




# DASF: A data analytics software framework for distributed environments

Daniel Eggert <sup>1</sup>, Mike Sips <sup>1</sup>, Philipp S. Sommer <sup>2</sup>, and Doris Dransch<sup>1</sup>

<sup>1</sup> Helmholtz Centre Potsdam – GFZ German Research Centre for Geosciences <sup>2</sup> Helmholtz-Zentrum Hereon

DOI: [10.21105/joss.04052](https://doi.org/10.21105/joss.04052)

## Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Martin Fleischmann](#)  

## Reviewers:

- [@cjwu](#)
- [@pritchardn](#)

Submitted: 17 December 2021

Published: 13 October 2022

## License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

## Summary

The success of scientific projects increasingly depends on using data analysis tools and data in distributed IT infrastructures. Scientists need to use appropriate data analysis tools and data, extract patterns from data using appropriate computational resources, and interpret the extracted patterns. Data analysis tools and data reside on different machines because the volume of the data often demands specific resources for their storage and processing, and data analysis tools usually require specific computational resources and run-time environments. The data analytics software framework DASF, which we develop in Digital Earth (Bouwer et al. (2022)), provides a framework for scientists to conduct data analysis in distributed environments.

## Statement of need

The data analytics software framework DASF supports scientists to conduct data analysis in distributed IT infrastructures by sharing data analysis tools and data. For this purpose, DASF defines a remote procedure call (RPC, White (1976)) messaging protocol that uses a central message broker instance. Scientists can augment their tools and data with this protocol to share them with others. DASF supports many programming languages and platforms since the implementation of the protocol uses WebSockets. It provides two ready-to-use language bindings for the messaging protocol, one for Python and one for the Typescript programming language. In order to share a python method or class, users add an annotation in front of it. In addition, users need to specify the connection parameters of the message broker. The central message broker approach allows the method and the client calling the method to actively establish a connection, which enables using methods deployed behind firewalls. DASF uses Apache Pulsar (Apache-Pulsar (2022)) as its underlying message broker.

The Typescript bindings are primarily used in conjunction with web frontend components, which are also included in the DASF-Web library. They are designed to attach directly to the data returned by the exposed RPC methods. This supports the development of highly exploratory data analysis tools. DASF also provides a progress reporting API that enables users to monitor long-running remote procedure calls.

One application using the framework is the Digital Earth Flood Event Explorer (Eggert et al. (2022)). The Digital Earth Flood Event Explorer integrates several exploratory data analysis tools and remote procedures deployed at various Helmholtz centers across Germany.

## State of the field

DASF aims at connecting various data analysis tools and methods, as well as data visualization components in a distributed environment. A chain of connected analysis tools and visualizations can be seen as a data analysis workflow, so Workflow Engines like Galaxy (Afgan et al. (2018)), Kepler (Kepler (2022)), Taverna (Taverna (2022)) and Pegasus (Pegasus (2022)) could be used to model, implement and connect the individual tools and visualizations. Yet, the intellectual hurdles to be mastered when dealing with workflow systems are relatively high and the systems often do not offer much flexibility in case of distributed deployment of individual methods. In contrast, common Web Frameworks, like Django (Django (2022)), Flask (Ronacher (2022)) or GWT (GWT-Open-Source-Project (2022)) could also be used to connect multiple analysis methods and visualization components into an integrated data analysis chain, but usually such frameworks are limited to a single programming language, like Python or Java and only support a single backend and are therefore not well suited for analysis methods deployed in distributed infrastructures.

## Structure

The Data Analytics Software Framework (DASF) facilitates using data analysis tools in distributed IT infrastructures. The framework consists of three major modules:

DASF-Web (Eggert (2021b)) collects all web components for the data analytics software framework DASF. It provides ready-to-use interactive data visualization components like time series charts, radar plots, stacked-parameter-relation (spr), and map components to support the visual analysis of spatio-temporal data. Moreover, DASF-Web includes the web bindings for the DASF RPC messaging protocol. It is implemented in Typescript and uses Vuejs/Vuetify, Openlayers and D3 as a technical basis.

DASF-Messaging-Python (Eggert & Sommer (2021)) is a RPC (remote procedure call) wrapper library for the python programming language. As part of the data analytics software framework DASF, it implements the DASF RPC messaging protocol.

DASF-Progress-API (Eggert (2021a)) provides a lightweight tree-based structure to be sent via the DASF RPC messaging protocol. Its generic design supports deterministic as well as non-deterministic progress reports. While DASF-Messaging-Python provides the necessary implementation to distribute the progress reports from the reporting backend modules, DASF-Web includes ready-to-use components to visualize the reported progress.

## Acknowledgements

We acknowledge funding from the Initiative and Networking Fund of the Helmholtz Association through the project Digital Earth.

## References

- Afgan, E., Baker, D., Batut, B., Beek, M. van den, Bouvier, D., Čech, M., Chilton, J., Clements, D., Coraor, N., Grüning, B. A., Guerler, A., Hillman-Jackson, J., Hiltmann, S., Jalili, V., Rasche, H., Soranzo, N., Goecks, J., Taylor, J., Nekrutenko, A., & Blankenberg, D. (2018). The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. *Nucleic Acids Res.*, 46(W1), W537–W544. <https://doi.org/10.1093/nar/gky379>
- Apache-Pulsar. (2022). Apache pulsar: A cloud-native, distributed messaging and streaming platform originally created at yahoo! And now a top-level apache software foundation project. In *Website*. Apache Software Foundation. <https://pulsar.apache.org/>

- Bouwer, L. M., Dransch, D., Ruhnke, R., Rechid, D., Frickenhaus, S., & Greinert, J. (2022). *Integrating data science and earth science: Challenges and solutions* (L. M. Bouwer, D. Dransch, R. Ruhnke, D. Rechid, S. Frickenhaus, & J. Greinert, Eds.). Springer Nature. <https://doi.org/10.1007/978-3-030-99546-1>
- Django. (2022). Django: The web framework for perfectionists with deadlines. In *Website*. Django Software Foundation. <https://www.djangoproject.com/>
- Eggert, D. (2021a). DASF-progress-API: A progress reporting structure for the data analytics software framework. In *Gitlab repository*. Gitlab. <https://git.geomar.de/digital-earth/dasf/dasf-progress-api>
- Eggert, D. (2021b). DASF-web: Web components for the data analytics software framework. In *Gitlab repository*. Gitlab. <https://git.geomar.de/digital-earth/dasf/dasf-web>
- Eggert, D., Rabe, D., Dransch, D., Lüdtkke, S., Nam, C., Nixdorf, E., Wichert, V., Abraham, N., Schröter, K., & Merz, B. (2022). *The digital earth flood event explorer: A showcase for data analysis and exploration with scientific workflows*. GFZ Data Services. <https://doi.org/10.5880/GFZ.1.4.2022.001>
- Eggert, D., & Sommer, S. P. (2021). DASF-messaging-python: A python RPC wrapper for the data analytics software framework. In *Gitlab repository*. Gitlab. <https://git.geomar.de/digital-earth/dasf/dasf-messaging-python>
- GWT-Open-Source-Project. (2022). GWT: Google web toolkit. In *Website*. GWT Open Source Project. <https://www.gwtproject.org/>
- Kepler. (2022). The kepler projekt: Your science. enabled. In *Website*. <https://kepler-project.org/>
- Pegasus. (2022). Pegasus: Makes the work flow. In *Website*. <https://pegasus.isi.edu/>
- Ronacher, A. (2022). Flask: Web development, one drop at a time. In *Website*. The Pallets Projects. <https://palletsprojects.com/p/flask/>
- Taverna. (2022). Taverna: A domain-independent suite of tools used to design and execute data-driven workflows. In *Website*. <https://taverna.incubator.apache.org/>
- White, J. E. (1976). RFC 707. A high-level framework for network-based resource sharing. *Proceedings of the 1976 National Computer Conference*.