EPSC Abstracts
Vol. 12, EPSC2018-37, 2018
European Planetary Science Congress 2018

The Web-based Interactive Mars Analysis and Research System for the iMars project

Sebastian. H. G. Walter (1), Jan-Peter Muller (2), Panagiotis Sidiropoulos (2), Yu Tao (2), Klaus Gwinner (3), Alfiah R. D. Putri (2), Jung-Rack Kim (4), Ralf Steikert (1), Stephan van Gasselt (1,4), Greg G. Michael (1), Gillian Watson (2), Björn P. Schreiner (1)

(1) Freie Universität Berlin, Germany (2) University College London, United Kingdom (3) German Aerospace Center (DLR), Germany (4) University of Seoul, South Korea (s.walter@fu-berlin.de)

Abstract

We describe the web-based interactive Mars Analysis and Research System (iMARS web-GIS), specialized on planetary surface change analysis with novel tools for simultaneous visualization of single images as time series in their original sequence. As the data foundation, we use the vast quantity of automatically co-registered ortho-images and digital terrain models (DTM) from three NASA missions' instruments processed and ingested in the context of the EU-funded iMars project [1]. The baseline for the co-registered images are the High Resolution Stereo Camera (HRSC) multi-orbit quadrangle image mosaics, which are based on bundle-block-adjusted multi-orbit DTM mosaics. Additionally we make use of the existing along-track bundle-adjusted HRSC single images and DTMs available at the planetary data archives. A web mapping application including the presented functionality has been implemented and is available at http://imars.planet.fu-berlin.de with the iMars project website (http://www.i-mars.eu/web-gis) serving as a mirror. The system has been built by using solely open source software, and the additional developments are publicly available.

1. Introduction

To achieve change detection visualization functionality for planetary application, we want to be able to display all co-registered images available from a certain spot on the surface in their full spatial resolution – and to show them together animated as a time series. In-depth analysis of the data requires functionality such as switching quickly between layers, known as flickering (in the case of two layers of images) or cycling (more than two layers).

2. Methods

The iMARS web-GIS comprises a browser-server infrastructure which loads dynamic code from a website and connects to web services of a server-based backend. The web services stream pre-processed data from connected databases and storage subsystems according to the standards of the OGC [2], adapted for planetary reference systems [3]. The iMARS web content (HTML, CSS) including dynamic code (JavaScript) is retrieved from a web server, gets executed in the browser and dynamically streams the topography, image and vector data from the server backends. Therefore we differentiate between the frontend and the backend of the system, with the stored web page on the server logically still belonging to the frontend side of the system, as it is interpreted on-the-fly and launched on the client (Fig. 1). The client code consists of the map canvas with user interface components, including attribute filter masks and data layer selection components. The selected data layers are requested directly from the map cache instance via concurrent OGC-based Web Map Service (WMS) and Web Feature Service (WFS) calls.

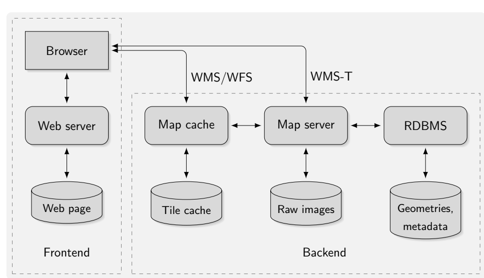

Figure 1: System design of the iMARS web-GIS outlining frontend (client) and backend (server) components.

Due to the vast number of single images we needed a way to set up the layers in a dynamic way since the map server does not know the images to be served beforehand. This requires an additional parameter associated with the PDS product ID (a unique identifier available for all NASA and ESA mission data products). In the OGC definition, such non-standard parameters providing enhanced capabilities can be realized as vendor-specific parameters (VSP) [2]. To provide dynamic layers, we introduce the `PRODUCTID` parameter in the WMS request to the map server. To ensure compatibility with regular WMS, the service will produce a valid result using a predefined default if the `PRODUCTID` parameter is omitted or malformed.

The dynamic service is implemented as a *MapScript* instance of MapServer. MapScript connects the Apache web server to the MapServer application programming interface (API) using CGI and provides an interface for the Python language. When used in the context of OGC services, it is capable of intercepting the request and using the request parameters for further processing. We use internal map template configurations with incomplete file paths for each available dynamic layer representing a camera instrument. With the product ID provided as an additional parameter, the internal path to the requested image on disk is constructed and the map is then finally configured. In this way, the individual layers are created on-the-fly, each representing only one single image. On the frontend side, the VSP request parameter (`PRODUCTID`) has to be appended to the regular set of WMS parameters. The pool of available product IDs is constructed from the requested attributes of the footprint coverage query on the frontend. Therefore, a coverage query always has to precede the single image query and visualization.

For external data queries and access, we have successfully extended the dynamic WMS services and implemented the Europlanet Data Model Table Access Protocol (EPN-TAP) [4] to provide Virtual Observatory (VO) access to the single HRSC "granules" via WMS [5]. As a first step, the single HRSC Level-3 images are available in Equidistant Cylindrical projection and can be queried by attribute keywords or geometry. Therefore the Python scripts have been extended to produce dynamic `GetCapabilities` output with available images as individual layers.

3. Frontend

The frontend is based on the *OpenLayers* JavaScript library. We have created special interface elements, such as the *Time panel* for temporal filtering and the ortho-image *Workflow toolbar* for visualizing single images. All available data produced from within the iMars project have been included together with other available data sets useful for providing additional context. Navigation is provided by an interactive pan-, zoom- and click environment common to all user interface elements placed on the actual map canvas (see Fig. 2).

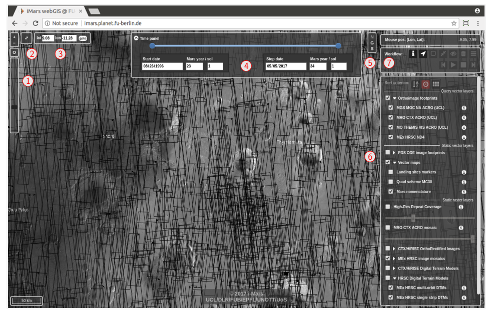

Figure 2: Browser view of the iMARS web-GIS with the user interface controls shown (numbered in red); 1: Zoom controls; 2: Fullscreen control; 3: Goto lat,lon; 4: Time panel; 5: Projection switcher; 6: Layer switcher; 7: Workflow toolbar.

Acknowledgements

This research has received funding from the EU's FP7 Programme under iMars 607379 and by DLR Bonn, grant 50QM1702. The project has made use of the USGS ISIS3. The software developed in the context of the iMARS web-GIS project is available at https://github.com/imars-fub.

References

[1] Muller, J.-P., et al. (2016), *ISPRS Archives*, *XLI-B4*, 453–458.

[2] Open Geospatial Consortium (2002), *Web Map Service Implementation Specification*.

[3] Hare, T. M., et al. (2018), *PSS 150*, 36–42.

[4] Erard, S., et al. (2014), *Astronomy and Computing 7-8*, 52–61.

[5] Minin, M., et al. (2016), *AGU 31*, p. 1759.