Reproducible Earth observation analytics: challenges, ideas, and a study case on containerized land use change detection

Marius Appel, Daniel Nüst, Edzer Pebesma
Institute for Geoinformatics, University of Münster, Germany
Contact: marius.appel@uni-muenster.de

The EO analytics workflow

Geoscientific analyses of Earth observation data typically involve a long path from data acquisition to scientific results and conclusions (Figure 1) . Before starting the actual processing, scenes must be downloaded from the providers' platforms and the computing infrastructure needs to be prepared. The computing environment often requires specialized software, which in turn might have lots of dependencies. The software is also highly customized and provided without commercial support, which leads to rather ad-hoc systems and irreproducible results.

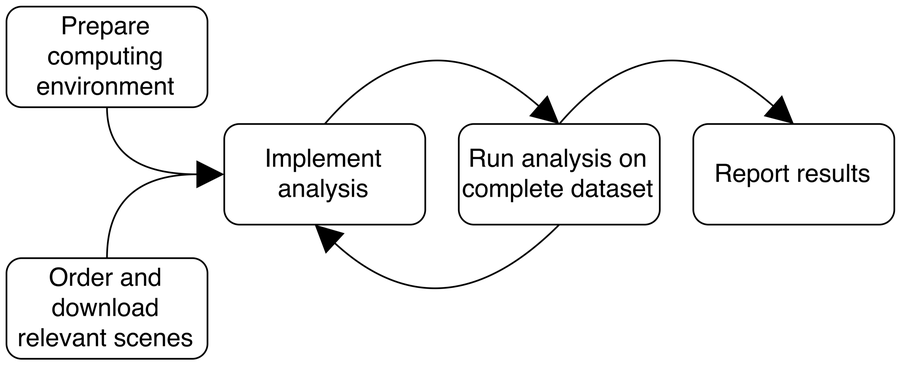

Figure 1: Typical EO analytics workflow.

Reproducibility with containerization

To let other scientists reproduce the analyses, full workspaces including data, code, the computing environment, and documentation must be bundled and shared. Technologies such as virtualization or containerization allow for the creation of identical computing environments with relatively little effort. Especially Docker is increasingly used in reproducible research [5,6].

References

[1] Verbesselt, J., Zeileis, A., & Herold, M. (2012). Near real-time disturbance detection using satellite image time series. *Remote Sensing of Environment*, *123*, 98-108.
[2] Appel M., Lahn F., Pebesma E., Buytaert W., Moulds S. (2016). Scalable Earth-observation Analytics for Geoscientists: Spacetime Extensions to the Array Database SciDB. accepted for poster presentation at EGU General Assembly 2016, Vienna, Austria April 17-22, 2016.
[3] https://appelmar.github.io/scalbf-wur/tutorial/tutorial_part3.html
[4] Stonebraker, M., Brown, P., Zhang, D., & Becla, J. (2013). SciDB: A database management system for applications with complex analytics. Computing in Science & Engineering, 15(3), 54-62.
[5] Nüst, D., Konkol, M., Schutzeichel, M., Pebesma, E., Kray, C., Przibytzin, H., & Lorenz, J. (2017). Opening the Publication Process with Executable Research Compendia. D-Lib Mag.
[6] Knoth, C., Nüst, D., 2017. Reproducibility and Practical Adoption of GEOBIA with Open-Source Software in Docker Containers. Remote Sensing 9, 290. doi:10.3390/rs9030290

Study Case: A Docker image for reproducible land-use change detection

We created a Docker image that

- uses SciDB [3,4] to efficiently store and process Landsat image time series as a three-dimensional array and
- executes an R script within SciDB by reusing the existing implementation of the BFAST algorithm [1] to monitor changes in NDVI time series over a region in south west Ethiopia.

The image can be used for

- ***automatic reproduction***, where the complete analysis is carried out within a container and
- ***interactive experimentation***, where software including SciDB run as services and users get access to RStudioServer.

Building the image prepares the complete platform, whereas the analysis runs in a container that receives scenes as input and produces a report of the results as output (Figure 2). The Dockerfile is available as open-source at https://github.com/appelmar/scidb-eo-egu2017.

Technologies

docker R SciDB

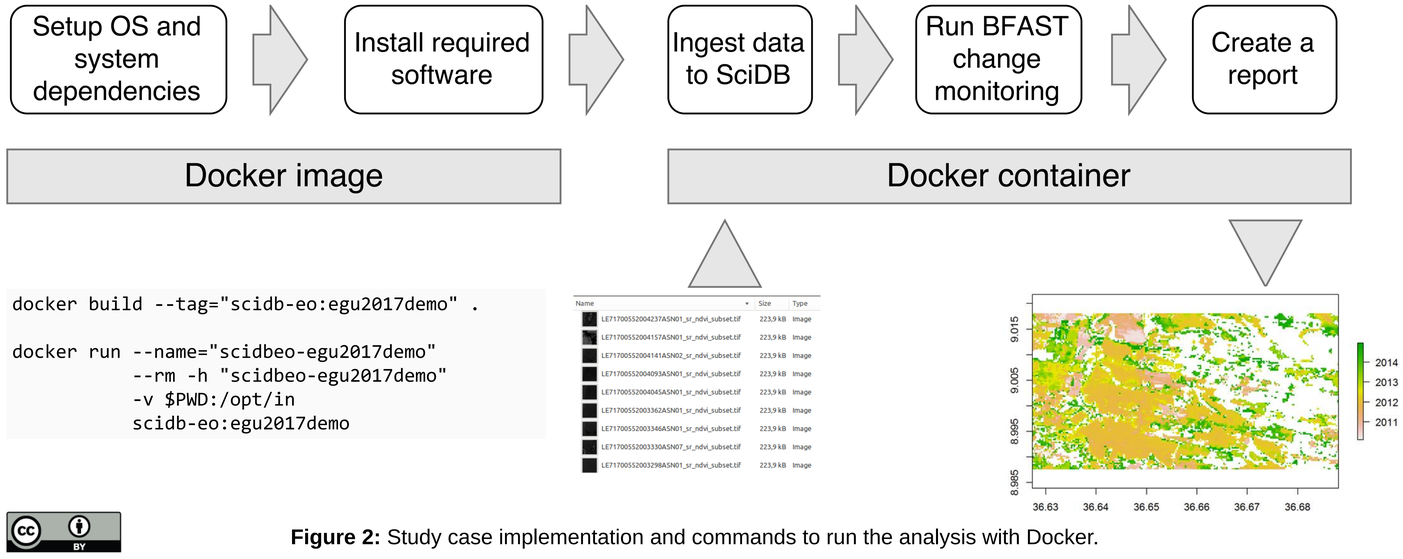

Figure 2: Study case implementation and commands to run the analysis with Docker.

Challenges

Data volume and distributed processing: Latest EO missions such as Copernicus with its Sentinel satellites require distributed processing. How to reproduce these distributed computing environments is complex and goes beyond single container environments.

Data access: Downloading EO data from providers is inefficient. EO data centers where reproducible containers can be deployed are an alternative.

Conclusions

- Containerization with Docker can be used to make EO analytics fully reproducible
- Even complex software such as the array-based data management system SciDB can be packaged for easy use within containers
- Analytics on large datasets requires multi-container environments and recipes to create these (e.g. as distributed application bundles, or service stacks)
- Dedicated EO data centers where complete stacks can be deployed are needed