Enabling, Common and Thematic) and defining related technical specifications that include an **high-level architecture, suggested EOSC standards and APIs and interoperability guidelines**. In this way, EOSC 'compliant' services will offer well-established and documented interfaces for usage and integration, based on well-known standard or APIs, facilitating:

- their exploitation from user communities willing to create new scientific services that could rely on well-established and documented interfaces for the integration. An example of exploitation of EOSC services is when a community creates a new scientific workflow re-using EOSC federation and common services, like AAI, accounting, Cloud orchestrator and/or data management solutions.
- the combined usage of EOSC services, indeed the adoption of well-known standards and interfaces will very-likely reduce the cost to integrate services. For example, two accounting infrastructures can be made easily interoperable if they use the same standard usage record format, in such case accounting data extracted from them can be merged and presented in a unique view. Another example is about data processing and data management services implementing compliant interfaces that enable a jointly usage by a thematic services.

As a consequence, less mature or small scientific communities can leverage on EOSC services for a series of IT functions and focus on their scientific work, access to scientific services will be open to new communities thanks to the documented interfaces and new scientific workflows can be created combining existing applications.

Defining the EOSC Technical Architecture

This section details the EOSC-hub proposal for the EOSC Technical Architecture. The architecture presented is a reference architecture where service categories, building blocks and related interfaces are identified. It focuses on the concepts of service interoperability and composition, fostering the

definition and the adoption of EOSC standards and interfaces. EOSC-hub is proposing an implementation of this reference architecture as described at the end of this document.

Reference Architecture

As stated above, the EOSC Technical Architecture presented in this section is reference architecture. In the field of software architecture or enterprise architecture, reference architecture provides a template solution for architecture for a particular domain. It includes a common vocabulary with which to discuss implementations, often with the aim of stressing commonalities. A reference architecture often consists of a list of functions, some indication of their interfaces (or APIs) and interactions with each other and with functions located outside of the scope of the reference architecture[1].

Reference architectures can be defined at different levels of abstraction, in the context of EOSC, EOSC-hub decided to work at the infrastructure/technical level. As part of this work, we are also defining a common vocabulary that can be used to define both existing services and those joining EOSC catalogue in the future. The architecture includes functions, interfaces, APIs and standards as technical concepts, with the final aim of fostering interoperability and, ultimately, service composability. It is based on a hierarchical structure with three levels. These are:

1. Category (the service categories introduced earlier).
2. Functional categories within the main category.
3. Individual building blocks usable in fulfilling these functions.

An overview is seen in Figure 1.

[1] S. Angelov, P. Grefen and D. Greefhorst, "A classification of software reference architectures: Analyzing their success and effectiveness," 2009 Joint Working IEEE/IFIP Conference on Software Architecture & European Conference on Software Architecture, Cambridge, 2009, pp. 141-150.
doi: 10.1109/WICSA.2009.5290800
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5290800&isnumber=5290660

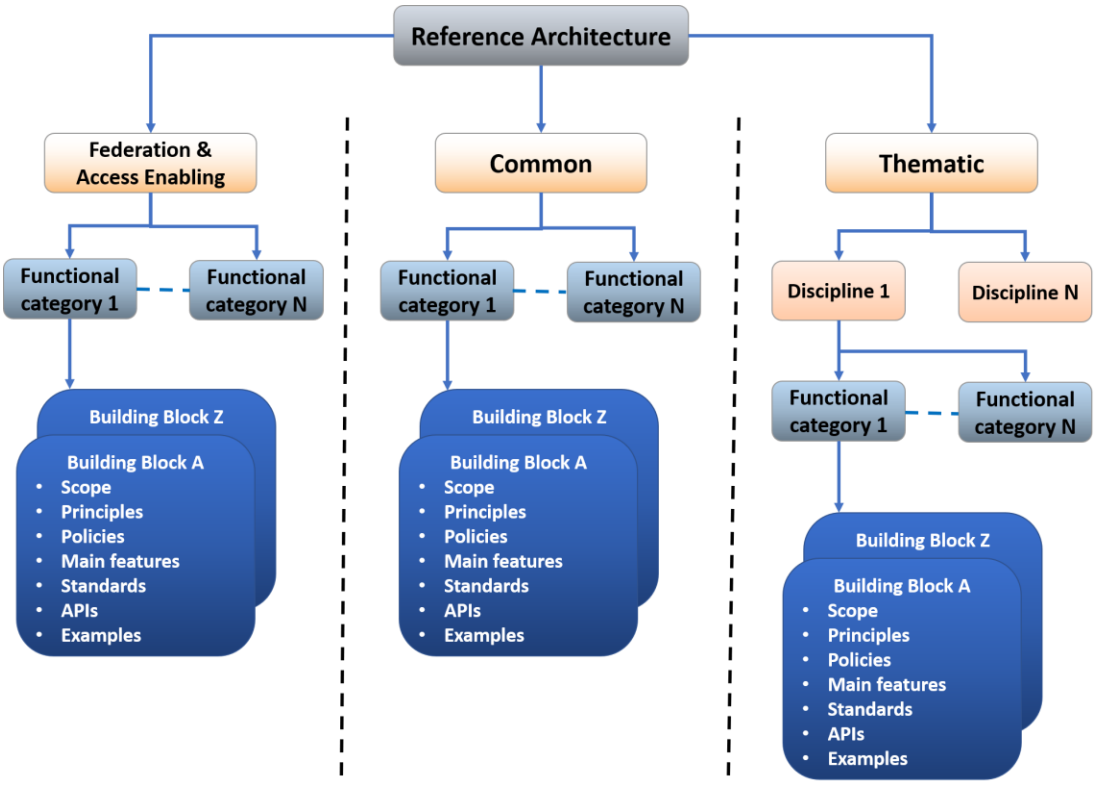

Figure 1. Hierarchical structure in the EOSC reference architecture.

The subdivision in categories allows differentiating services according to their function within EOSC: The top level categories (Federation & Access Enabling, Common and Thematic) have already been introduced. The second level of the hierarchy introduces the functional categories that groups technical functions to facilitate their identification. To take an example, within Federation and Access enabling services, we may see Authentication & Authorisation or Monitoring as functional categories. In the case of thematic services, the functional categories are identified per scientific discipline.

Beneath this, we see the individual building blocks that implement technical functions. To continue the example, within the Authentication & Authorisation functional category, we see the AAI building block.

The reference architecture described in this section gives flexibility on defining the second and third level of the hierarchy, functional categories and building blocks. Last section of this document presents the implementation of this reference architecture proposed by EOSC-hub where functional categories and building blocks are started to be defined for each service category.

Approach to define building blocks

EOSC-hub is working on defining the building blocks of the architecture for each service type and has specified a common approach to complete this task. It foresees the identification of the main **building blocks/technical functions** in each service category and, for each of those, defining a **technical specification** that includes an **high-level architecture, suggested EOSC standards and APIs and interoperability guidelines**. This method would allow providers offering services implementing the

technical function of a given building block to be compliant with the related EOSC technical specification. As a consequence, thanks to the provided guidelines, interoperability between services offering the same technical function(s) and following the EOSC specifications will be easier to achieve. Hence sets of services implementing the same building block and compliant with the EOSC specification can be made able to work together with less effort, to deliver a given technical function in the EOSC environment. Examples of these service families can be AAI services compliant with the AARC blueprint architecture and guidelines or monitoring and/or accounting systems able to exchange/share information and provide integrated views to the EOSC customers and service providers. This approach is tailored to the varied environment seen in EOSC, where many solutions to satisfy a given technical requirement already exist.

Furthermore, the definition of EOSC standards and APIs along with related interoperability guidelines for each of the identified building blocks will foster the end-to-end composition of services. Being compliant with a specification for a given building block, would allow a service to interoperate with other services offering the same function (as described above) and, conversely, building blocks/services offering different technical functions can interoperate thanks to the EOSC interfaces, described in the technical specification (e.g. it would be easier for a thematic service integrating a common service if clear interoperability guidelines are available).

EOSC interoperability specifications are not intended to be mandatory, but being compliant with them would be an added value for services. Indeed they could interoperate with other services with less effort and reduced cost. Therefore, providers willing to expand their user base by making their services composable will be inclined to support such specifications.

In this approach, identifying building blocks and the respective technical specifications could be a complex and long work, so the consortium has agreed to follow an iterative approach, starting from the functions that are more requested by the EOSC use cases[2]. Technical specifications, initially prepared by the technical experts within the EOSC-hub project, should also be iteratively improved, collecting feedback by external people with expertise in the area and involving them in the maintenance and evolution of such specifications. The same is true for the list of building blocks: they will evolve and change in the future, adding/removing functions depending on the user requirements and on the projects/service providers that may join the EOSC in the future. This will be an ongoing, continuous activity that should be continued within EOSC after the end of the project.

Technical Specification template

We have defined a template to collect information about each of the identified building blocks and define a technical specification, regardless of the service category they belong to. It is structured as follows:

[2] EOSC-hub is taking into account in this work requirements collected from EOSC Pilot Scientific Demonstrator (see D5.6 Evaluation Report of service pilots), EOSC-hub Thematic Services (see D7.2 First report on Thematic Service architecture and software integration), EOSC Competence Centers (see D8.1 Report on progress, achievements and plans of the Competence Centres) and EOSC use cases identified through the EOSC Portal (see the EOSC-hub Community Requirements Database).

- Introduction: short description of the building block highlighting its main functions.
- High-level Service Architecture: reference architecture of the building block, highlighting the interfaces towards the other building blocks. It does not refer to any specific service.
- Adopted Standard: list with references of the main adopted standards and protocols/API.
- Interoperability guidelines: describe how services implementing this building block can be made interoperable.
- Examples of solutions implementing this specification: list of already available services that are compliant with this specification.

Proposed EOSC Technical Architecture

Figure 2 shows the functional view of the proposed EOSC technical architecture, as implementation of the reference architecture described in the previous section, where the interactions between services belonging to different categories are highlighted.

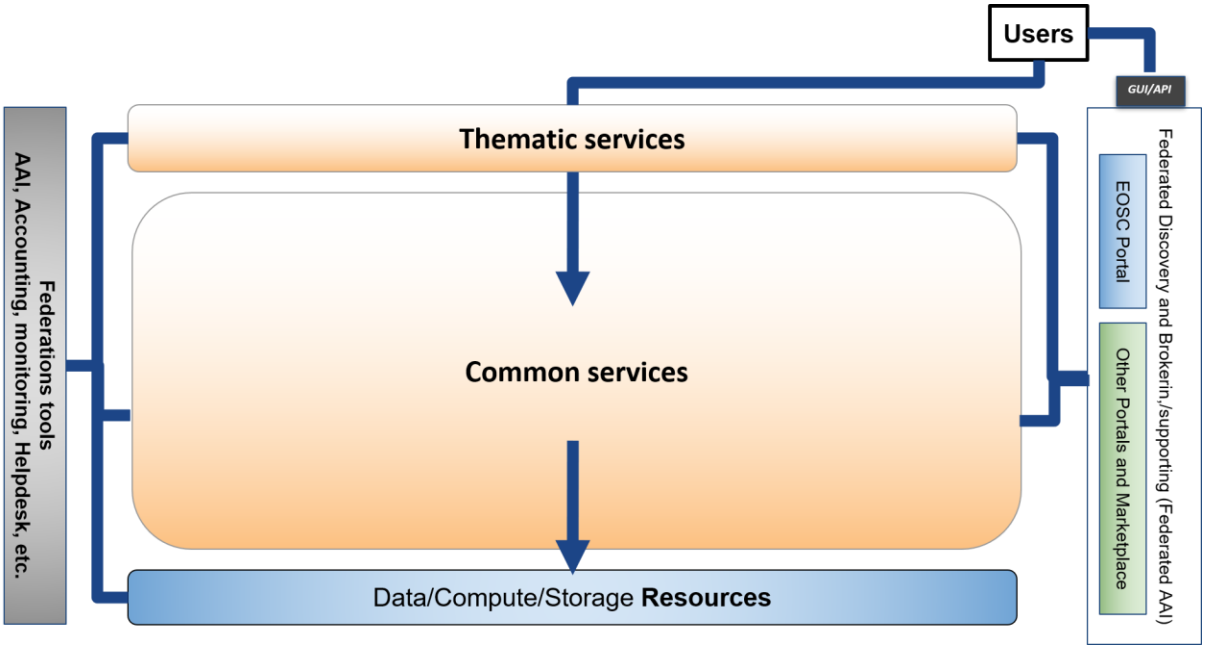

Figure 2. EOSC Technical Architecture - Functional view.

EOSC users can exploit EOSC Thematic and Common services directly or through the GUI or API of an access enabling services like the EOSC Portal (or others portals and Marketplaces). Thematic services can leverage Common services for added value features on top of data, compute and storage resources. Federation tools support all these services providing basic features like authentication and authorisation, accounting, monitoring, etc. Pledged shared resources centrally managed by EOSC, including both commodity services and service capacity, are part of the Resources and complement other EOSC resources directly managed by other service providers.

This functional view is better detailed in the D10.4 with information on the already identified and defined building blocks per service category.