



HAL
open science

Deploying container-based applications on EGI with VIP

Sandesh Patil, Axel Bonnet, Sorina Camarasu-Pop, Tristan Glatard

► **To cite this version:**

Sandesh Patil, Axel Bonnet, Sorina Camarasu-Pop, Tristan Glatard. Deploying container-based applications on EGI with VIP. EGI2022 - EGI conference, Sep 2022, Prague, Czech Republic. pp.1-1. hal-03792703

HAL Id: hal-03792703

<https://hal.science/hal-03792703v1>

Submitted on 30 Sep 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Deploying container-based applications on EGI with VIP

Sandesh Patil, Axel Bonnet, Sorina Camarasu-Pop, Tristan Glatard

VIP : Virtual Imaging Platform

- Free and open platform for the simulation and processing of medical images
 - 1400+ registered users;
 - 20+ applications available as a service;
 - Web portal at <https://vip.creatis.insa-lyon.fr/>
- Uses EGI resources from the biomed Virtual Organization to provide users with transparent access to high-throughput computing (though the DIRAC framework).



How it works?

- VIP [ref. 1] relies on Boutiques [ref. 3] to facilitate application deployment and execution on EGI resources. [fig. 1]
- Applications are fully described through Boutiques JSON descriptors.
- Descriptors point to Linux containers to facilitate application installation and sharing.

Containers

- Docker** containers are very popular, but the Docker daemon requires root privileges, preventing its support on HPC and HTC infrastructures.
- Singularity** answers this problem and thus popular in HPC centers. However, on EGI, it comes in variety of versions and configurations, which might prevent its seamless use.
- Udocker** [ref. 2] is another alternative, which can be installed on the fly without root privileges.

Container pre-deployment with udocker on CVMFS

- Udocker can pull the images (udocker pull <image_name>) on the fly from a central hub (e.g docker hub) on the EGI worker nodes [fig. 2]
- However, this may cause network issues for larger number of parallel jobs
- One alternative is to pre-deploy containers on CVMFS (CernVM File System) [ref. 4].
 - The image is pulled from the hub, then converted into a container (udocker create <container_name>).
 - The container is stored on CVMFS shared storage for future usage.
 - VIP jobs are then able to use udocker with the pre-deployed containers.

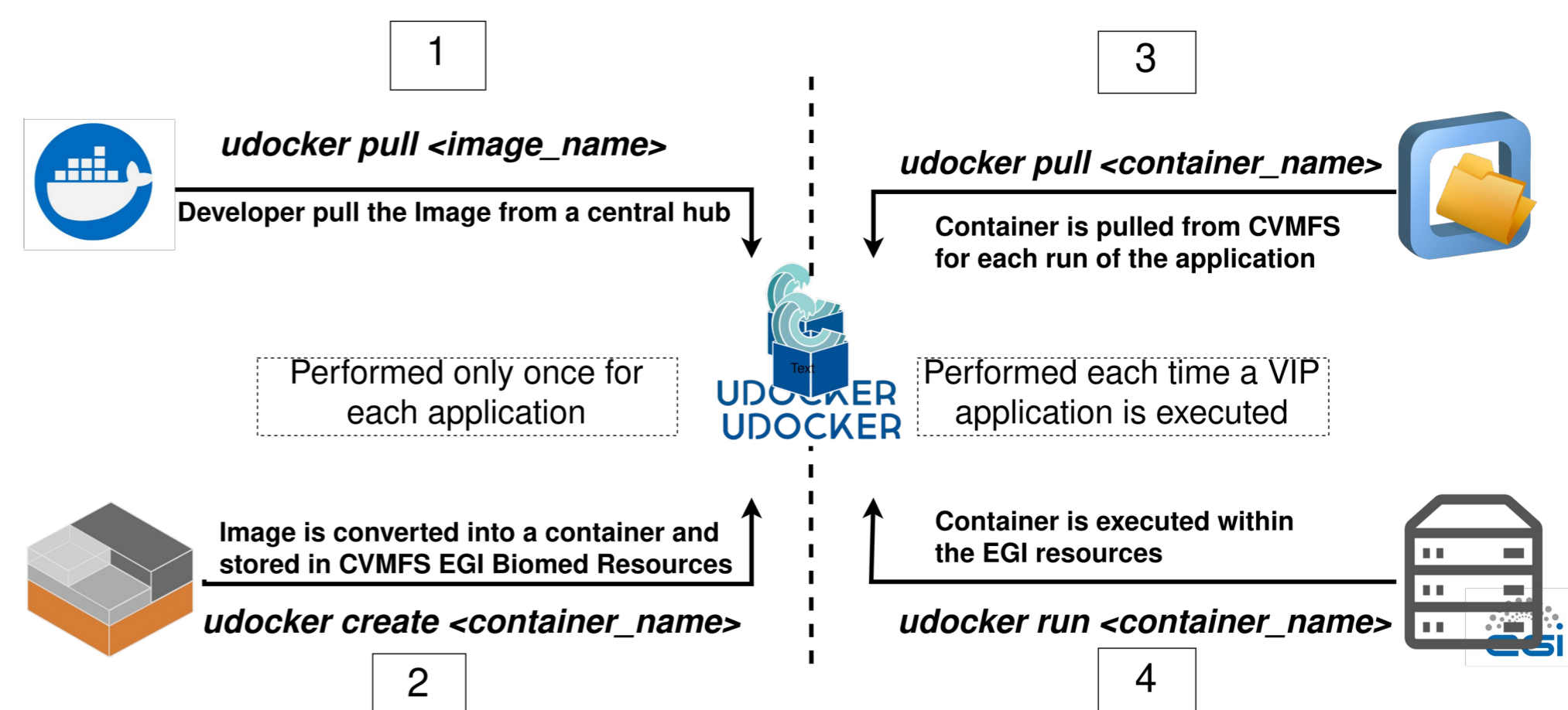


Fig. 2 – Pre-deployment of containers on CVMFS.

Conclusion and perspectives

- VIP is used by more than 1400 users for more than 20 applications of medical simulation and imaging, which are executed on EGI resources using linux containers.
- Containers are managed with udocker, which is deployed on the fly on EGI Worker Nodes.
- For some VIP applications, containers are pre-deployed on CVMFS.
- Pre-deployment of containers on CVMFS for production usage is very useful, but it may be cumbersome for testing and development.
- Having an EGI dedicated container registry could be a good alternative for testing and development usage.

References

- VIP : Glatard, Tristan, et al. 2013. « A Virtual Imaging Platform for Multi-Modality Medical Image Simulation ». IEEE Transactions on Medical Imaging 32 (1): 110-18
- Udoker : <https://github.com/indigo-dc/udocker>
- Boutiques : academic.oup.com/gigascience/article/7/5/giy016/4951979
- CVMFS : <https://cvmfs.readthedocs.io/en/stable/>

Funding

- M Patil is funded by the French program “Investissement d’Avenir” run by the Agence Nationale pour la Recherche (ANR-11-INBS-0006).

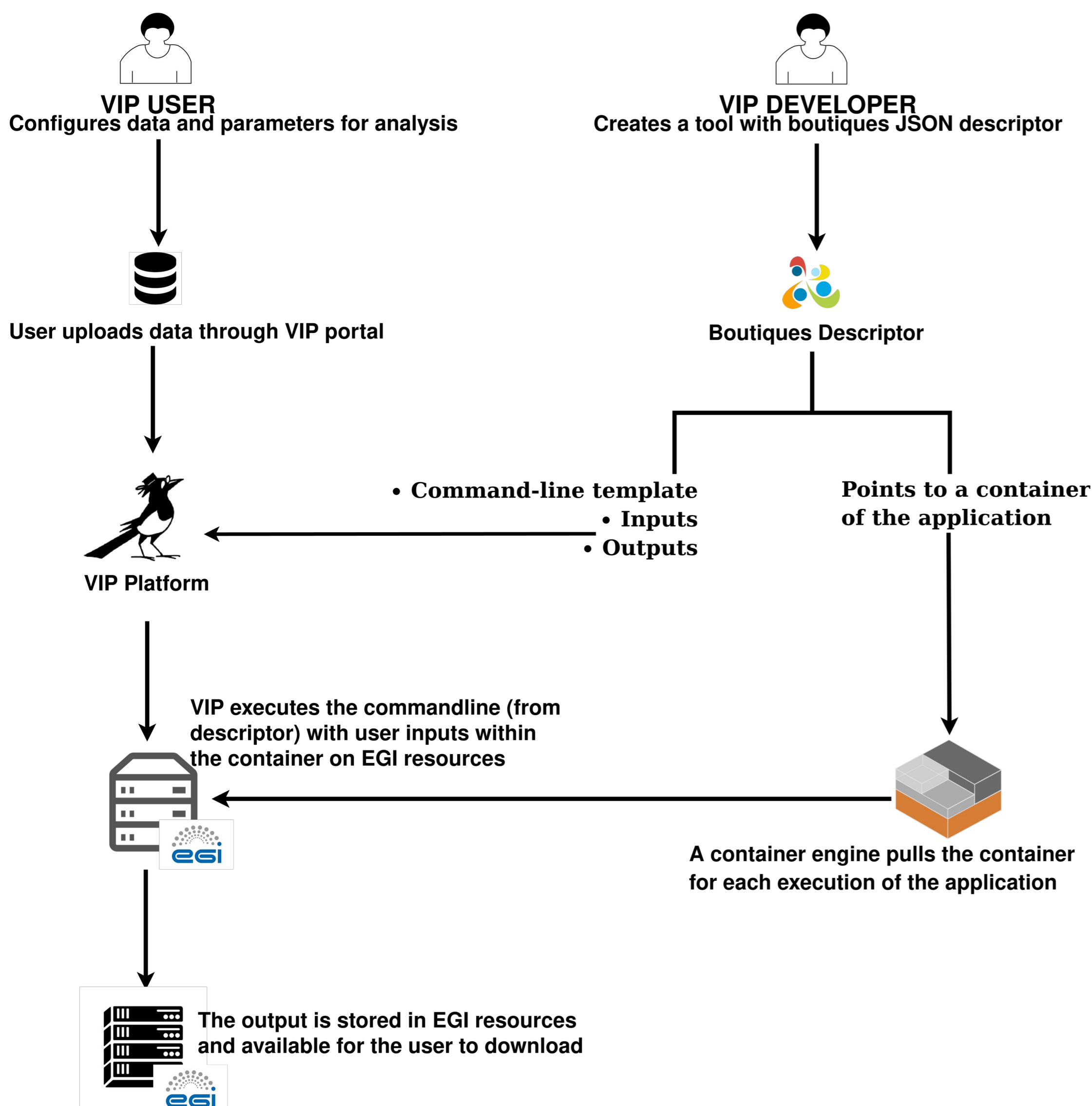


Fig. 1 – VIP relies on Boutiques and Linux containers for application deployment.