The Journal of Open Source Software

DARE Platform: a Developer-Friendly and Self-Optimising Workflows-as-a-Service Framework for e-Science on the Cloud

Iraklis A. Klampanos[*1], **Chrysoula Themeli**[1], **Alessandro Spinuso**[2], **Rosa Filgueira**[3], **Malcolm Atkinson**[3], **André Gemünd**[4], **and Vangelis Karkaletsis**[1]

1 National Centre for Scientific Research "Demokritos", Greece **2** Koninklijk Nederlands Meteorologisch Instituut, the Netherlands **3** The University of Edinburgh, UK **4** Fraunhofer-Institut für Algorithmen und Wissenschaftliches Rechnen (SCAI), Germany

DOI: 10.21105/joss.02664

Software

- Review
- Repository
- Archive

Editor: Daniel S. Katz

Reviewers:

- @rafaelfsilva
- @Himscipy

Submitted: 05 August 2020
Published: 16 October 2020

Statement of need

In recent years, science has relied more than ever on large-scale data as well as on distributed computing and human resources. Scientists and research engineers in fields such as climate science and computational seismology, constantly strive to make good use of remote and largely heterogeneous computing resources (HPC, Cloud, institutional or local resources, etc.), process, archive and analyse results stored in different locations and collaborate effectively with other scientists.

The DARE platform enables the seamless development and reusability of scientific workflows and applications, and the reproducibility of the experiments. Further, it provides Workflow-as-a-Service (WaaS) functionality and dynamic loading of execution contexts in order to hide technical complexity from its end users. This paper introduces the software implementing the DARE platform. More information on the H2020 DARE project is provided in Klampanos et al. (2019), Atkinson et al. (2019), and Atkinson et al. (2020).

The DARE platform

The DARE platform is designed to live in-between user applications and the underlying computing resources. It is built on top of containerisation as well as parallelisation technologies, e.g., Kubernetes and MPI. Interfacing with client systems and end-users is achieved via RESTful APIs. The execution of scientific workflows is achieved via a Workflows-as-a-service layer, which can handle workflows described in either the dispel4py Python library (Filgueira, Krause, Atkinson, Klampanos, & Moreno, 2017), or in the Common Workflow Language (CWL) (Amstutz et al., 2016).

The software

The DARE platform consists of a number of largely independent software components developed by the partners of the DARE project. All core software components are provided via the DARE GitLab group. The DARE Platform repository provides pointers to all relevant repositories, documentation and more. Installation instructions and API documentation are

*Corresponding author.

provided in a GitLab page. A demo is available in the DARE Execution API GitLab Repository, which can also be used as an integration test.

The DARE platform and its components are published with the Apache 2.0 License. Everyone is welcome to download, deploy, and modify the source code, as well as to propose bug fixes and changes, either by creating issues or by contributing source code. The most straightforward way to contribute code to the DARE platform and to its component repositories is by working on a fork and creating a pull request.

The core DARE platform components are the following:

dispel4py

dispel4py is a Python library for describing abstract stream-based workflows for distributed data-intensive applications. It can translate higher-level workflows to diverse computing contexts, such as Apache Storm, MPI and plain shared-memory multi-core, to enable moving seamlessly into production with large-scale data loads. More information can be found at the dispel4py repository.

s-ProvFlow

s-ProvFlow implements a provenance framework for storage and access of data-intensive streaming lineage. It offers a web API and a range of dedicated visualisation tools based on the underlying provenance model, S-PROV, which utilises and extends PROV and ProvONE models. Complete documentation for this component can be found at the s-ProvFlow repository.

dispel4py Registry

The dispel4py Registry is a RESTful Web service providing functionality for registering workflow entities, such as processing elements (PEs), functions and literals, while encouraging sharing and collaboration via groups and workspaces. More information is provided in the dispel4py Registry repository.

CWL Workflow Registry

The CWL Workflow Registry provides a similar functionality as the Dispel4py Registry, with the difference that it is associated with CWL workflows. More information is provided at the CWL workflow registry repository.

DARE Execution API

The DARE Execution API enables the distributed and scalable execution of dispel4py and CWL workflows, and is extensible to other contexts. The Execution API also offers services such as uploading/downloading and referencing of data and process monitoring. More information is provided in the Execution API repository.

DARE playground

The purpose of the playground is to provide an environment for testing and debugging purposes, especially dispel4py workflows. This helps users debug their methods before making them available for execution on the platform. More information is provided in the DARE playground repository.

Characteristics of the DARE platform

1. It interfaces with users and external systems via a comprehensive RESTful API.
2. It facilitates the development of modular, reusable and shareable data-intensive solutions.
3. It combines two different workflow approaches, dispel4py and CWL, within the same platform and development environment.
4. Via its execution API, it orchestrates the dynamic spawning and closing of MPI clusters on the cloud for MPI-enabled components.
5. It provides a flexible environment, which local administrators can parametrise, by supporting custom docker-based environments and user interfaces.
6. It supports the collection, mining and visualisation of provenance information.

DARE platform use cases

The DARE platform is currently used in the following domain applications:

1. Seismology: Rapid Assessment (RA) of ground motion parameters during large earthquakes.
2. Seismology: Moment Tensor 3D (MT3D) for ensemble-type of seismic modelling.
3. Volcanology: Ash fall hazard modelling.
4. Climate-change: Extending Climate4Impact with efficient and transparent access to diverse computing resources.
5. Atmospheric sciences: Cyclone tracking and visualisation application.

State of the field

The DARE platform implements research coming from multiple areas. This section is therefore not meant to be exhaustive but rather to provide basic state-of-the-field information for further study. The need for unifying underlying e-infrastructures and platforms via higher-level interfaces, programmatic or interactive, is especially pronounced in Europe due to the widespread policy and technological diversity. Generic technological solutions, such as the ones produced by the COLA project (Kiss, Terstyanszky, & others, 2018), move towards providing unifying low-level views of underlying infrastructures. However, to raise the level of abstraction for researchers also requires automation powered by tighter integration of heterogeneous components. Much of this functionality is powered by shared catalogues within and outside proposed technological solutions.

Using shared catalogues as a basis for integration is central to projects, such as VRE4EIC project, which has developed research environments for collaborating research communities (Martin, Remy, Theodoridou, Jeffery, & Zhao, 2019). Similar to DARE, the SWITCH project has demonstrated using knowledge-bases for supporting enactment-target selection, optimisation, mapping and coping with heterogeneity (Štefanič, Cigale, & others, 2019), focusing on time-critical applications.

In terms of leveraging the Cloud paradigm to raise the abstraction level, the project DEEP-Hybrid-DataCloud makes use of underlying data representation and transformation functionality to provide machine learning as a service to a variety of target user groups (López García, 2019). DEEP focuses on the exposure of computational resources, e.g. GPU clusters over federated Clouds. The PROCESS project has built a set of services and tools to enable extreme scale data processing in scientific and advanced industry settings. Similar to the DARE platform, PROCESS offers a set of composable services covering from data processing to workflow

specification and enactment. However DARE places more weight on supporting reflection via catalogues and registries to aid automation and optimisation.

Acknowledgements

This work has been supported by the EU H2020 research and innovation programme under grant agreement No 777413.

References

Amstutz, P., Crusoe, M. R., Tijanić, N., Chapman, B., Chilton, J., Heuer, M., Kartashov, A., et al. (2016). Common workflow language, v1.0. doi:10.6084/m9.figshare.3115156.v2

Atkinson, M., Filgueira, R., Gemünd, A., Karkaletsis, V., Klampanos, I., Koukourikos, A., Levray, A., et al. (2020, March). DARE architecture and technology internal report. Zenodo. doi:10.5281/zenodo.3697898

Atkinson, M., Filgueira, R., Klampanos, I., Koukourikos, A., Krause, A., Magnoni, F., Pagé, C., et al. (2019). Comprehensible control for researchers and developers facing data challenges. In *Proceedings of the 15th IEEE International Conference on eScience*. doi:10.1109/eScience.2019.00042

Filgueira, R., Krause, A., Atkinson, M., Klampanos, I., & Moreno, A. (2017). Dispel4py: A Python framework for data-intensive scientific computing. *International Journal of High Performance Computing Applications*. doi:10.1177/1094342016649766

Kiss, T., Terstyanszky, G., & others. (2018). Automated Scalability of Cloud Services and Jobs. In *10th international workshop on science gateways, IWSG 2018*. Edinburgh, UK.

Klampanos, I., Davvetas, A., Gemünd, A., Atkinson, M., Koukourikos, A., Filgueira, R., Krause, A., et al. (2019). DARE: A reflective platform designed to enable agile data-driven research on the cloud. In *2019 15th International Conference on eScience (eScience)* (pp. 578–585). doi:10.1109/eScience.2019.00079

López García, Á. (2019). DEEPaaS API: a REST API for Machine Learning and Deep Learning models. *Journal of Open Source Software*. doi:10.21105/joss.01517

Martin, P., Remy, L., Theodoridou, M., Jeffery, K., & Zhao, Z. (2019). Mapping heterogeneous research infrastructure metadata into a unified catalogue for use in a generic virtual research environment. *Future Generation Computer Systems*. doi:10.1016/j.future.2019.05.076

Štefanič, P., Cigale, M., & others. (2019). SWITCH workbench: A novel approach for the development and deployment of time-critical microservice-based cloud-native applications. *Future Generation Computer Systems*, *101*. doi:10.1016/j.future.2019.04.008