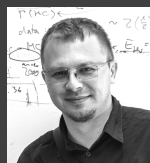
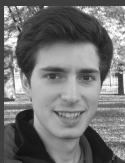


# EOSC-Performance: find most suitable EOSC site for your task

EGI Conference 2022

Karlsruhe Institute of Technology (KIT-SCC):

Borja Esteban Sanchis, Marcus Hardt, Valentin Kozlov, Christophe Laures



# EOSC Synergy in a nutshell



To expand **EOSC** by leveraging

- **Investments**
- **Resources** of **national** digital infrastructures
- Existing **experiences & know-how**

Foster EOSC with **Software and Service Quality**

Thematic services in **Astrophysics, Biomedicine, Earth Observation, Environment**

**Increase** in the number of **resources, services**, and data **repositories** offered to **researchers** through EOSC

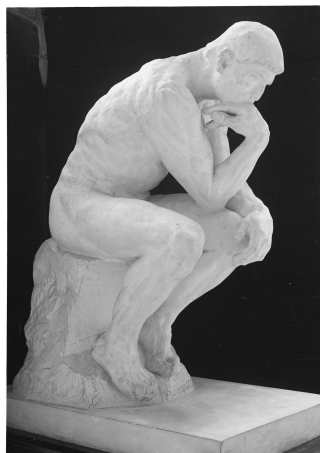


Spain, Portugal, UK,  
Czech Republic, Germany,  
Slovakia, Poland, Netherlands

*It is in the **general interest** of **users** and **service providers** to **compare** the available **computing resources***

# Introduction (aka Motivation)

Example *user stories* for comparing computing resources:



THE THINKER - AUGUSTE RODIN  
OF ART

**As:** a domain scientist **US1**  
**I want:** to **compare** various computing resources available in EOSC  
**So that:** I can choose **most suitable** resources to run my own software  
**Ask for:** **well-structured and searchable information**

**As:** an advanced user / a resource provider **US2**  
**I want:** to **store results** of any benchmarks of interest  
**So that:** I can **compare** resources for the metrics of interest  
**Ask for:** a platform to **store results of any benchmark of interest**

# EOSC-Performance

# Key Developers



Christophe Lares  
FRONTEND DEVELOPER



Marcus Hardt  
PROJECT MANAGER

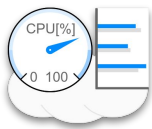


Borja Esteban Sanchis  
BACKEND DEVELOPER



Valentin Kozlov  
TECHNICAL LEAD

# EOSC-Perf: in brief

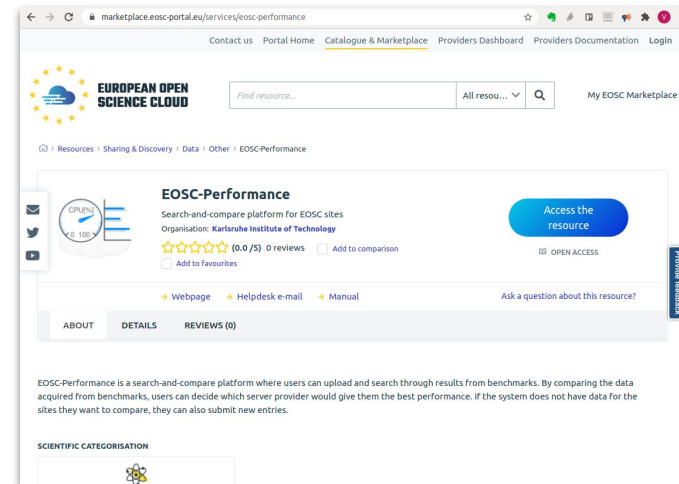


**EOSC-Performance** is a **web application** to **search, compare** and **submit** results from **benchmarks** run on a variety of different machines (single, cloud, HPC).

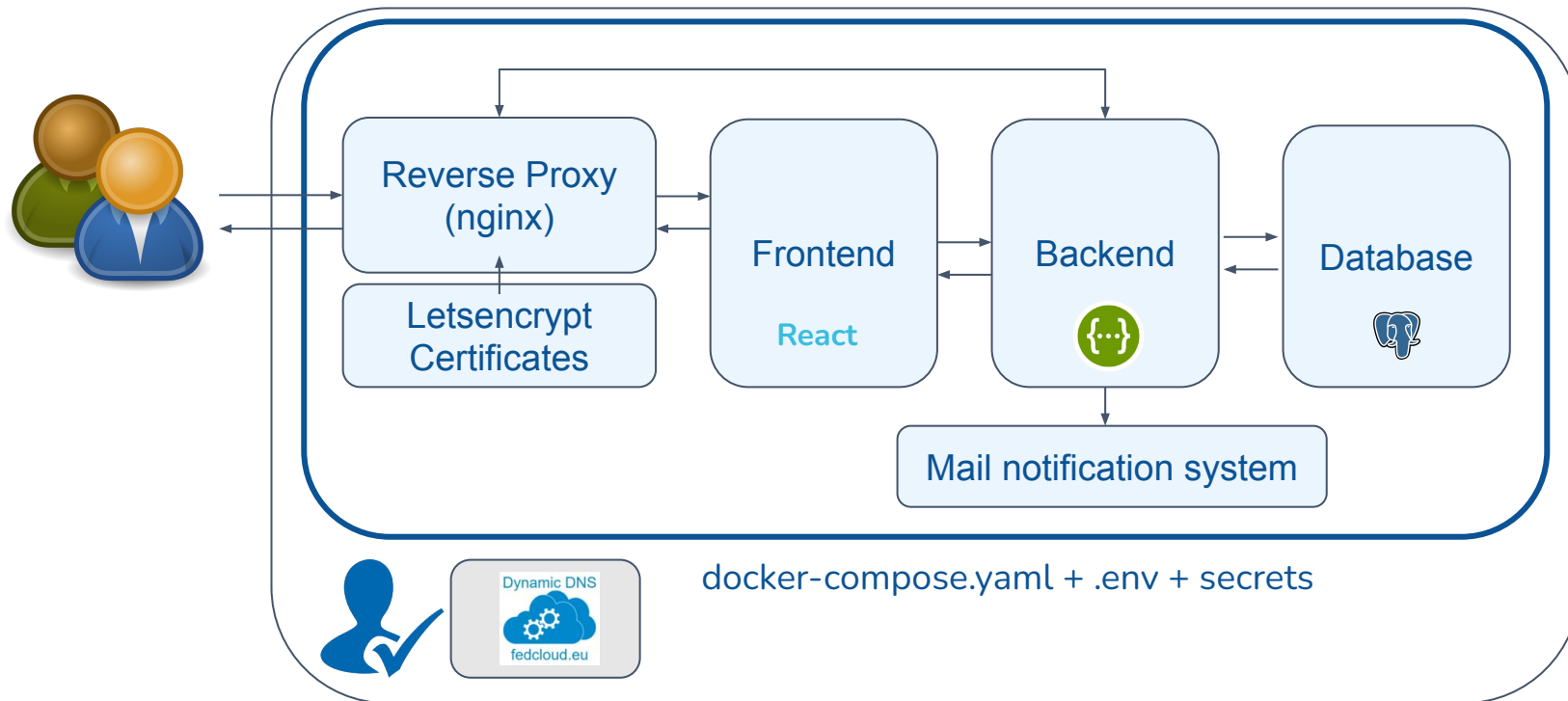
**Benchmark:** is a code packed in a Docker container, available on the Docker Hub, and produces JSON output.

Main endpoint: <https://performance.services.fedcloud.eu/>

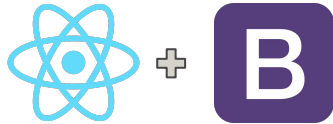
It is available in the [EOSC-Marketplace](#):




# EOSC-Perf: structure under the hood



# EOSC-Perf: structure under the hood





## Frontend, User interface:

- Responsive website developed using  React and  react-bootstrap



## Backend, API:

- Written entirely in  Flask
- Based on  OpenAPI spec v3
- Swagger interface
- Communication between Database and Interface
- Handles authentication using Flaar, OIDC and EGI-Check-In



## Database/Model:

- Powered by  SQLAlchemy
- PostgreSQL 





# Web Frontend

Note: This demo will use a non production endpoint for the demonstration:

<https://perf-stage.test.fedcloud.eu>

# EOSC-Perf: Supported Flows

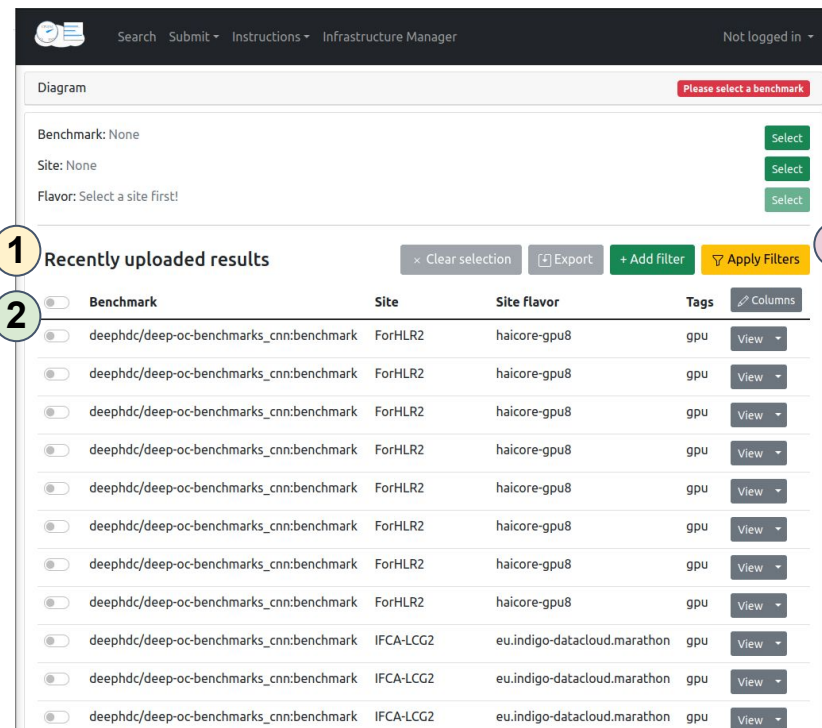
**US1:** well-structured and searchable information

Any user can:

Browse the existing results 1

Compare (table, plotting) 2

Filter them 3



The screenshot shows the 'Infrastructure Manager' interface. At the top, there's a navigation bar with 'Search', 'Submit', 'Instructions', and 'Infrastructure Manager'. A 'Not logged in' status is shown on the right. Below the navigation bar, there's a 'Diagram' section with a red button 'Please select a benchmark'. Underneath, there are labels for 'Benchmark: None', 'Site: None', and 'Flavor: Select a site first!', each with a green 'Select' button. The main section is titled 'Recently uploaded results' and includes buttons for 'Clear selection', 'Export', 'Add filter', and 'Apply Filters'. A table follows with columns for 'Benchmark', 'Site', 'Site flavor', 'Tags', and 'Columns'. The table contains 10 rows of data. The first 8 rows have 'ForHLR2' as the site, and the last 2 rows have 'IFCA-LCG2'. Each row has a 'View' button next to the 'Tags' column.

Benchmark	Site	Site flavor	Tags	Columns
deepcdc/deep-oc-benchmarks_cnn:benchmark	ForHLR2	haicore-gpu8	gpu	View
deepcdc/deep-oc-benchmarks_cnn:benchmark	ForHLR2	haicore-gpu8	gpu	View
deepcdc/deep-oc-benchmarks_cnn:benchmark	ForHLR2	haicore-gpu8	gpu	View
deepcdc/deep-oc-benchmarks_cnn:benchmark	ForHLR2	haicore-gpu8	gpu	View
deepcdc/deep-oc-benchmarks_cnn:benchmark	ForHLR2	haicore-gpu8	gpu	View
deepcdc/deep-oc-benchmarks_cnn:benchmark	ForHLR2	haicore-gpu8	gpu	View
deepcdc/deep-oc-benchmarks_cnn:benchmark	ForHLR2	haicore-gpu8	gpu	View
deepcdc/deep-oc-benchmarks_cnn:benchmark	ForHLR2	haicore-gpu8	gpu	View
deepcdc/deep-oc-benchmarks_cnn:benchmark	IFCA-LCG2	eu.indigo-datacloud.marathon	gpu	View
deepcdc/deep-oc-benchmarks_cnn:benchmark	IFCA-LCG2	eu.indigo-datacloud.marathon	gpu	View
deepcdc/deep-oc-benchmarks_cnn:benchmark	IFCA-LCG2	eu.indigo-datacloud.marathon	gpu	View

# EOSC-Perf: Supported Flows

**US1:** well-structured and searchable information

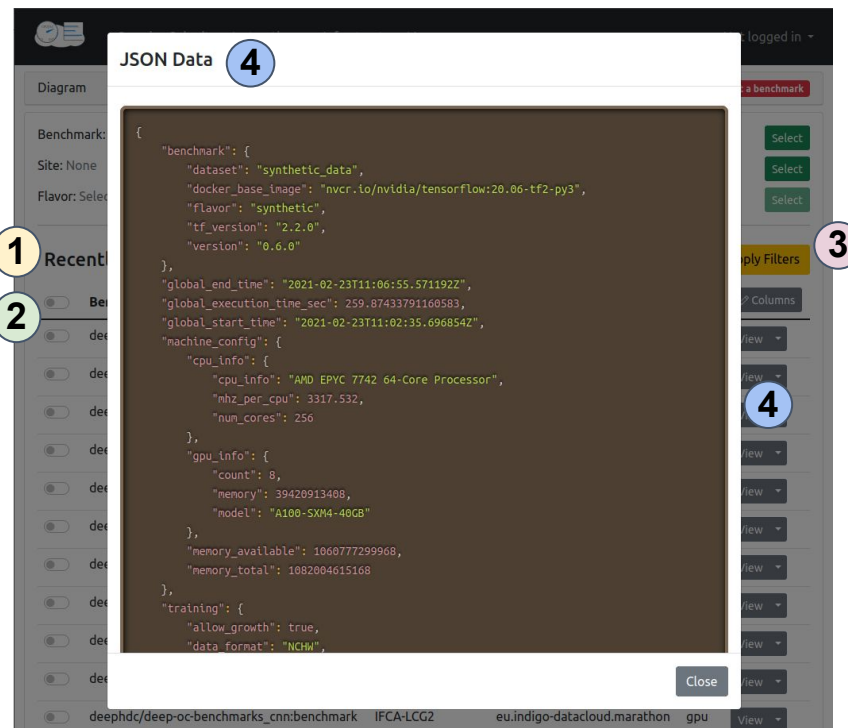
Any user can:

Browse the existing results 1

Compare (table, plotting) 2

Filter them 3

View full JSON output 4



The screenshot shows the EOSC-Perf web interface. A modal window titled 'JSON Data' is open, displaying the following JSON output:

```
{
  "benchmark": {
    "dataset": "synthetic_data",
    "docker_base_image": "nvcr.io/nvidia/tensorflow:20.06-tf2-py3",
    "flavor": "synthetic",
    "tf_version": "2.2.0",
    "version": "0.6.0"
  },
  "global_end_time": "2021-02-23T11:06:55.571192Z",
  "global_execution_time_sec": 259.87433791160583,
  "global_start_time": "2021-02-23T11:02:35.696854Z",
  "machine_config": {
    "cpu_info": {
      "cpu_info": "AMD EPYC 7742 64-Core Processor",
      "mhz_per_cpu": 3317.532,
      "num_cores": 256
    },
    "gpu_info": {
      "count": 8,
      "memory": 39420913408,
      "model": "A100-SXM4-40GB"
    },
    "memory_available": 106077299968,
    "memory_total": 1082004615168
  },
  "training": {
    "allow_growth": true,
    "data_format": "NCW"
  }
}
```

The interface includes a 'Recent' section on the left, a 'Filters' section on the right, and a 'Close' button at the bottom right of the modal.

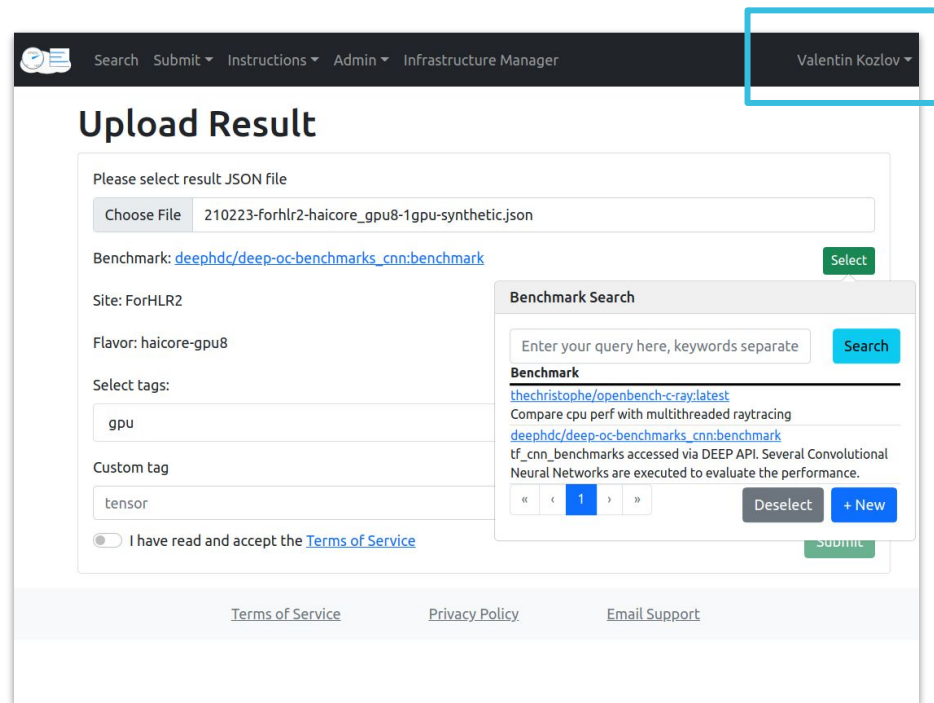
# EOSC-Perf: Supported Flows

**US2:** store results of  
any benchmark of interest

**Contributors (⇒ authentication):**

Upload a new result:

- JSON file
- Benchmark
- Site
- Site Flavor
- Tags



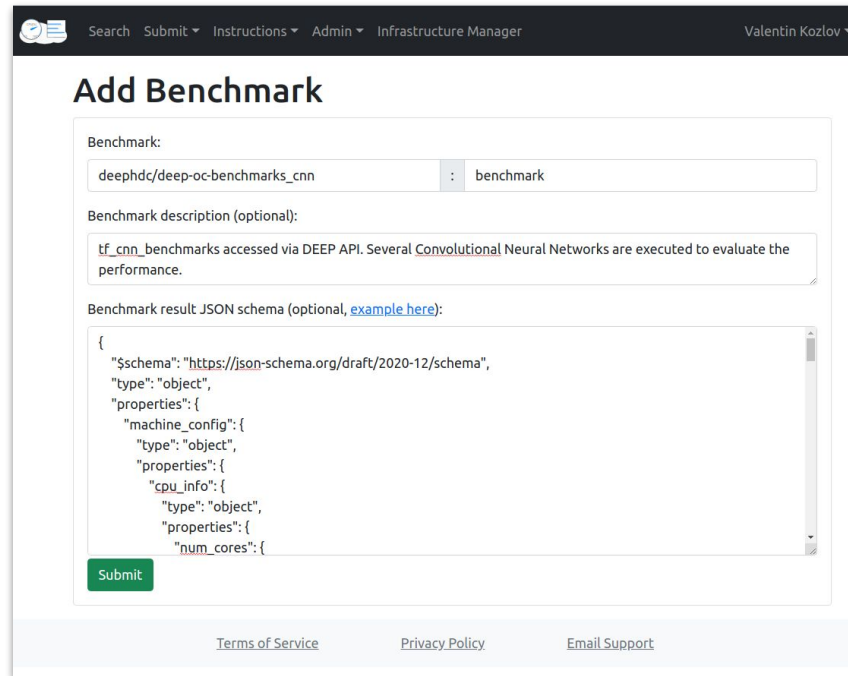
# EOSC-Perf: Supported Flows

**US2:** store results of  
any benchmark of interest

## Contributors (⇒ authentication):

Add a new benchmark:

- Benchmark:  
Dockerhub image + docker tag
- Benchmark description
- JSON schema  
(may include  
"suggestToUser" key)



The screenshot shows the 'Add Benchmark' form in the EOSC-Perf web application. The form is titled 'Add Benchmark' and is located under the 'Infrastructure Manager' section. It contains three main input fields: 'Benchmark', 'Benchmark description (optional)', and 'Benchmark result: JSON schema (optional, [example here](#))'. The 'Benchmark' field contains the text 'deepfdc/deep-oc-benchmarks\_cnn' and 'benchmark'. The 'Benchmark description (optional)' field contains the text 'tf\_cnn\_benchmarks accessed via DEEP API. Several Convolutional Neural Networks are executed to evaluate the performance.' The 'Benchmark result: JSON schema (optional, [example here](#))' field contains a JSON schema for a machine configuration. At the bottom of the form is a green 'Submit' button. The footer of the page contains links for 'Terms of Service', 'Privacy Policy', and 'Email Support'.

**Add Benchmark**

Benchmark:

deepfdc/deep-oc-benchmarks\_cnn : benchmark

Benchmark description (optional):

tf\_cnn\_benchmarks accessed via DEEP API. Several Convolutional Neural Networks are executed to evaluate the performance.

Benchmark result: JSON schema (optional, [example here](#)):

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "type": "object",
  "properties": {
    "machine_config": {
      "type": "object",
      "properties": {
        "cpu_info": {
          "type": "object",
          "properties": {
            "num_cores": {
```

Submit

[Terms of Service](#) [Privacy Policy](#) [Email Support](#)

# EOSC-Perf: Supported Flows

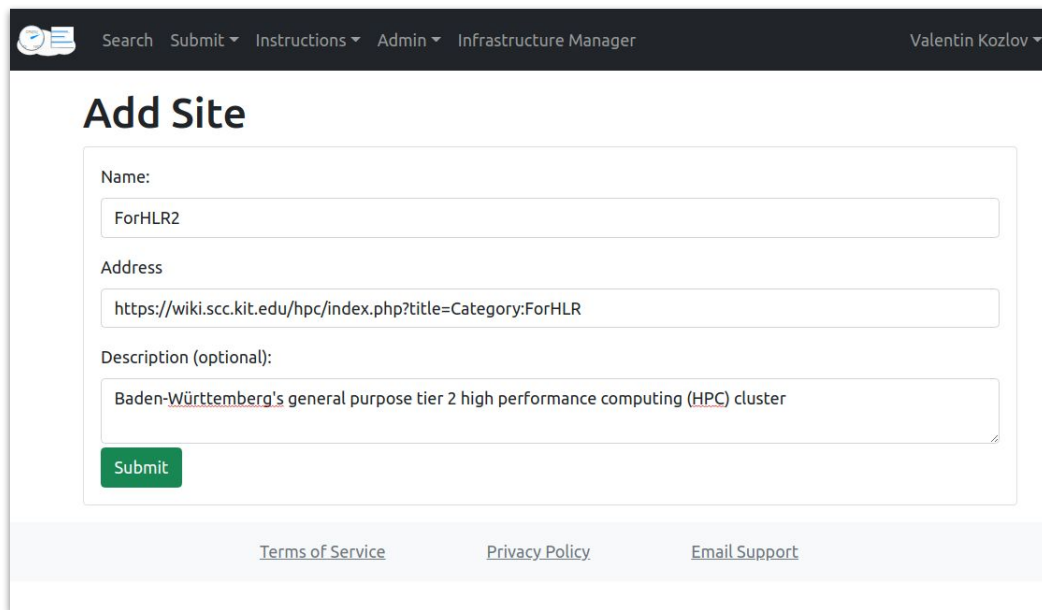
**US2:** store results of  
any benchmark of interest

**Contributors (⇒ authentication):**

Add a new resource:

- Name
- Address
- Description

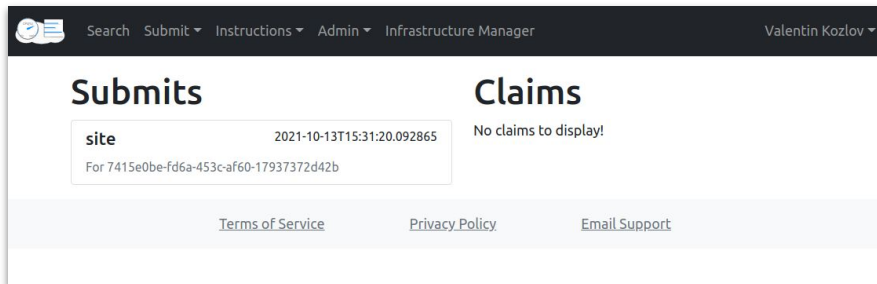
**N.B.:** Adding Site / Flavor is also  
possible during “Upload Result”

A screenshot of a web application interface for adding a new site. The interface has a dark header bar with navigation links: Search, Submit, Instructions, Admin, and Infrastructure Manager. The user's name, Valentin Kozlov, is displayed in the top right. The main content area is titled 'Add Site' and contains three input fields: 'Name' (with the value 'ForHLR2'), 'Address' (with the URL 'https://wiki.scc.kit.edu/hpc/index.php?title=Category:ForHLR'), and 'Description (optional):' (with the text 'Baden-Württemberg's general purpose tier 2 high performance computing (HPC) cluster'). A green 'Submit' button is located below the description field. At the bottom of the form, there are three links: 'Terms of Service', 'Privacy Policy', and 'Email Support'.

# EOSC-Perf: Supported Flows

## Admin functions:

### Review Submissions and Reports



Search Submit Instructions Admin Infrastructure Manager Valentin Kozlov

### Submits

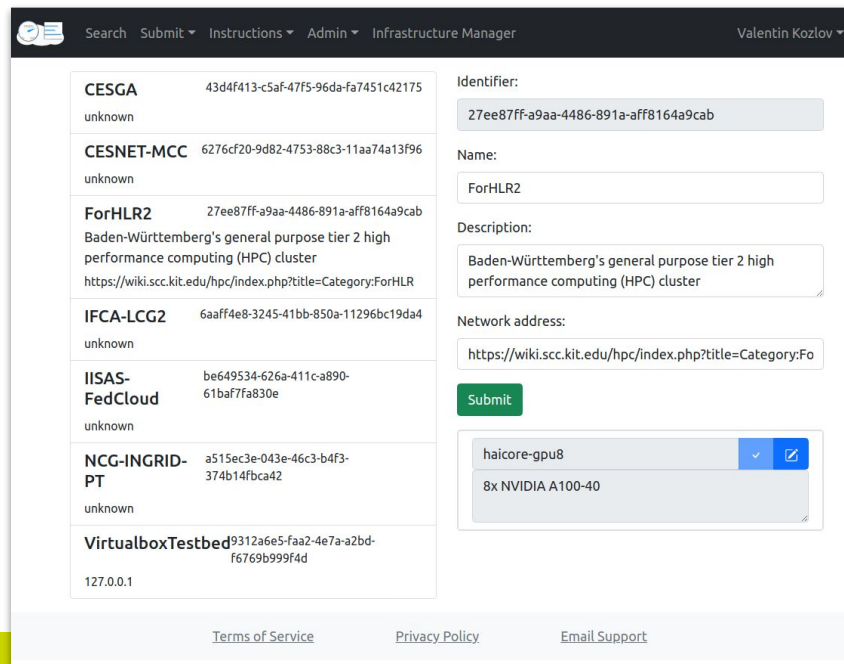
site	2021-10-13T15:31:20.092865
For 7415e0be-fd6a-453c-af60-17937372d42b	

### Claims

No claims to display!

[Terms of Service](#)
[Privacy Policy](#)
[Email Support](#)

## Edit sites with “Site Editor”



Search Submit Instructions Admin Infrastructure Manager Valentin Kozlov

<b>CESGA</b> unknown	43d4f413-c5af-47f5-96da-fa7451c42175	Identifier: 27ee87ff-a9aa-4486-891a-aff8164a9cab
<b>CESNET-MCC</b> unknown	6276cf20-9d82-4753-88c3-11aa74a13f96	Name: ForHLR2
<b>ForHLR2</b> Baden-Württemberg's general purpose tier 2 high performance computing (HPC) cluster https://wiki.scc.kit.edu/hpc/index.php?title=Category:ForHLR	27ee87ff-a9aa-4486-891a-aff8164a9cab	Description: Baden-Württemberg's general purpose tier 2 high performance computing (HPC) cluster
<b>IFCA-LCG2</b> unknown	6aaff4e8-3245-41bb-850a-11296bc19da4	Network address: https://wiki.scc.kit.edu/hpc/index.php?title=Category:Fo
<b>IISAS-FedCloud</b> unknown	be649534-626a-411c-a890-61ba77fa830e	Submit
<b>NCG-INGRID-PT</b> unknown	a515ec3e-043e-46c3-b4f3-374b14fbca42	haicore-gpu8 8x NVIDIA A100-40
<b>VirtualboxTestbed</b> 127.0.0.1	9312a6e5-faa2-4e7a-a2bd-f6769b999f4d	

[Terms of Service](#)
[Privacy Policy](#)
[Email Support](#)

## EOSC-Perf: more features

- Custom filters
- Expand table with benchmark-specific JSON fields
- Report results (authenticated users)
- Custom tags
- ...



# API

Note: This demo will use a non production endpoint for the demonstration:

<https://perf-stage.test.fedcloud.eu>

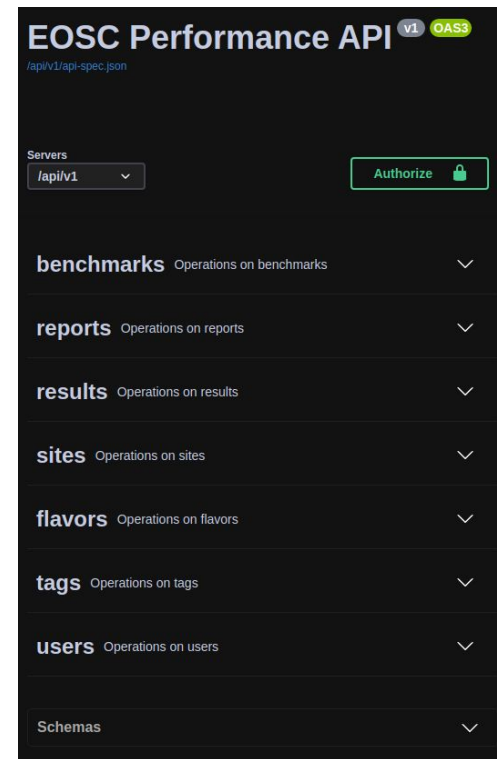
# API - Powered by Swagger and OpenAPI

Based on OpenAPI v3

- Possibility to provide multiple hosts.
- Supports Bearer/JWT authentication natively.

Friendly web user interface by Swagger  
accessible at:

<https://performance.services.fedcloud.eu/api/v1/>



# API - Designed to follow REST architecture

benchmarks Operations on benchmarks		
GET	/benchmarks	(Public) Filters and list benchmarks
POST	/benchmarks	(Users) Uploads a new benchmark
GET	/benchmarks:search	(Public) Filters and list benchmarks
GET	/benchmarks/{id}	(Public) Retrieves benchmark details
PUT	/benchmarks/{id}	(Admins) Implements JSON Put for benchmarks
DELETE	/benchmarks/{id}	(Admins) Deletes an existing benchmark
POST	/benchmarks/{id}:approve	(Admins) Approves a benchmark to include it on default list methods
POST	/benchmarks/{id}:reject	(Admins) Rejects a benchmark to safe delete it.

List items

Create items

Get item Admins

Edit item

Delete item

## API - Automate results and more

### DEMO

Powered by bash and Jupyterlab.

How to automatically submit a new result.



Set of demo notebooks:

<https://github.com/EOSC-synergy/eosc-perf-backend/blob/main/examples/>

# API - To contribute, register

Please, read our privacy policy at:

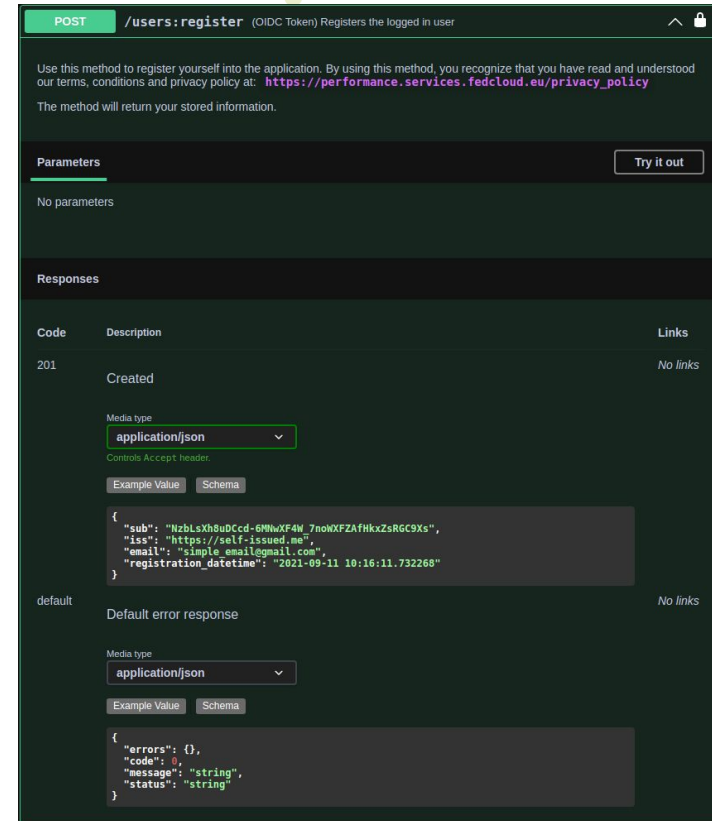
[https://performance.services.fedcloud.eu/privacy\\_policy](https://performance.services.fedcloud.eu/privacy_policy)

Registration is very simple:

```
$ curl -X 'POST' \
  'https://performance.services.fedcloud.eu/api/v1/users:register' \
  -H 'accept: application/json' \
  -H "Authorization: Bearer $access_token"
```

Get access to:

- **Create;** Benchmarks, Results, Sites, Flavors, Tags
- **Report;** Results



**POST** /users:register (OIDC Token) Registers the logged in user

Use this method to register yourself into the application. By using this method, you recognize that you have read and understood our terms, conditions and privacy policy at: [https://performance.services.fedcloud.eu/privacy\\_policy](https://performance.services.fedcloud.eu/privacy_policy)

The method will return your stored information.

**Parameters** Try it out

No parameters

**Responses**

Code	Description	Links
201	Created	No links

Media type: **application/json** Controls Accept header.

Example Value Schema

```
{
  "sub": "NzblxVh8uDCcd:6MhwVFAN_7noWKFZAfhkxZsRGc9Xs",
  "iss": "https://self-issued.me",
  "email": "simple_email@gmail.com",
  "registration_datetime": "2021-09-11 10:16:11.732268"
}
```

default No links

Default error response

Media type: **application/json**

Example Value Schema

```
{
  "errors": {},
  "code": 0,
  "message": "string",
  "status": "string"
}
```

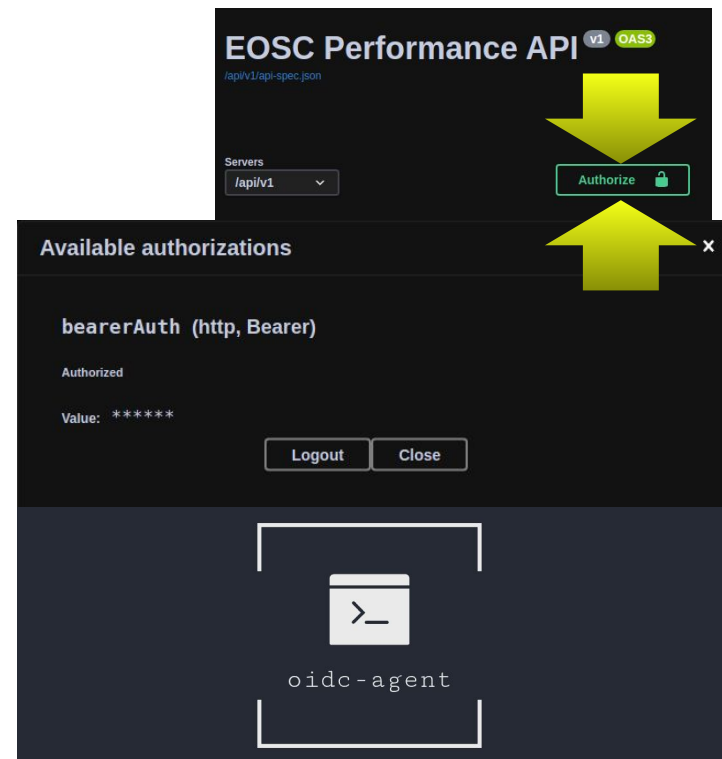
# API - How to provide identification

To get your access token is just as simple as:

```
$ access_token=$(oidc-token egi-prod)
```

Full documentation about 'oidc-agent' can be found at:

<https://indigo-dc.gitbooks.io/oidc-agent/>



# Summary and Outlook

**EOSC-Performance** is the place to **compare** the **performance** of various EOSC sites

**Core features** available include:



- **Search** for benchmarks and results
- Result **filtering, comparison, and visualisation**
- **Adding** new results, benchmarks, sites
- **Admin** features
- **Full-featured API** to communicate with the platform

Any **suggestions?** Looking forward to [your feedback!](#)

## EOSC-Perf public links:

Detailed documentation: [perf.readthedocs.io](https://perf.readthedocs.io)

Git organisation/repos: [github.com/EOSC-Synergy/eosc-perf](https://github.com/EOSC-Synergy/eosc-perf)

EOSC-Marketplace: <https://marketplace.eosc-portal.eu/services/eosc-performance>

Or directly: <https://performance.services.fedcloud.eu/>

Contact [perf-support@lists.kit.edu](mailto:perf-support@lists.kit.edu)

**Stay tuned!**



EGI'22 Abstract