VESPA-Cloud DACHS on Docker

Baptiste Cecconi
Pierre Le Sidaner
Philippe Hamy

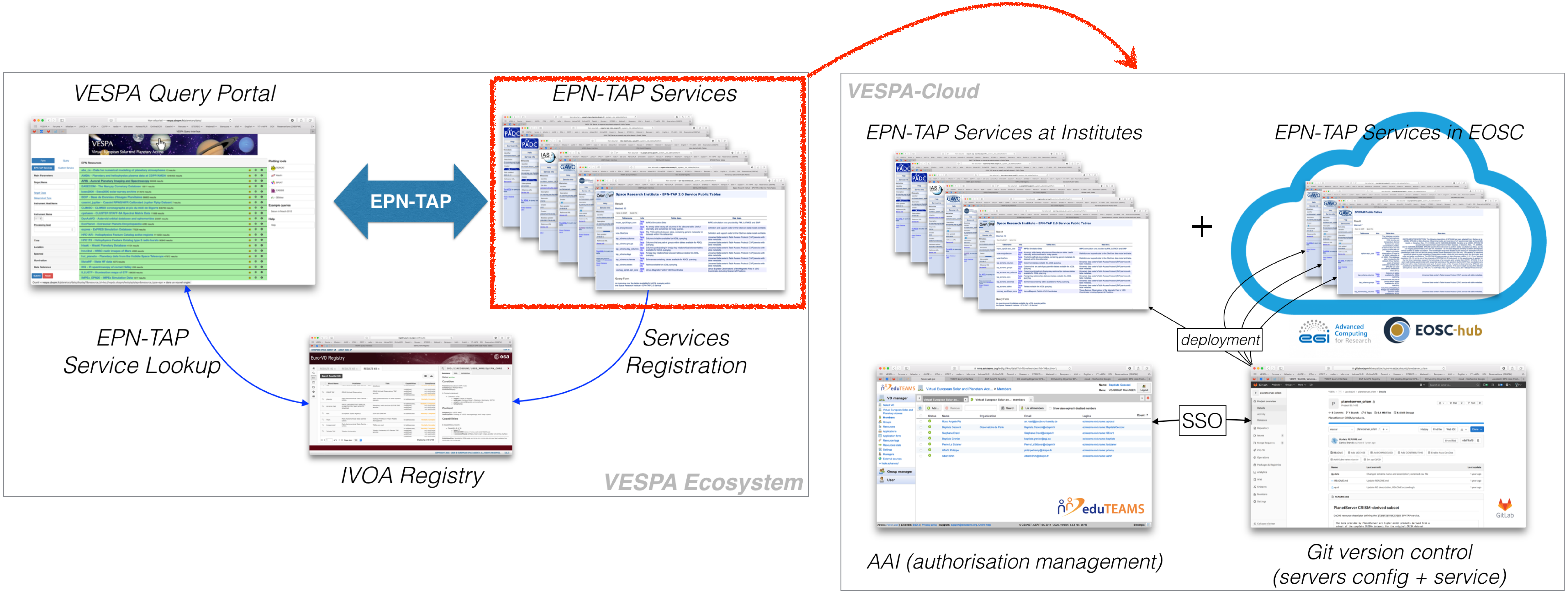
IVOA May 2021 - Apps session

The Europlanet-2024 Research Infrastructure project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 871149. This work used the EGI Infrastructure with the dedicated support of IN2P3-IRES and CESNET-MCC. This work used the eduTEAMS service provided by GÉANT.

VESPA-Cloud

Virtual European Solar and Planetary Access

A distributed Virtual Observatory for Solar System Sciences



VESPA Query Portal

EPN-TAP

EPN-TAP Services

EPN-TAP Service Lookup

Services Registration

IVOA Registry

VESPA Ecosystem

VESPA-Cloud

EPN-TAP Services at Institutes

+

EPN-TAP Services in EOSC

deployment

SSO

AAI (authorisation management)

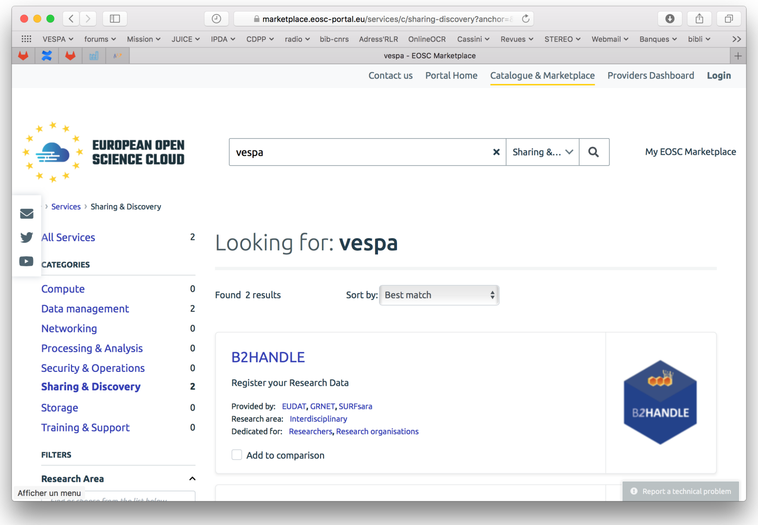
Git version control (servers config + service)

http://voparis-vespa-eosc.obspm.fr

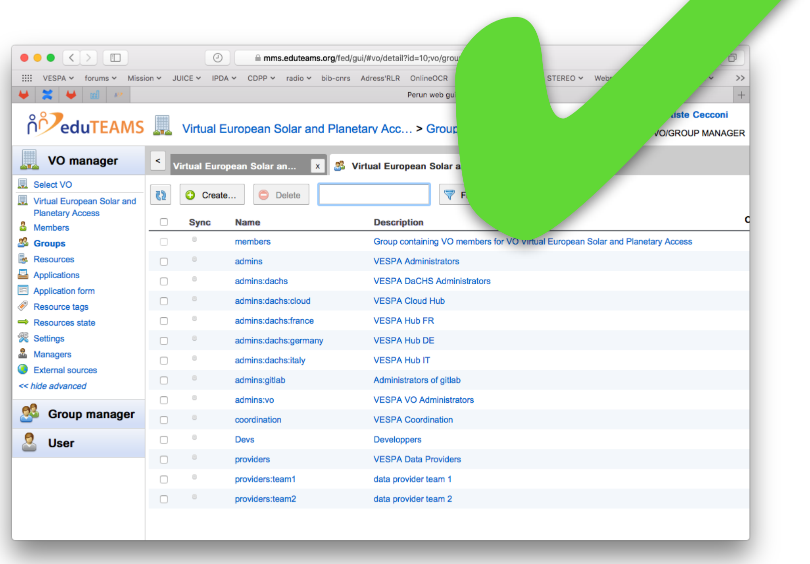
Service configuration and metadata management

- *Provider* orders a VESPA-Cloud service
- *VESPA-Cloud* registers *Provider* in VESPA-Cloud VO (eduTEAMS)
- *Provider* is authenticated through eduTEAMS to access GitLab server, and authorised based on group membership (*virtual organisation*, or sub-group), as managed by *virtual organisation* admins.
- *Provider* manages his service configuration and metadata in GitLab
- *VESPA-Cloud* deploys the service configuration on server instances (DaCHS on EGI, Storage on B2SAFE…)
- Deployment through docker (to be independent of underlying OS availability at data center)

Marketplace

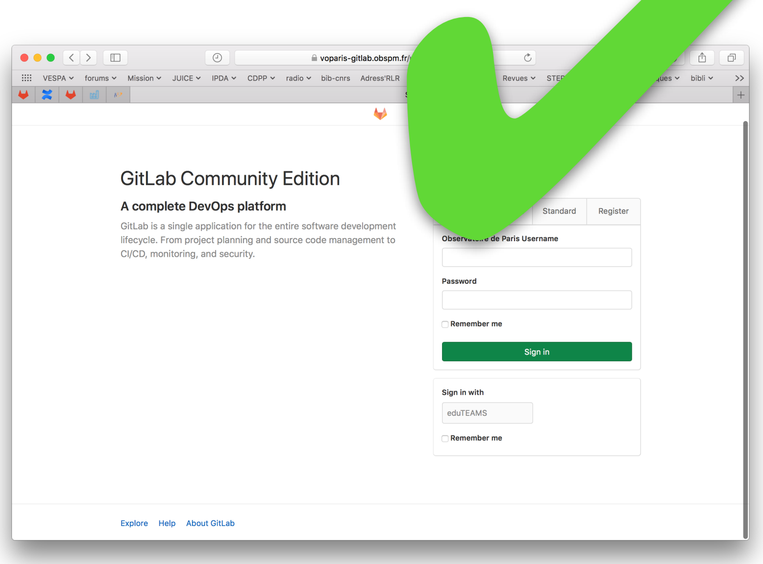


eduTEAMS



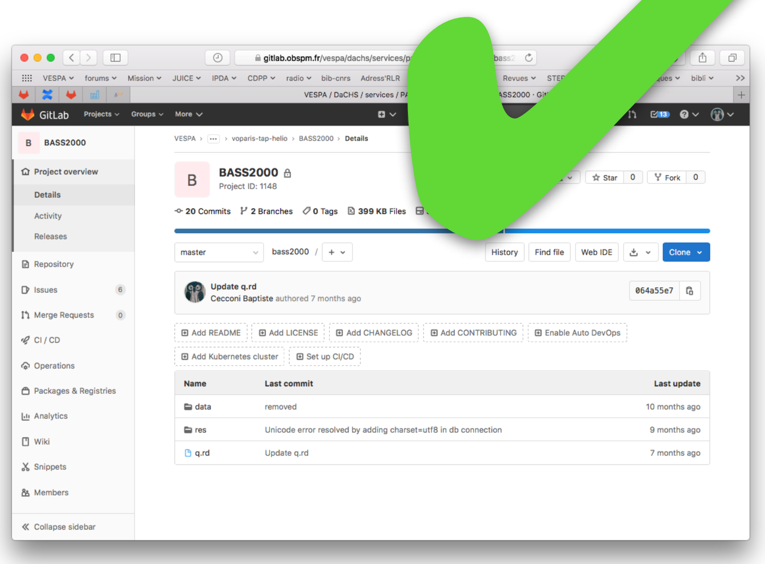
AAI

GitLab



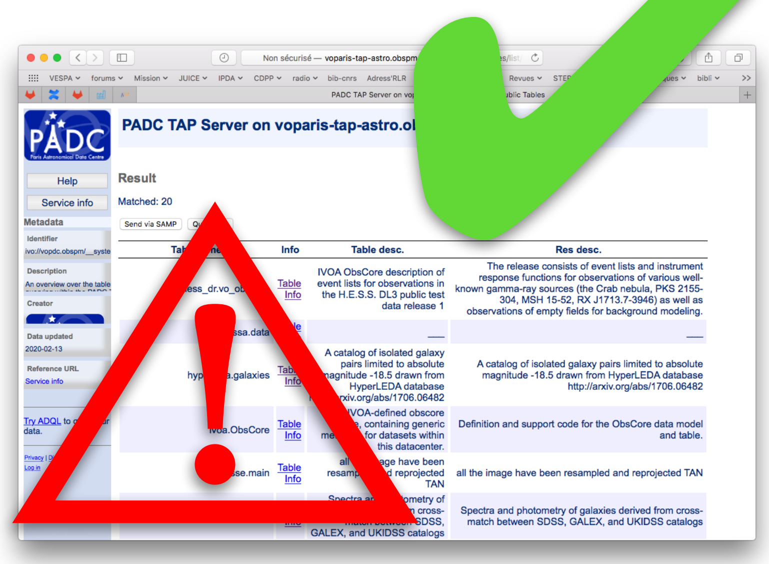
server config.

GitLab



service config.

DaCHS



openstack deployment (not failsafe, redeploy script needed)

VESPA

accessible and findable

VESPA-Cloud

Virtual European Solar and Planetary Access

- DaCHS on Docker
  - Whole configuration in a YAML file, from a simple Debian docker
  - Installation and configuration with docker-compose
  - Configuration stored in gitlab repo: registry metadata, logo, data service…

- Prototyping repository
  https://gitlab.obspm.fr/phamy/voparis-tap-planeto

- Docker installation on a distant VM

  Host configuration, install Docker and Docker-compose:
  - example on debian 10 or 11

```
sudo apt install docker docker.io wmdocker
```

  - clone/create git with all configuration file
  - in conf/ directory:

```
sudo docker-compose up -d
```

- Configurations files on the git repository
  - **.env** : define variables, which will be used in docker-compose.yml
  - **docker-compose.yml** : define all dockers and build params
  - **dachs-dockerfile** : define command lines to build DaCHS
  - **docker-entrypoint.sh** : script use on docker start
  - **secrets / id_rsa** : Never commit your private key in git !!!

  - **defautmeta.txt** : metadata for your DaCHS instance
  - **gavo.rc** : global configuration for gavo
  - apache configuration file
  - awstats configuration file
  - and most important config file… the logo

- **Docker compose detailed steps**:
  - mapping of port 80 for Dachs and 8080 for AWSTATS which gives the consultation stats
  - install the necessary packages
  - install dachs repository in debian
  - install Awstats and apache2 package and geoip package
  - install of dachs2
  - install public/private keys on the docker
  - clone the gitlab repository
  - set up dachs and awstats gitlab conf files -> final place

- **Extra steps**:
  - launch the *docker-entrypoint.sh* script
  - restart *ssh* and *postgres* to be sure
  - ingest the service data into DaCHS
  - command to publish the service in the local registry
  - restart DaCHS and Apache for security
  - place the gitlab serveur DNS in known_hosts to avoid prompting and a private key that allows git clone

Any questions?

- **services** folder:

  This folder contains configuration files for the EPN-TAP service to be installed on the server (either as submodules, or directly in the folder).

- **conf/.env** folder:
  This file contains all environment variables used by Docker :

```
# variables to be used in docker-compose.yaml file
VERSION=latest
SERVER=voparis-tap-planeto
# repository without .git
GIT=voparis-tap-planeto
GITURL=https://gitlab.obspm.fr/login
```

- **conf/docker-compose.yml**
  - file describe container (number of network socket, external port use, build...)

```
version: "3.7"
    Services :
        Dachs :
          container_name: dachs-voparis
          image: dachs-voparis
          expose:                                              <— external port for docker
             - "80"
          ports:                                               <— link between host with docker
              - "80:80"
        Build :
          dockerfile: dachs-dockerfile                         <— docker configuration file
          args:                                                <— retrieve variable
            - "VERSION=${VERSION}"
        secrets:
          - source: id_rsa
```

- **conf/dachs-dockerfile**
  - Server configuration file, OS version, packages …
  - Always start by arguments (ARG), before actions (RUN) :

```
ARG VERSION
    FROM debian:$VERSION

    ARG SERVER
    ARG GITURL
    ARG GIT
    ARG HOMEPATH=/home/dachsroot

    RUN apt-get -y update
    RUN apt-get -y upgrade
    RUN apt-get install -y curl apt-utils
```

- **conf/dachs-dockerfile (continued)**
  - line-by-line build operation :
  - build KO because postgresql never start

```
    RUN service postgresql start
    RUN apt-get install -y git gavodachs2-server geoip-database...
```

  - build OK

```
    RUN service postgresql start && apt-get install -y git gavodachs2-
server geoip-database...
```

  Configuration finish by entrypoint configuration and « tail -f » for maintains the docker bubble up

```
- RUN echo "git pull" >> /usr/bin/docker-entrypoint.sh
  ENTRYPOINT /usr/bin/docker-entrypoint.sh && tail -f /dev/null
```

- **conf/docker-entrypoint.sh**
  - This is a simple bash script, executed at the end step of build.
  - use it for start services (EPN-TAP), clone git repository, paste data for dachs ...

```
#!/usr/bin/env bash
# script use on start
service ssh restart && service postgresql restart

for rep in $(find /var/gavo/inputs/[0-9a-zA-Z]* -maxdepth 0 -type d | cut -f5 -d'/')
do
   su - dachsroot bash -c "dachs imp ${rep}/q.rd"
   su - dachsroot bash -c "dachs pub //services"
   su - dachsroot bash -c "dachs pub //tap"
   su - dachsroot bash -c "dachs pub ${rep}/q.rd"
done
dachs serve restart && service apache2 restart
```