EUROPEAN WEATHER CLOUD

EWC Community and Platform Services

EWC User Workshop

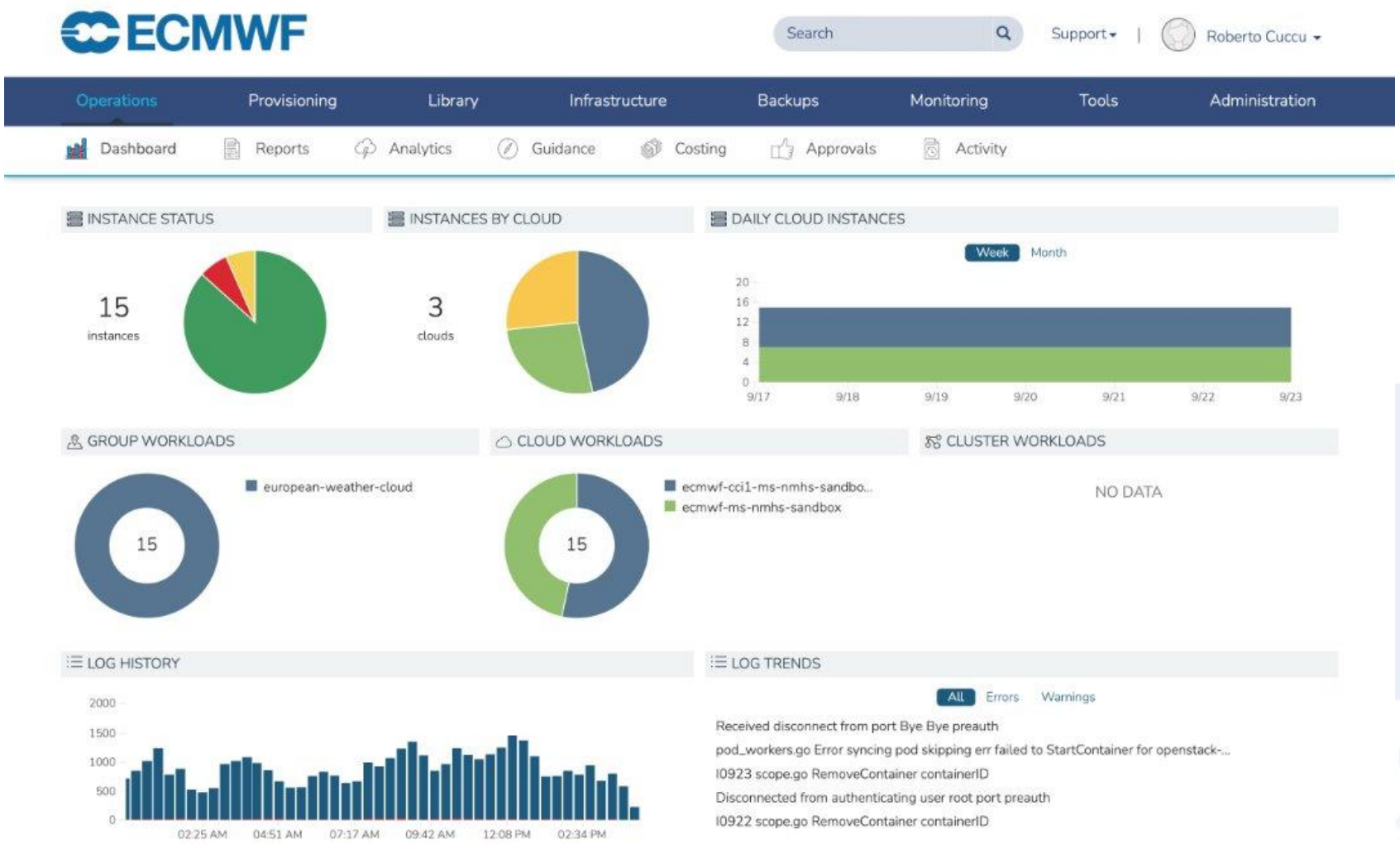
13th November 2024

Francesco Murdaca (EUMETSAT), on behalf of the EWC Team

Outline

- Morpheus – Cloud Management Platform
- Knowledge Base and Support Portal
- EWC Website
- Accounting Tool
- Discussion Platform
- Feature Roadmap and Software Marketplace

Morpheus – Cloud Management Platform



Morpheus is a cloud management platform able to control multiple cloud infrastructure.

Each project with access to the European Weather Cloud is called a **Tenant**.

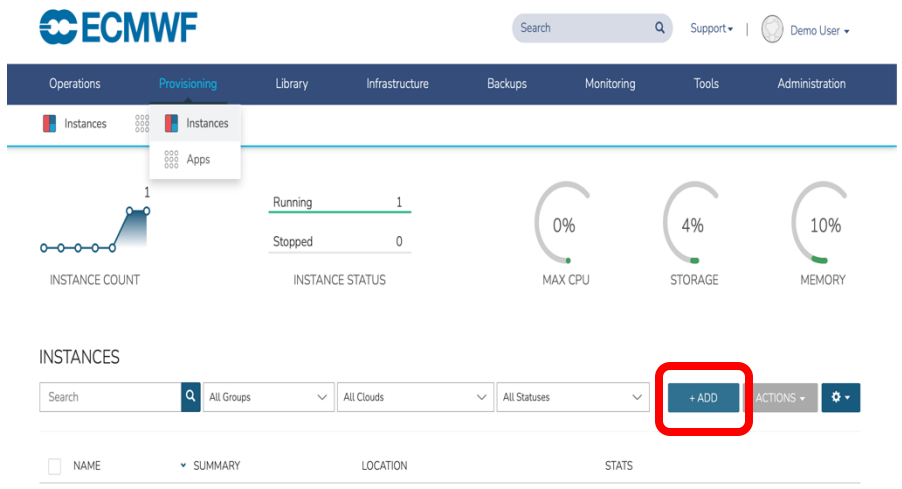
Each partner in the EWC makes available their infrastructure to the users which is seen as a **Cloud** in Morpheus.

One **Tenant** can have access to multiple **clouds** (e.g. one on EUMETSAT side and one on ECMWF side), managed by the same **Morpheus**

Virtual Machine Provisioning

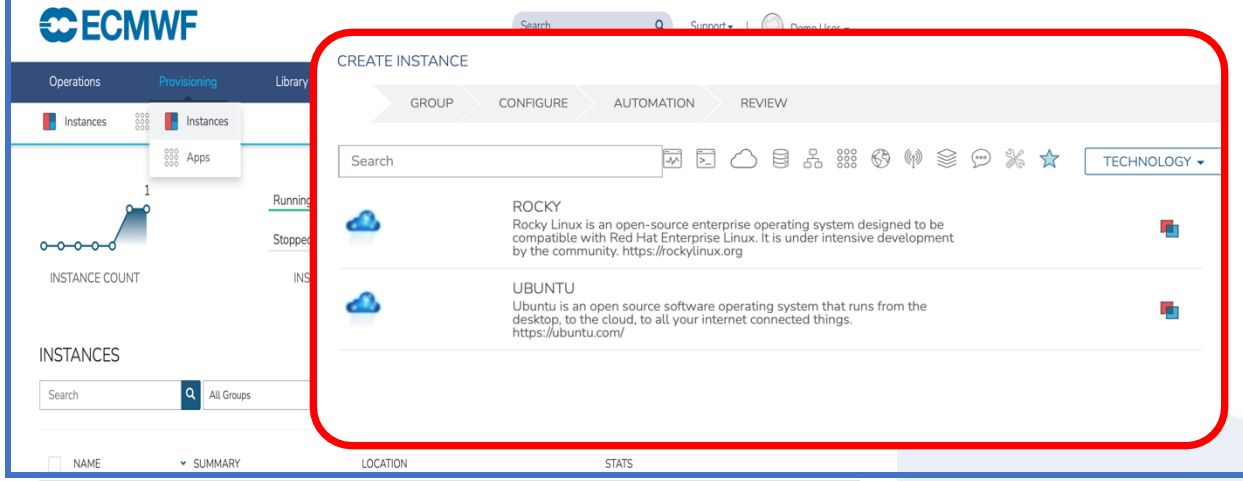
Provisioning-> Instances -> +ADD

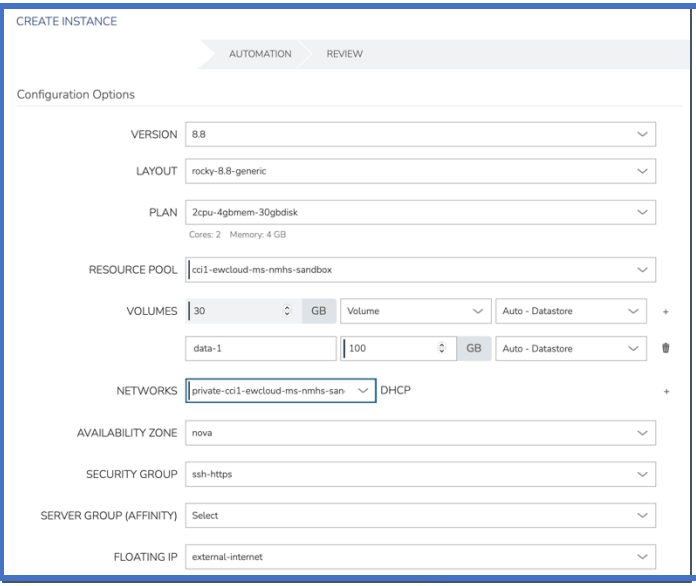
1



2

Instance Type selection



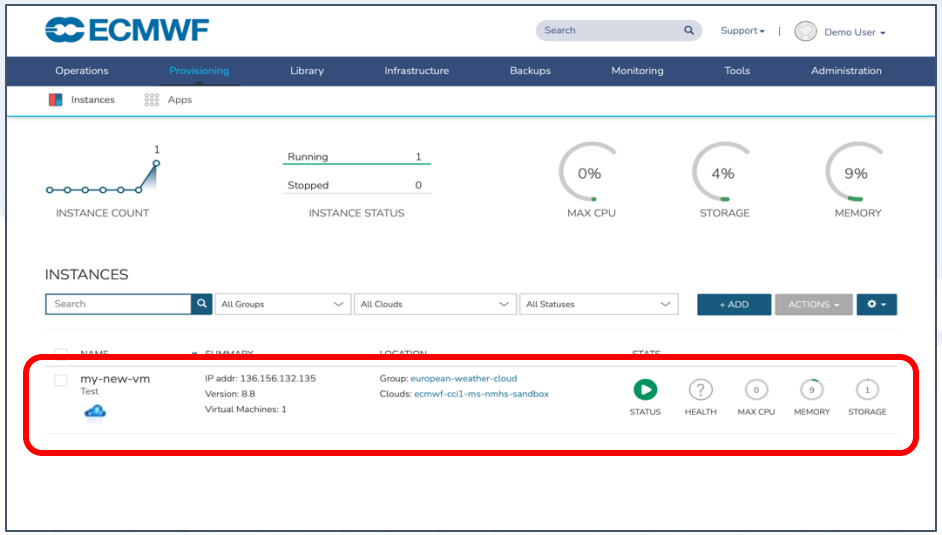


Instance configuration options:

- Layout
- Plan
- Volumes
- Networks (public and/or private)
- Security Group
- Floating IP

3

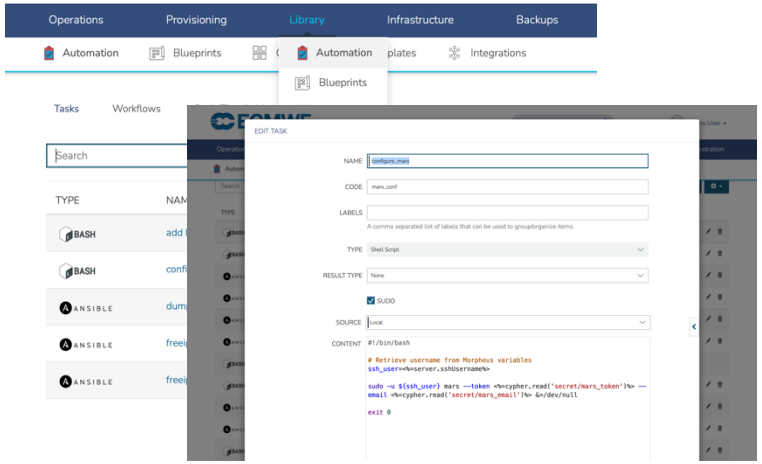
4



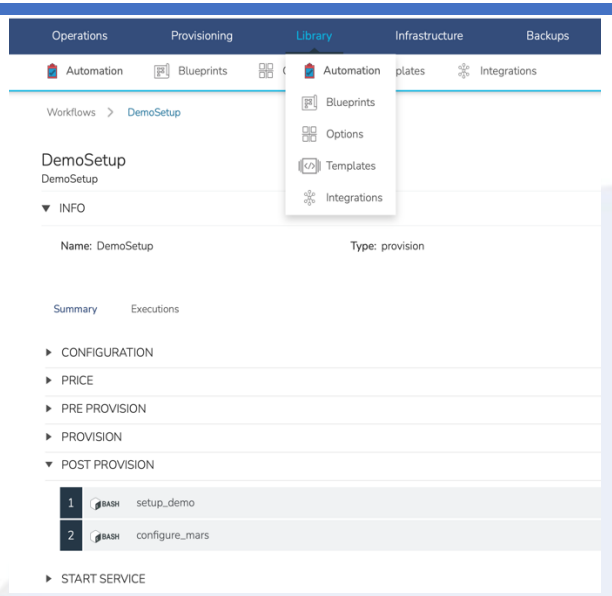
Tasks and Workflows and Integrations

Library-> Automation-> Workflows and Tasks

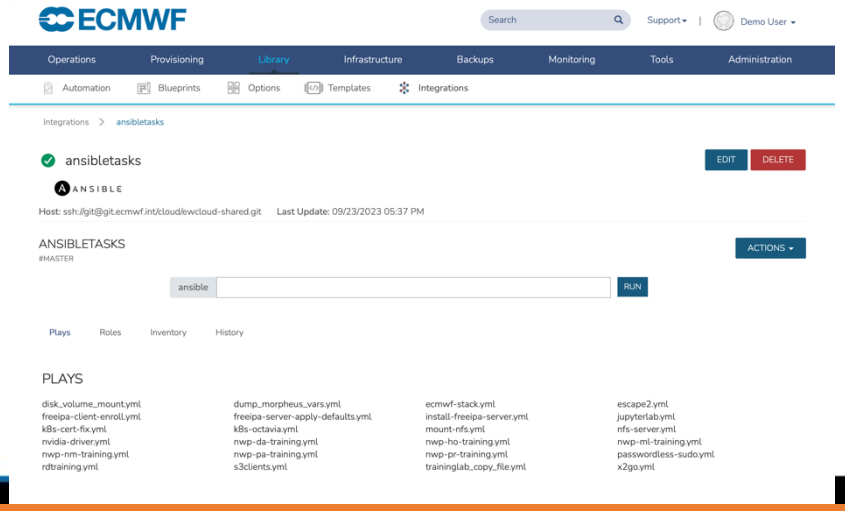
Tasks



Workflows



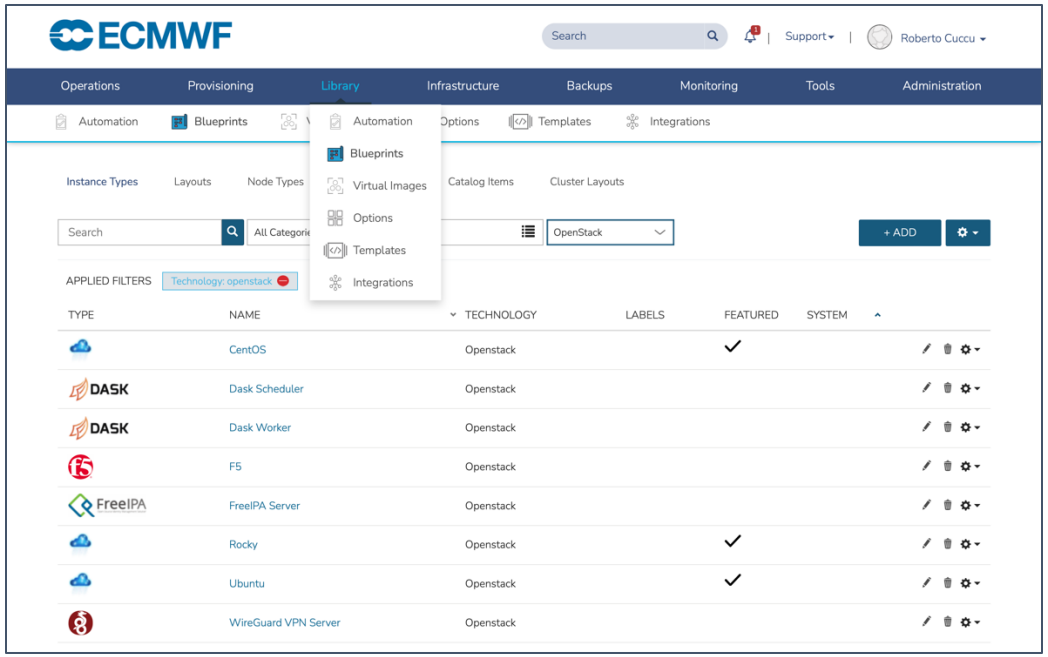
Library-> Integrations

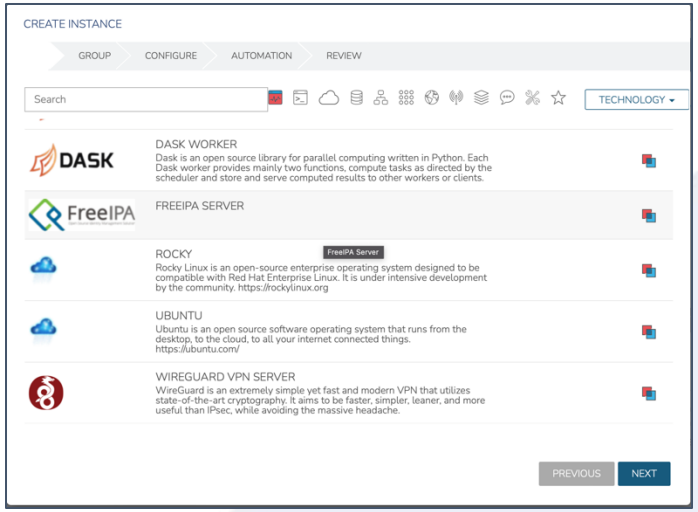


Automation Integrations can be added to enable the use of configuration management tools, e.g. Ansible

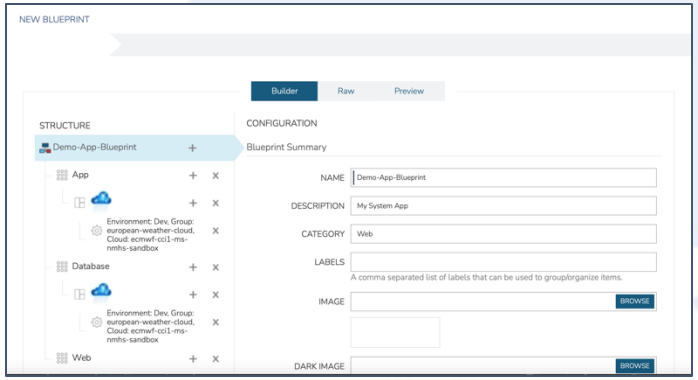
Blueprints, Instance Types

Library-> Blueprint





Instance Types and Layouts

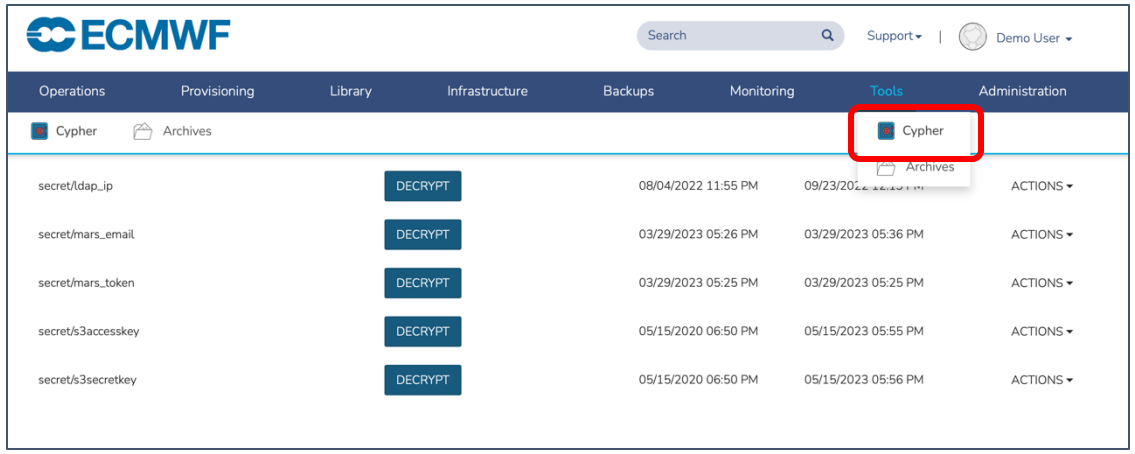


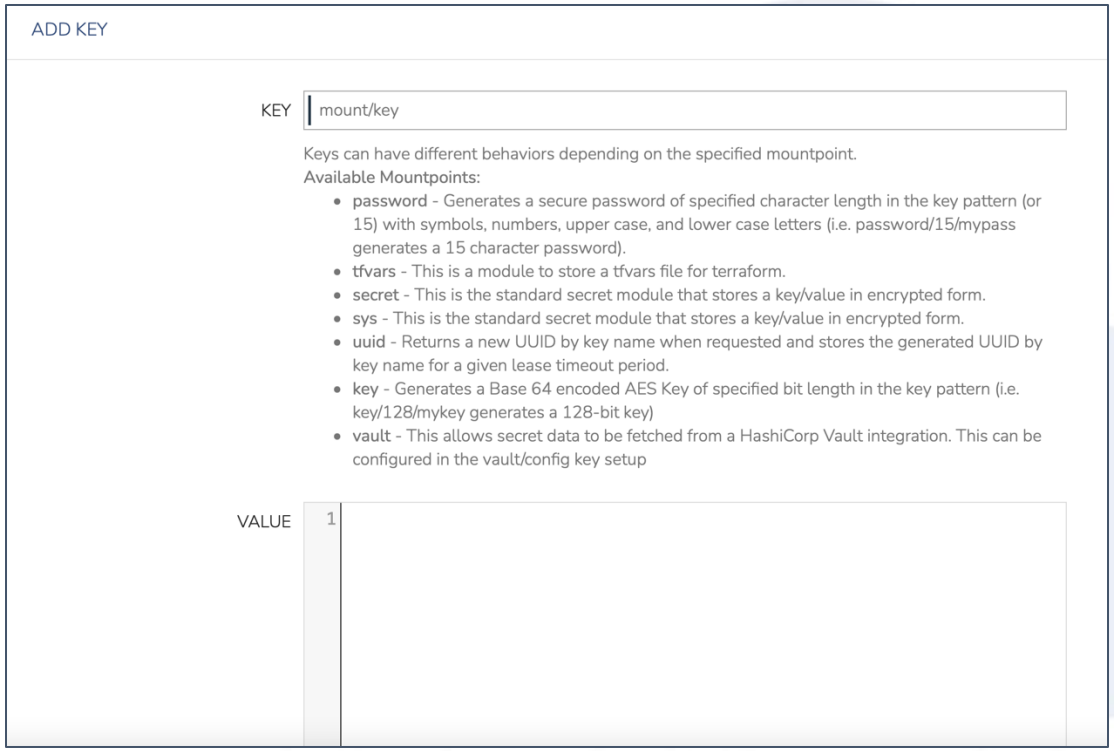
App Blueprints

Blueprints feature together with automation tools allow the setup of reproducible application environment for their flexible on-demand deployment.

Cypher – Secret Management

Tools -> Cypher





Morpheus Cypher is a built-in functionality to store sensitive information, e.g. to store secrets like passwords or API keys.

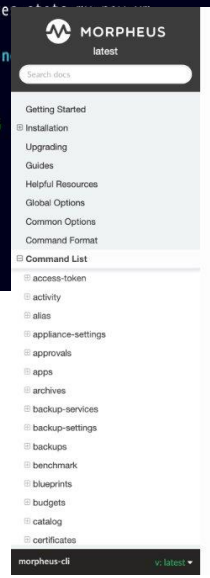
The feature is integrated with the Automation tools and it is possible to access stored secrets programmatically e.g. via shell scripts or Ansible tasks.

Stored secrets can be revoked either manually or automatically after a timeout.

Automation – Morpheus CLI and APIs

CLI

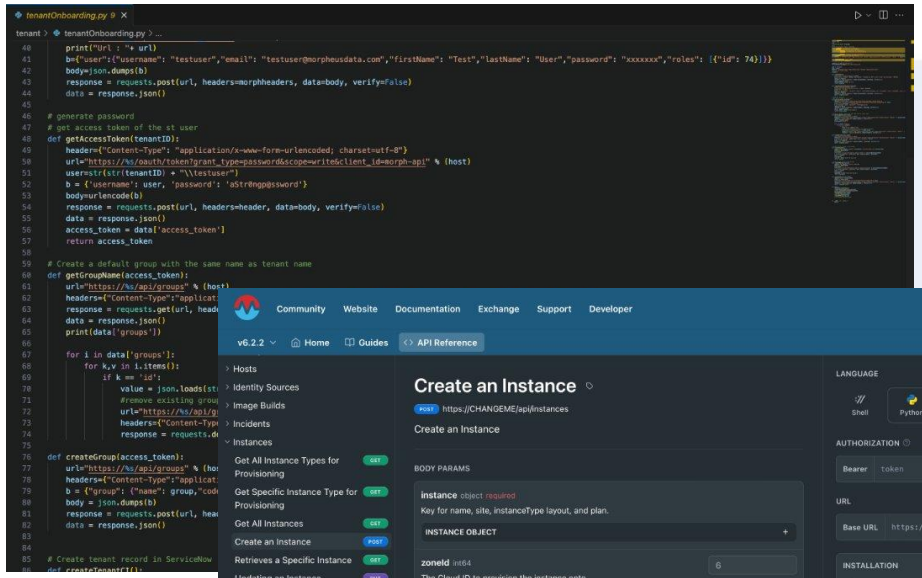
```
~$ morpheus instances list

Morpheus Instances
==================

ID   | NAME      | GROUP                | CLOUD                        | TYPE  | VERSION | ENVIRONMENT
-----|-----------|----------------------|------------------------------|-------|---------|------------
5382 | my-new-vm | european-weather-cloud | ecmwf-cci1-ms-nmhs-sandbox | Rocky | 8.8     | qa

Viewing 1-1 of 1

~$
~$ morpheus instance

Instance Stats: my-n
====================

   Status: RUNNING
    Nodes: 1
      CPU: []
   Memory: [|||||
  Storage: []

~$
~$
```

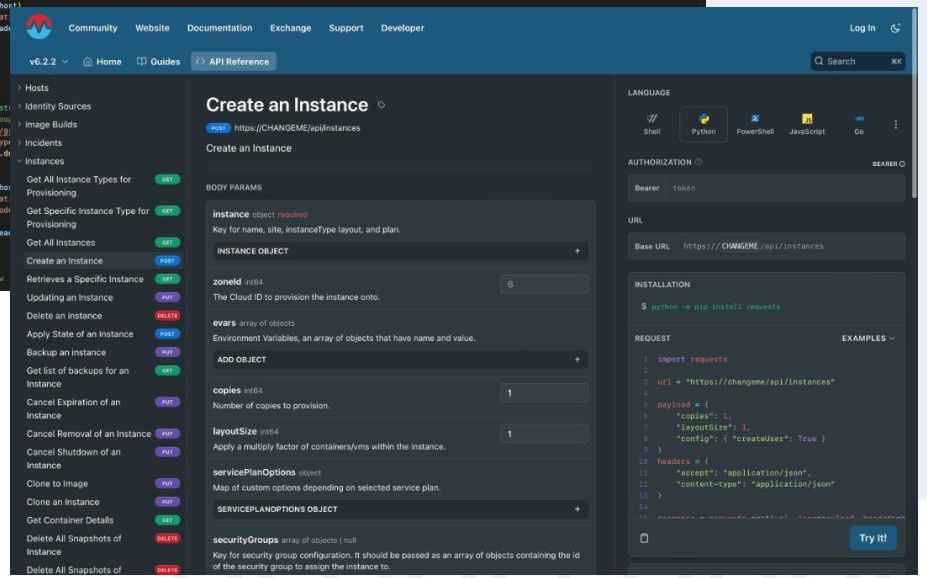


instances list

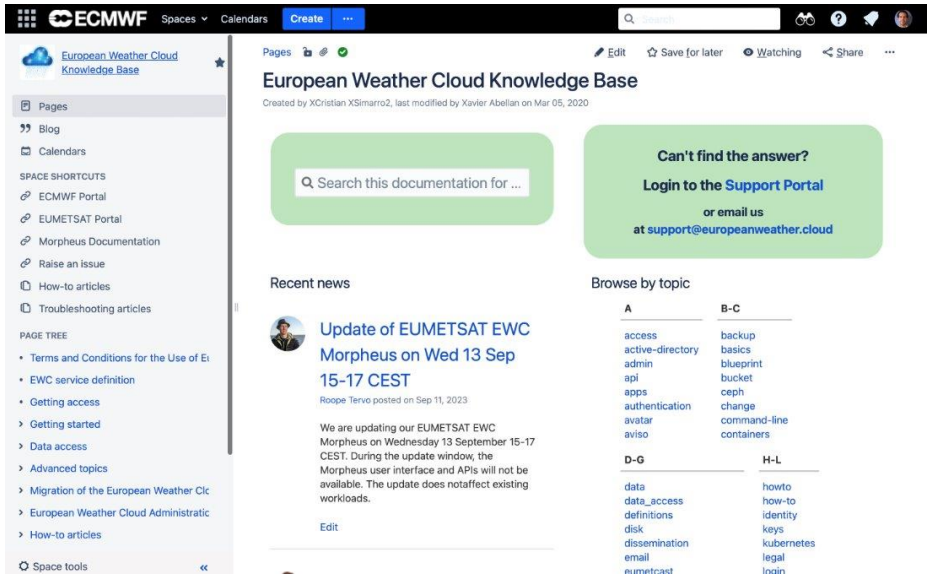
```
Usage: morpheus instances list
    -g, --group GROUP                Group Name or ID
    -c, --cloud CLOUD                Cloud Name or ID
        --host HOST                  Host Name or ID
        --owner USER                 Owner Username or ID
        --agent                      Show only Instances with the agent installed
        --noagent                    Show only Instances with No agent
        --status STATUS              Filter by status i.e. provisioning,running,starting,stopping
        --type CODE                  Filter by Instance Type code
        --environment CODE           Filter by Environment code(s)
        --pending-removal            Include instances pending removal.
        --pending-removal-only       Only instances pending removal.
        --plan NAME                  Filter by Plan name(s)
        --plan-id ID                 Filter by Plan id(s)
        --plan-code CODE             Filter by Plan code(s)
    -l, --labels LABEL               Filter by labels, can match any of the values
        --all-labels LABEL           Filter by labels, must match all of the values
        --tags Name=Value            Filter by tags (metadata name value pairs).
        --stats                      Display values for memory and storage usage used / max values.
    -a, --details                    Display all details: plan, stats, etc
    -m, --max MAX                    Max Results (use -1 for all results)
    -o, --offset OFFSET              Offset Results
    -s, --search PHRASE              Search Phrase
    -S, --sort ORDER                 Sort Order. DIRECTION may be included as "ORDER [asc|desc]".
    -D, --desc                       Descending Sort Direction.
        --reverse                    Reverse order of results. This invert is done by the client, not necessarily the entire data
    -Q, --query PARAMS               Query parameters. PARAMS format is 'foo=bar&category=web'
    -j, --json                       JSON Output
    -y, --yaml                       YAML Output
        --csv                        CSV Output
        --quotes                     Wrap CSV values with ". Default: false
        --no-header                  Exclude header for CSV Output.
    -f, --fields x,y,z               Filter Output to a limited set of fields. Default is all fields for json,csv,yaml.
        --all-fields                 Show all fields present in the data.
        --wrap                       Wrap table columns instead hiding them when terminal is not wide enough.
        --select x,y,z               Filter Output to just print the value(s) of specific fields.
        --delimiter [CHAR]           Delimiter for output values. Default: ',', use with --select and --csv
        --newline [CHAR]             Delimiter for output rows. Default: '\n', use with --select and --csv
    -q, --quiet                      No Output, do not print to stdout
    -d, --dry-run                    Dry Run, print the API request instead of executing it.
        --curl                       Curl, print the API request as a curl command instead of executing it.
        --scrub                      Mask secrets in output, such as the Authorization header. For use with --curl and --dry-run.
    -r, --remote REMOTE              Remote name. The current remote is used by default.
        --remote-url URL             Remote url. This allows adhoc requests instead of using a configured remote.
    -T, --token TOKEN                Access token for authentication with --remote. Saved credentials are used by default.
    -u, --username USERNAME          Username for authentication.
    -P, --password PASSWORD          Password for authentication.
    -I, --insecure                   Allow insecure HTTPS communication.  i.e. bad SSL certificate.
    -C, --nocolor                    Disable ANSI coloring
    -B, --benchmark                  Print benchmark time and exit/error after the command is finished.
    -V, --debug                      Print extra output for debugging.
```

REST APIs





Knowledge Base



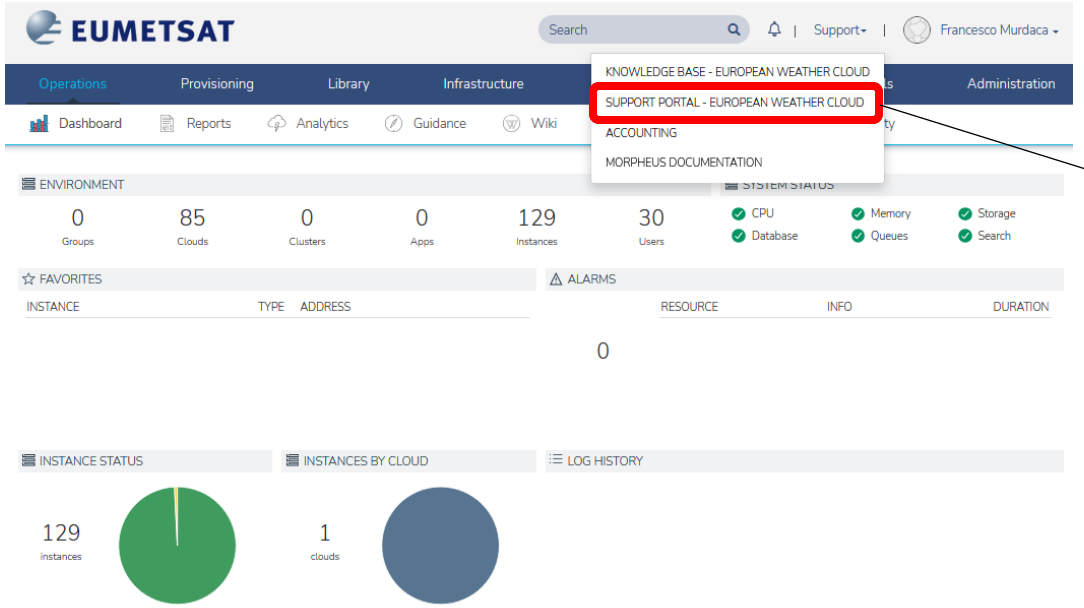
https://confluence.ecmwf.int/x/6J83Cg

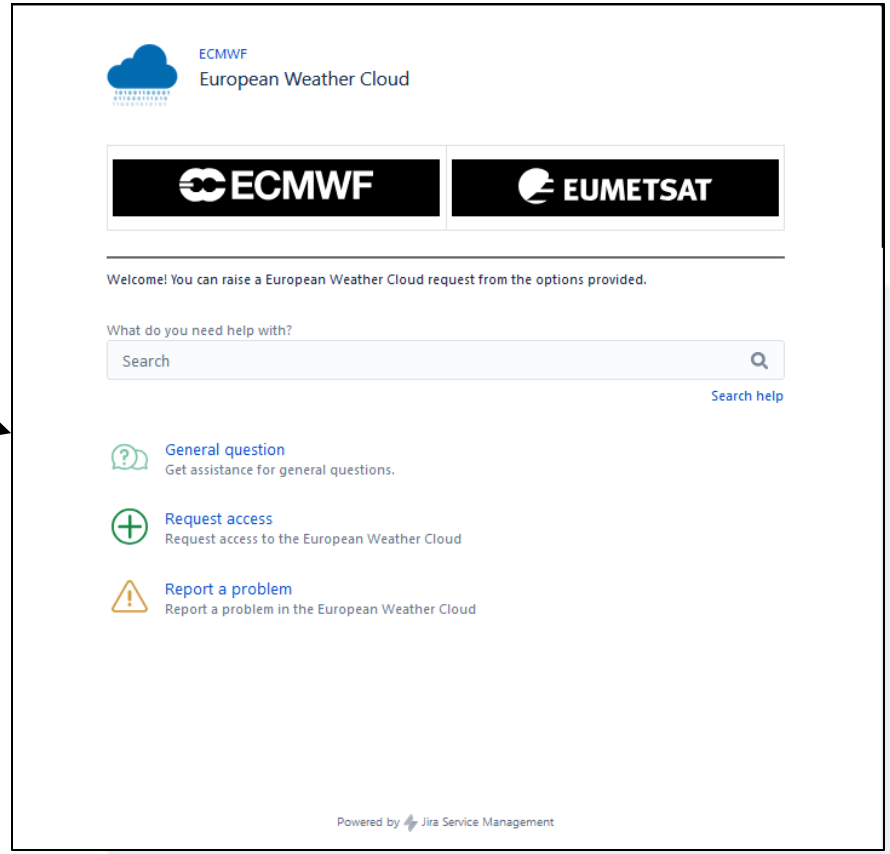
KB Content

- Terms and Conditions
- Getting access
- Getting started
- Compute
- Data access
- Storage
- How-to articles
- Troubleshooting articles
- Discussion platform
- Training and Templates
- EWC Templates

Blog Posts

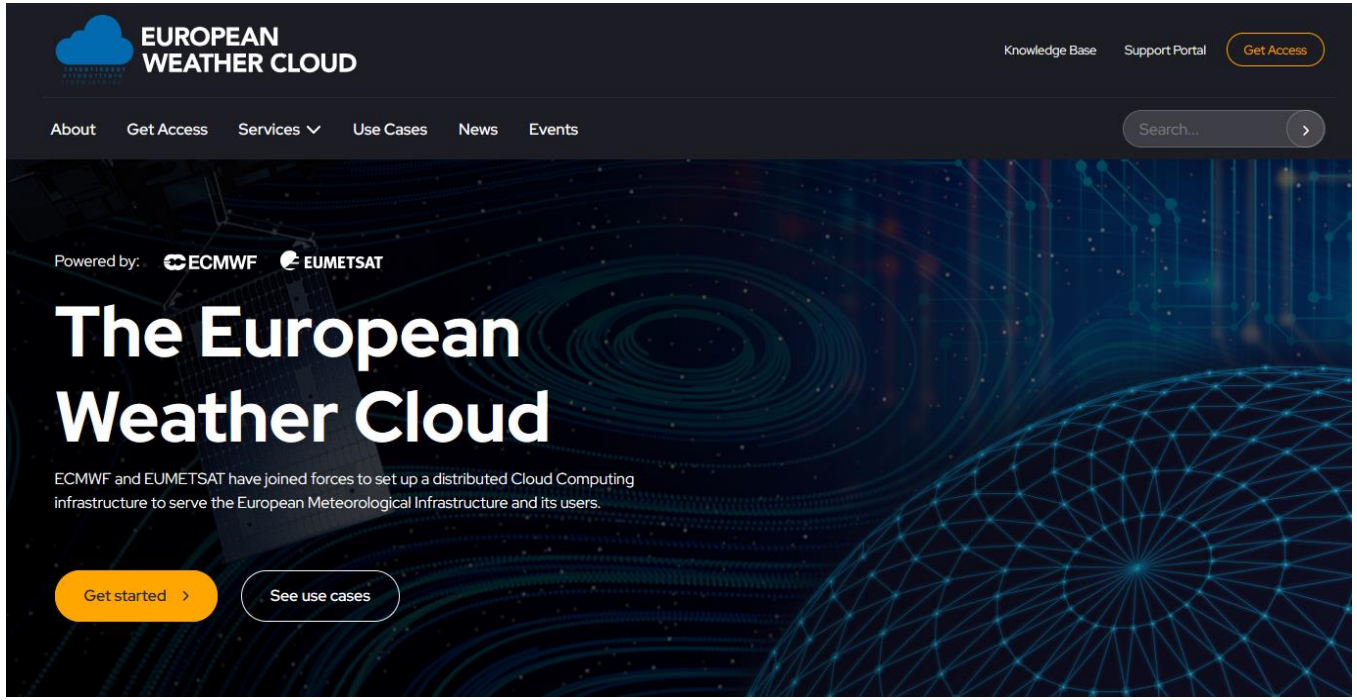
Support Portal





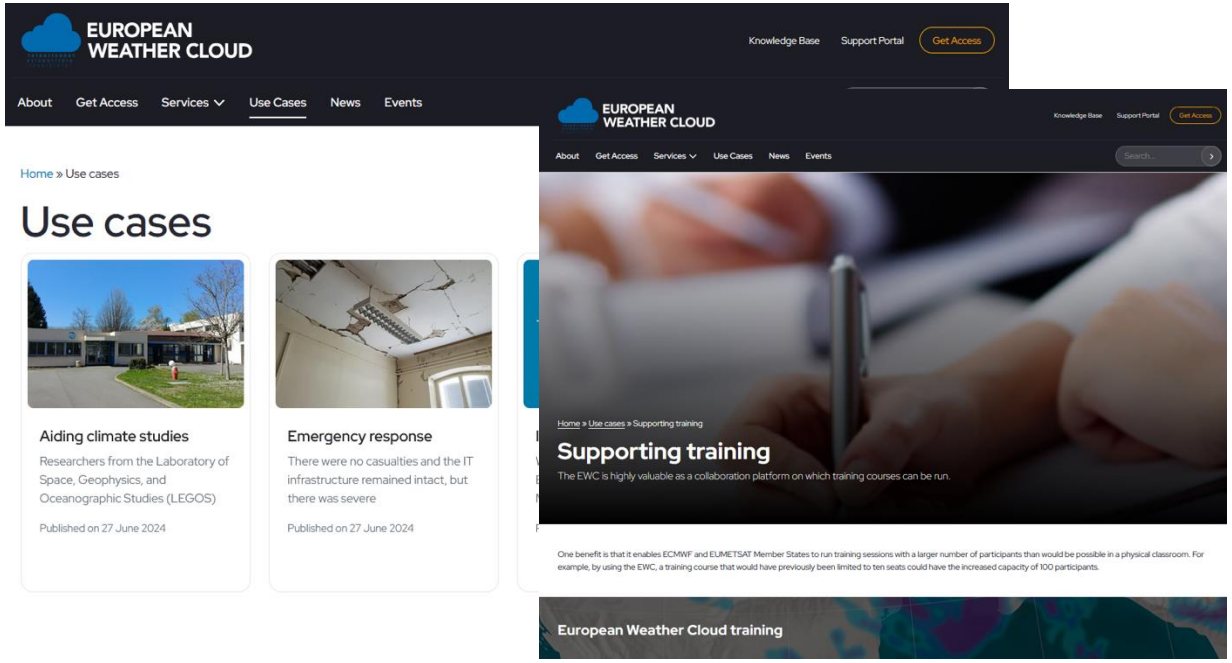
https://support.europeanweather.cloud/

EWC Website



- **New design**
- **Better Description of the services**
- **Use Cases**
- **News and Events**
- **Service status dashboard *COMING SOON***

EWC Website – Contribute with your Use Case



You can **publish your EWC use case on the EWC website**.

Inputs needed:

- Title
- Organizations involved
- Description about the use case
- Picture (if any)

Contact us by email, user portal or in rocketchat if you would like to contribute with your use case!

EWC Accounting

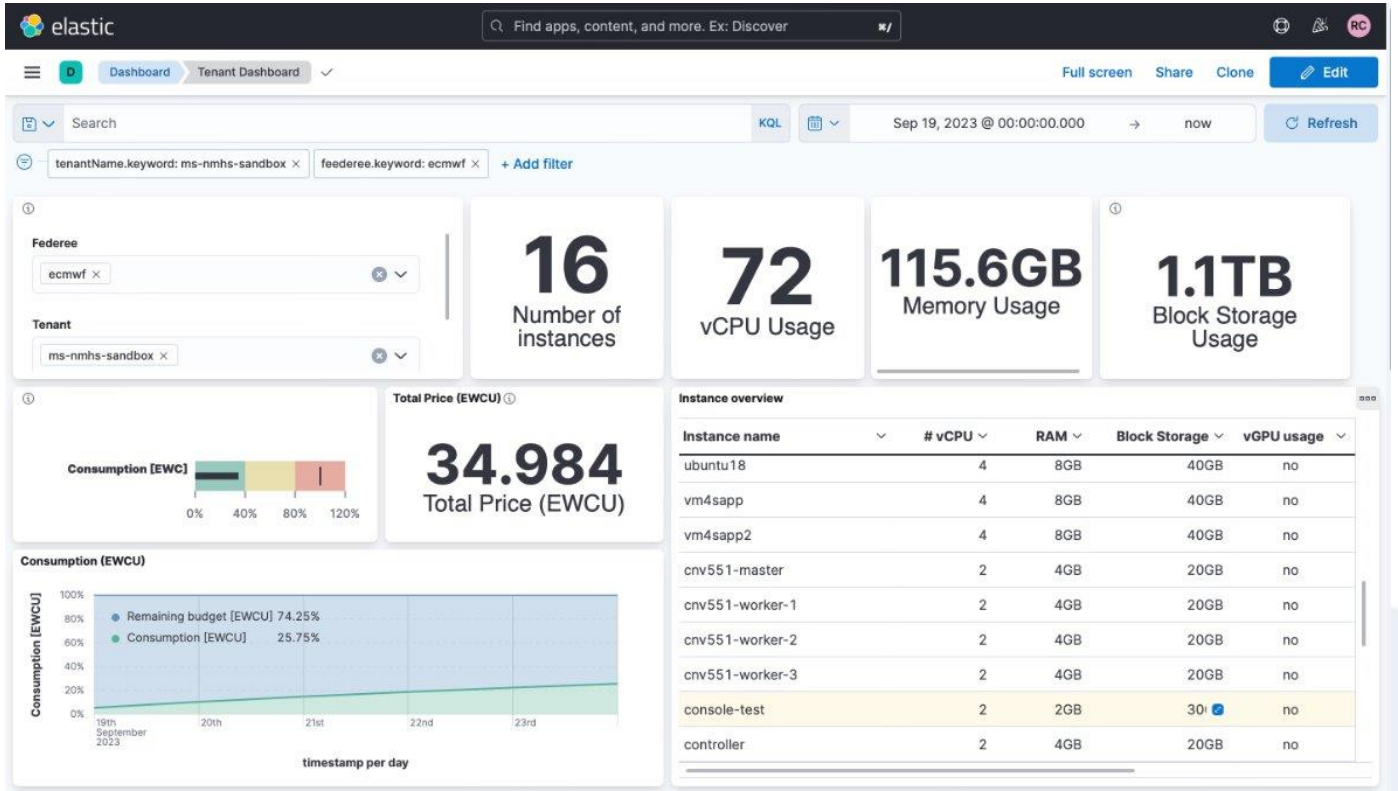
The **Accounting service** provides a cross-cloud overview of resource usage of the tenancies.

A web GUI provides metrics, time series, graphs, and dashboards displaying the accounting information.

Accounted resources include:

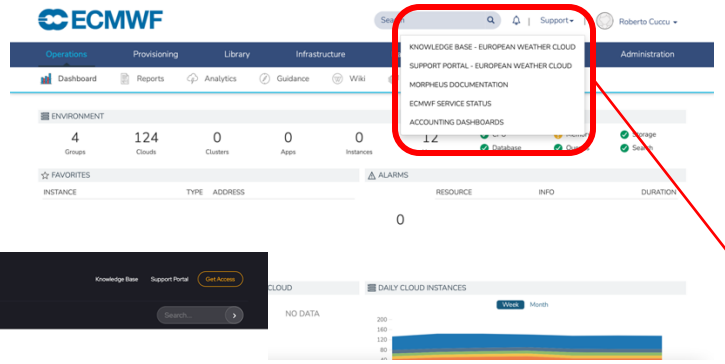
- **Virtual Machines** (CPU, RAM)
- **Storage**
  - Block Storage (volumes attached to the VM)
  - Object Storage (S3 buckets)
  - Shared File System (extra volume shared with multiple VMs)
- **GPU usage** (set of VM configuration including GPU card)

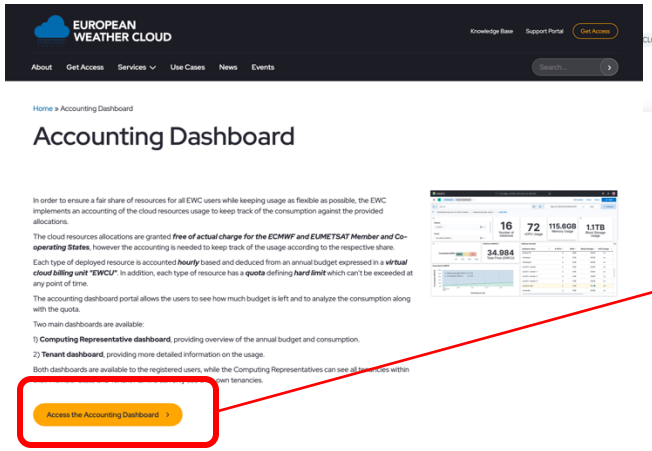
A **Cloud Billing Unit (CBU)** is adopted as virtual currency for the accounting of the consumed cloud resources.

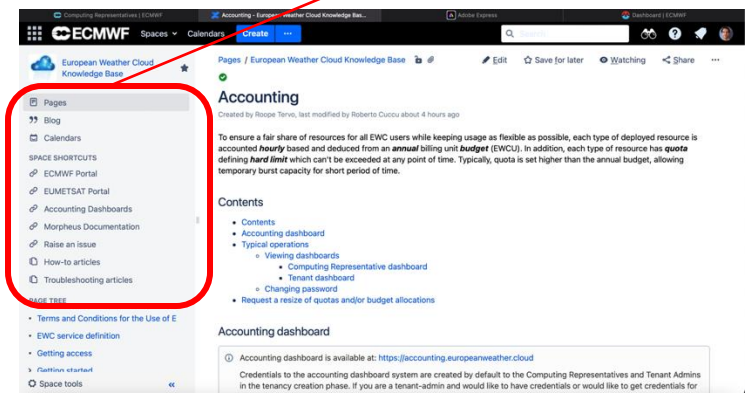


Dashboard	Description
Computing Representative Dashboard	providing overview of the annual budget and consumption
Tenant Dashboard	providing more detailed information on the usage

EWC Accounting – How to access it?





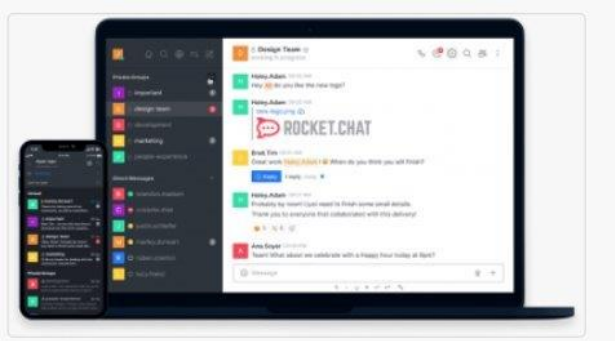


https://accounting.europeanweather.cloud

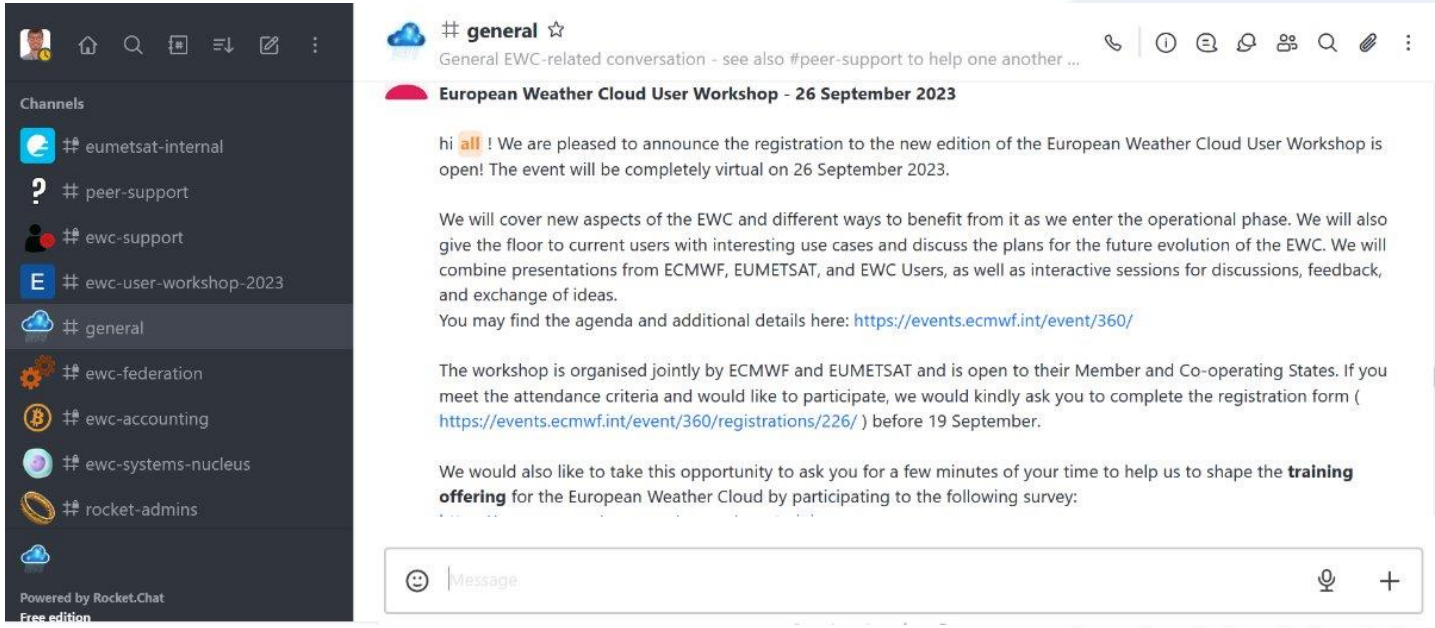
Accounting Dashboards link can be found on the EWC website, in the Morpheus portal support menu or in the EWC Knowledge Base side bar.

EWC Discussion Platform

Installation
- Web based
- Desktop App
- Mobile



Rocketchat
Customizable communication platform hosted on EWC
Secure and compliant to standards (e.g. E2E encryption, GDPR)
Open Source
Unlimited history of discussion
Not an official support tool!! -> JIRA ticket or support email
It's a pilot -> review process in place



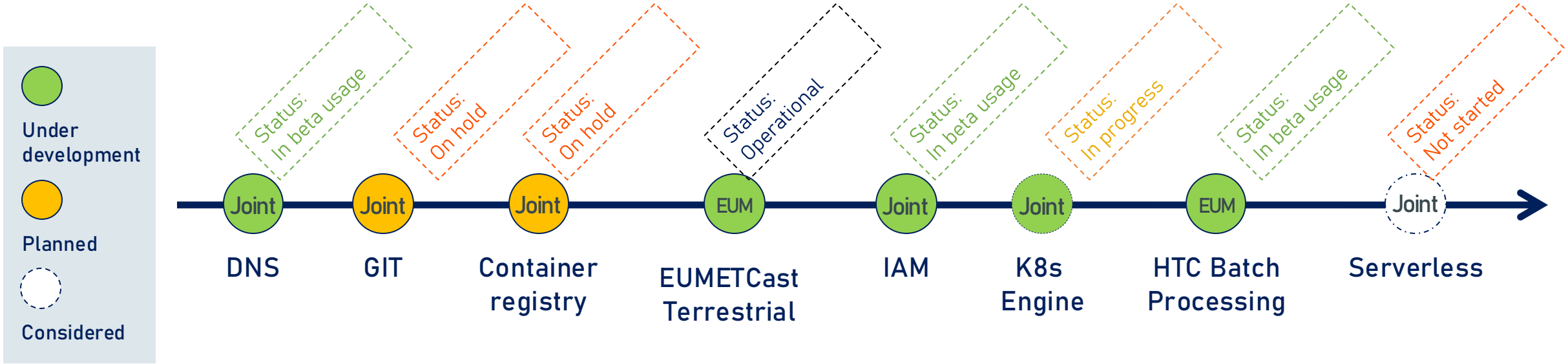
EWC Discussion Platform

How do I join?
All Member States developers can self-register (based on email domain): https://chat.europeanweather.cloud
External users can't self-register but can be invited. They need to be accepted by admins.

What channels are featured?	What Can I do here?
#general	• Learn about any news, updates from EWC community • Share interesting article/topics for the community • Discuss tools/solution you are using that could benefit the community
#peer-support	**#community-support channel**, if you are encountering issues with any tools or need recommendations on some technology or solution, you can ask the community
#random	**#off-topic channel** to share pictures, meme, anything to chill with the community

EWC Feature Roadmap

- Currently, the service mainly IaaS meaning that users can deploy VMs and additional infrastructure resources
- We are updating the EWC to contain more centrally managed platform and software services
- All services are first released as beta versions and later either operationalized or terminated based on their usefulness and feasibility
- The roadmap is open for updates and reprioritization, based on the user needs



EWC Marketplace

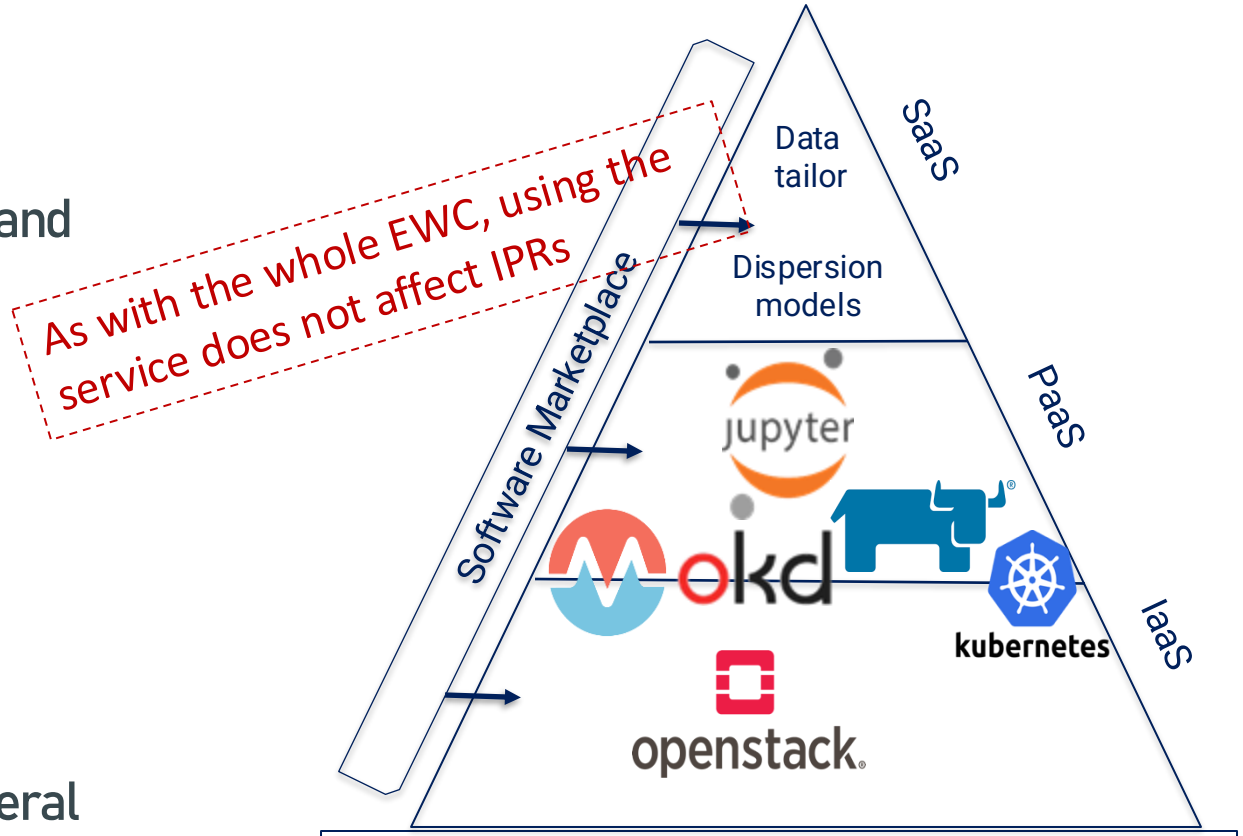
EWC marketplace aims to enable EWC tenants to share and reuse their assets (including e.g. software, ML models, infrastructure as a code, and examples) with each other

Contributions and application support based on the community contributions:

EWC-specific support for creating and maintaining assets to be provided

Envisaged to contain one primary GIT repository and several auxiliary channels to distribute the content:

GIT Repositories (hosted and others) / Container registries / VM images;



Examples of possible marketplace offering:

- Web Map Service on ECMWF and EUMETSAT data
- PyTroll satellite image production
- Nowcasting model templates
- Notebook environment on ML data analytics on archive or forecast data

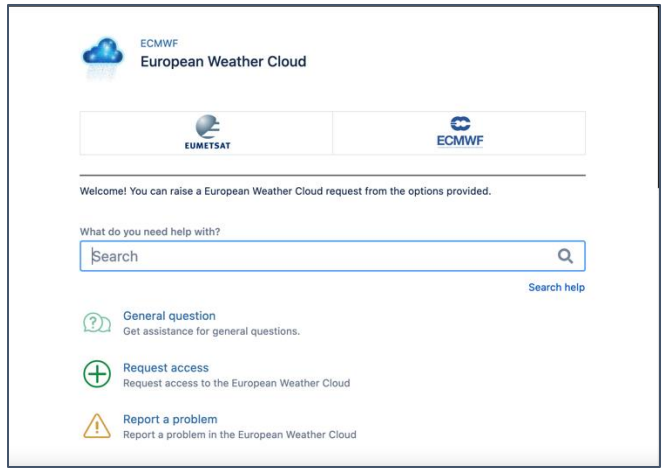
Contacts and references

https://europeanweather.cloud

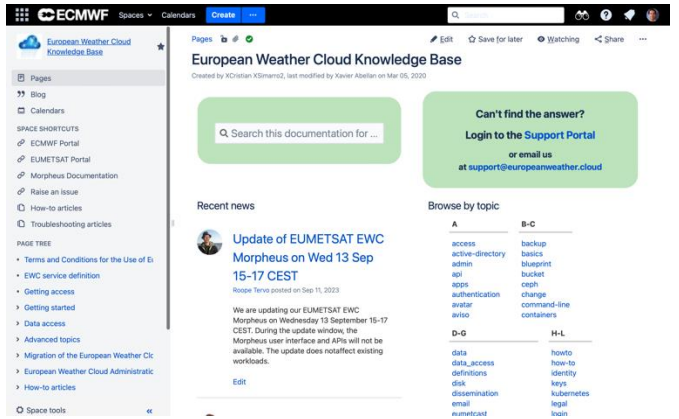
https://chat.europeanweather.cloud

https://support.europeanweather.cloud

Support Portal



Knowledge Base



Thank you for your attention!