Bilag 2
Integration Portal – software development project

Specification and Requirements

Content

# 1   Integration portal – Software Development

The purpose of this project is to let one university or a consortium of universities drive and host the development of a shared self-service and dashboard with integration into the national HPC centers but also for others like LUMI. This is a pure software development project where the bidder will have to work closely with the DeiC HPC centers to ensure a smooth integration process. The project should include a start-up workshop with the HPC centers to more closely define use-cases and user requirements.

The description below is a draft example, and the final design and architecture will depend on the results of the workshop. Therefore this is primarily intended as a guide when submitting an expression of interest.

1.1   Overview

The goal of the project is to make it as easy to use the national HPC centers as AWS-, Azure- and Google cloud service. Like easy way to get access to type 1 machine, add job to queue, get jobs from queue for type 2-4, manage data (share with partners, delete, upload to site etc.). It will also provide an overview of "how well" the system runs for Front and Back Office.

A high-level view of the integration work showed below.
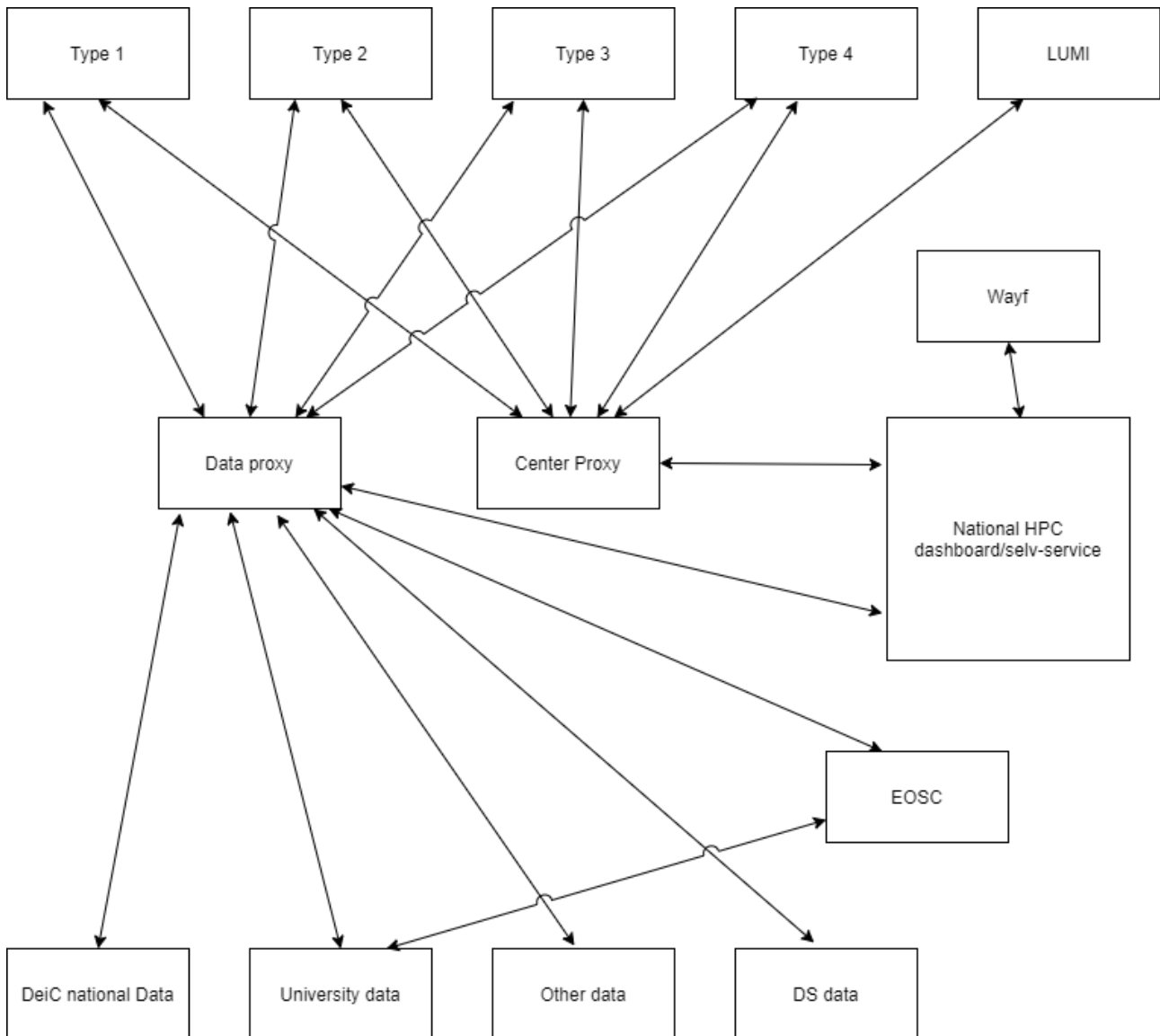
The main idea is to use indirection to allow the development of the system as the different parts are finalized. The data proxy is presently not in scope of this project but will later be included.

The project will consist of two work packages

- *Design and create a web portal.*
- *Intergrade and develop a common API that will make it easy to integrate new sites to the portal. All the integration work is done at the proxy and the portal will talk to the proxy service with a clean and stable API.*

1.1.1 Center Proxy

As of today nearly all the HPC centers provide different ways of added jobs to a queue, remove from queue, list software on the nodes, etc.

The aim of the center proxy is to provide the user with a simple and uniform API that can be used to create a self-service portal like most cloud solutions have today. The aim is also to provide the HPC center hosting

universities a way to provide resource reporting, support help tools for back-office, announce free resources etc.

1.1.2   Data proxy

Today data is place on many different sources with different interfaces and different ways of managing the data. "European Open Science Cloud" (EOSC) and other will help closing the gap and make it easier to use and get access to different kinds of data.

The goal of the data proxy is to create an uniform access API for all the different data solutions that are used today. We want to provide the users of the Danish HPC centers of with a simple and open API. This API will provide access to all data in a simple and transparent way. The user do not need to know how the EOSC API or the university storage API.

In the diagram EOSC is shown to be called from the proxy and interface with a data store that supports the API. Over time as EOSC have integrated or all storage solutions have implemented the EOSC API the data proxy can be removed to be replace with the EOSC API.

1.2   Software

- *Source code must be placed in a DeiC git repository and must be in a way that follow the principle of open source.*
- *Source code must follow basic principle of modem software projects. Have unit tests, integration tests and good documentation.*
- *Programming language must be one with a well-known track record, easy to recruit for etc. like net core 3.1 etc.*
- *Tools for tracking bugs and report new bugs is necessary.*

1.3   Hosting

Hosting is not limited to one university but will at least run on the hosting university.