Software Infrastructure on the publishing of planetary geo-morphological maps

Carlos H. Brandt[1*], Luca Penasa[2], Angelo P. Rossi[1], Giacomo Nodjoumi[1] for the GMAP consortium
[1] Jacobs University Bremen, Bremen, Germany (*c.brandt@jacobs-university.de),
[2] Istituto Nazionale di Astrofisica, Padova, Italy

Introduction: In the quest for FAIR (Findable, Accessible, Interoperable, Reproducible) [7] data it is fundamental to work out solutions the serve the data consumer as well as the data producer through a suitable suite of standards, software, and workflows. On one hand, data producers are busy with collecting ancillary data for their final product, that will go public typically through an article describing it; On the other, data consumers are looking for all the available data suitable to support their work; The interface between those two scientists, how they communicate their results and their demands, is of great importance to make the data flow in between as easy as possible. In GMAP [1], we are developing software, deploying systems, and preparing data to cover the gap between producers and consumers of planetary geo-morphological maps.

As highlighted in [2] geological maps are a sensible research product to the discussion of data FAIRness for their richness of information and great format freedom. Findability and Accessibility are covered by data publication platform such as Zenodo [12], Astropedia [4], or PANGAEA [5,6], by providing easy-to-use interactive, searchable web interfaces to data stores to metadata rich research products. Interoperability and Reusability, on the other hand, are aspects that demand support at a lower-level: in the making of the maps so the final product can cope to a sufficient level of homogeneity regarding their metadata as well as digital formats of distribution.

In this poster, we present the system we are implementing for the GMAP community to support the community of planetary scientist and students in their production of geo-morphological maps through the establishment of communication channels but also hosting of software and the data products.

System architecture: In Figure 1, we see the structure of GMAP systems, composed by the following services:

- *User space*[1]: Gitlab is used for users authentication by other services (e.g, Jupyter notebooks), and for project management and internal communication;
- *Research Data Management*[2]: InvenioRDM provides the research data management repository interface;
- *Data Processing*[3]: JupyterHub provides an environment for planetary data processing (see poster from Nodjoumi et al, [3]);
- Wiki[4]: for internal documentation;
- Website: https://europlanet-gmap.eu .

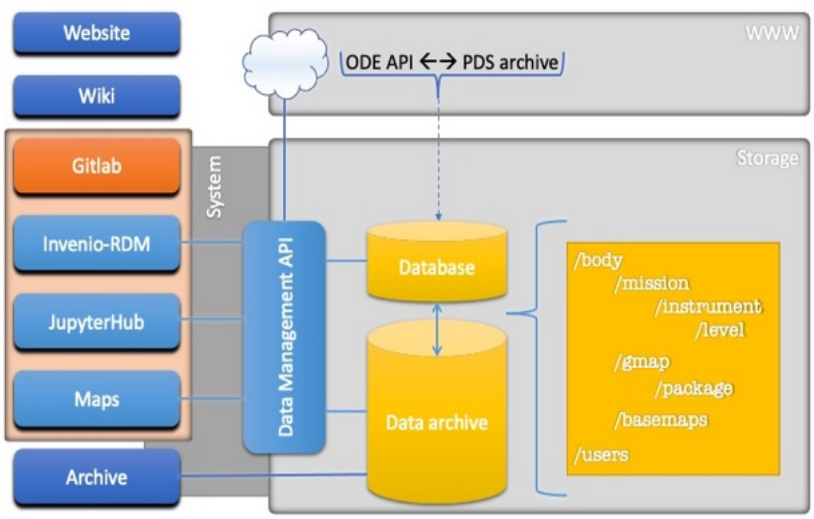

Figure 1: GMAP System Architecture

On Gitlab we keep the database of users, besides providing a space for users software projects it provides the authentication service for the other GMAP data and processing services (i.e, Invenio and JupyterLab). Users are registered on a *per-request* mode, optionally whole institution domain-names (e.g, 'unipd.it') can be

[1] https://git.europlanet-gmap.eu
[2] https://data.europlanet-gmap.eu
[3] https://jupyter.europlanet-gmap.eu
[4] https://wiki.europlanet-gmap.eu

allowed to provide open access to their members. With InvenioRDM we are able to provide a data publication platform suitable for our data products and directly linked to our storage system and Python libraries. The JupyterLab/Hub service complements the user cloud space by providing a fully functional environment for planetary data processing, including GDAL, ISIS, ASP and all known Python geospatial data science tools (see poster from Nodjoumi *et al*. [2]).

Software and Data: Handling the data and supporting the users processing and management of it there a number of software libraries that have being developed by the GMAP team:

- JupyterLab: the setup deployed on GMAP systems is portable and openly available for any individual or institution willing to host the same service on their premises [9];
- Mappy: a QGIS Python plugin for easy map edition [8];
- GPT: a Python library for geo-planetary data handling [10];
- Shoosh: a Python library for remote in-docker containers processing abstraction [11].

And numerous Jupyter notebooks making use of these and third-party software as example material for our community.

Conclusion: The GMAP system is deployed to provide a one-stop solution for planetary data scientists and is implemented having portability and easy-of-use as guidelines. To the matters of *reproducibility* and *interoperability*, we take particular attention to publish the very same software installation used in our premises as well as relying on open standards for data formats.

References:

[1] Nass A. *et al*. (2021) *GMAP – European Mapping efforts for Geologic Mapping of Planetary bodies,* European Planetary Science Congress 2021, https://doi.org/10.5194/epsc2021-383 .

[2] Nass A. *et al*. (2021) *Facilitating reuse of planetary spatial research data – Conceptualizing an open map repository as part of a Planetary Research Data Infrastructure*, Planetary and Space Science (v.204), https://doi.org/10.1016/j.pss.2021.105269 .

[3] Nodjoumi G. *et al*. (2022) *Open-Source Planetary Data Processing Environments Based on JupyterHub and Docker Containers*, Planetary Science Informatics and Data Analytics Conference 2022.

[4] Bailen M.S. (2012) *Astropedia – A Data Portal for Planetary Science*, 43rd Lunar and Planetary Science Conference, https://www.lpi.usra.edu/meetings/lpsc2012/pdf/2478.pdf .

[5] Devaraju A. *et al.* (2018) *PANGAEA Data Recommender: An Out of the Box Approach to Discover Earth and Environmental Datasets*, EGU General Assembly 2018.

[6] Diepenbroek M. *et al.* (2002) *PANGAEA – An Information System for Environmental Sciences*, Computer & Geosciences, 29(10), https://doi.org/10.1016/S0098-3004(02)00039-0 .

[7] Wilkinson, M.D. *et al.* (2016) *The FAIR Guiding Principles for scientific data management and stewardship*, Scientific Data 3, https://doi.org/10.1038/sdata.2016.18 .

[8] Penasa L. *et al.* (2020) *Constructing and deconstructing geological maps: a QGIS plugin for creating topologically consistent geological cartography*, European Planetary Science Congress 2020, https://doi.org/10.5194/epsc2020-1057 .

[9] *Docker-ISIS3* code repository: https://github.com/europlanet-gmap/docker-isis3 .

[10] *GeoPlanetary Tools* code repository: https://github.com/chbrandt/gpt .

[11] *Shoosh* code repository: https://github.com/chbrandt/shoosh .

[12] Zenodo website: https://www.zenodo.org/ .

Acknowledgments: GMAP is part of Europlanet 2024 RI, funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 871149.