GEOGEN: A new approach and tool for computing the geometry metadata of ESA's PSA observational data products. N. Manaud[1], S. Besse[2], A. Montero[2], A. Escalante[2], R. Valles[2], I. Barbarisi[2], G. de Marchi[2], B. Merin[2], J. Gaspéri[3], and the PSA team[2], [1]SpaceFrog Design, Toulouse, France (nicolas@spacefrog.design), [2]ESA-ESAC, Madrid, Spain, [3]SnapPlanet, Toulouse, France

Introduction: The observational data products geometry metadata available within the ESA's Planetary Science Archive's (PSA) [1] datasets are typically computed and provided by the instrument teams. However, the level of completeness and consistency of these metadata may vary considerably across instruments and missions.

In order to enable PSA end-users with searching capabilities for all observational data products based on geometrical parameters in a consistent way, and to offer rich map-based visualisation and features, the PSA team, together with SpaceFrog Design, has implemented a common and independent approach and tool for computing observational geometry metadata.

System Overview: GEOGEN is a SPICE-based command-line application, written in Python [2], which essentially takes as input a JSON file (PLF) containing a list of PDS data products properties, and generates one or several output "coverage" GeoJSON files containing the geometry metadata associated to each valid input observational data product.

The Figure 1 illustrates the overall data flow, components and activities: from the PDS datasets ingestion process, to the generation of observational geometry metadata, to the end-users geometrical search and PSA UI interactions.

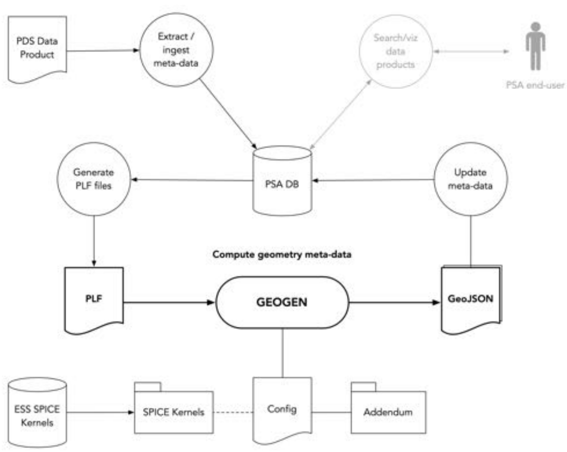

Figure 1: Overview of the PSA geometry generation end-to-end system using GEOGEN

PLF / data product properties - A minimum set of data product properties (metadata) is required for GEOGEN to compute the associated observation geometry metadata: product ID, UTC observation start and stop time, target name, instrument host ID, instrument ID and product type. Additional properties are required to identify the SPICE detector FOV and "simplified detector geometry model" applicable to a given data product. These properties are typically extracted from a PDS3 or PDS4 data product label in a previous step.

```
{
  "DATA_SET_ID": "MEX-M-OMEGA-2-EDR-FLIGHT",
  "PRODUCT_ID": "ORB1448_3_DATA",
  "INSTRUMENT_HOST_ID": "MEX",
  "INSTRUMENT_ID": "OMEGA",
  "TARGET_NAME": "MARS",
  "START_TIME": "2005-03-04T05:17:53.001",
  "STOP_TIME": "2005-03-04T05:25:38.003",
  "PRODUCT_TYPE": "EDR",
  "QUBE/CORE_ITEMS": "(128,352,580)"
}
```

Figure 2: Example of PLF data product properties.

Mission and instrument configuration - The computation of the input data products geometry metadata is done according to a user-defined configuration JSON file. This configuration file mainly allows for the definition of "computation-enabled" targets, instrument hosts, instruments, and product types; defining the required additional data product properties for a given instrument data product type. It links to applicable ESA's SPICE Kernels [3] and to "addendum" text kernels, which are used to define supplementary information required to interpret the applicable detector geometry model of a given input instrument data product.

Coverage GeoJSON - Output geometry metadata, including observational geometry parameters and footprint, are generated in the standard GeoJSON format [4] to ease the ingestion process into the PSA GIS system. Observational products within a "coverage" are grouped by target, mission, instrument, product type, and target surface model (ellipsoidal or digital shape model).

Key concepts:

Simplified Detector Geometry Model - We define and use 3 "detector types": point, line, and frame detectors. Point detectors are solely defined by a SPICE detector FOV boresight; they are typically used to model spectrometers. Line detectors are defined by a SPICE detector FOV boresight, an angle-of-view and rotation vector; they are used to model whiskbroom and pushbroom cameras. Frame detectors are used to model framing cameras.

Geometry Parameters - Geometry metadata includes a set of 51 geometrical parameters [5] that are intended to cover the interest of as many scientific disciplines as possible, and be relevant to all detector types. They have been defined by the PSA team and the PSA User Group.

Observation Footprint - Geometry metadata also includes the observation footprint planetocentric coordinates as GeoJSON Geometry. As Figure 3 illustrates, we define an observation footprint as the map-projected "observed-target" geometry. Using TAMN [6], footprints can be split when crossing the target body antimeridian and/or crossing the north or south pole. This enables correct observation footprint visualisation, and search results, in a simple cylindrical map-projected coordinate system; a significant improvement with respect to many similar GIS services.

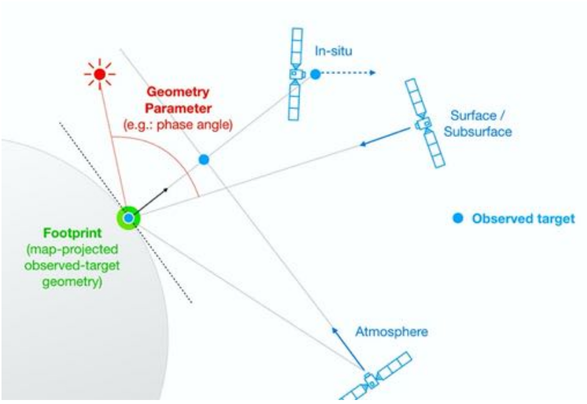

Figure 3: Illustration of the concept of observed-target and observation footprint, applicable to all instrument types.

Status and development: GEOGEN has been configured, tested and is being operationally used for Mars Express and Rosetta missions, respectively for the following instruments: ASPERA-3, HRSC, MARSIS, OMEGA, PFS, SPICAM; and OSIWAC, OSINAC, NAVCAM, ALICE, VIRTIS, MIRO. It enabled the first release of major new 2D/3D features of the PSA UI in December 2020 [7,8].

Development is on-going to improve the geospatial representation, and related geometry parameters, of "limb" observations (including target surface and off-surface pointing). Configuration and testing is on-going for Mars Express VMC and EM16 CaSSIS instruments. It is intended to enable geometry metadata computation on a systematic basis for all instruments of active and legacy missions of the PSA.

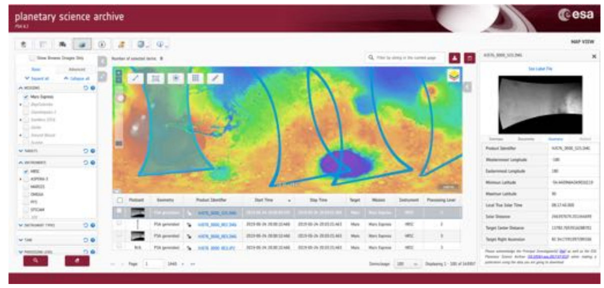

Figure 4: PSA UI v6.1 displaying Mars Express/ HRSC observations footprints and geometry parameters

References: [1] S. Besse et al. (2018), ESA's Planetary Science Archive: Preserve and present reliable scientific data sets, Planetary and Space Science, Volume 150, p. 131-140, [2] Annex et al., (2020). SpiceyPy: a Pythonic Wrapper for the SPICE Toolkit. Journal of Open Source Software, 5(46), 2050, https://doi.org/10.21105/joss.02050, [3] ESA SPICE Services: https://www.cosmos.esa.int/web/spice, spice@sciops.esa.int, [4] Butler, et al., The GeoJSON Specification (RFC 7946), https://tools.ietf.org/html/rfc7946, [5] PSA Geometry Parameters, http://bit.ly/psa_geoparams, [6] J. Gaspéri (2019), The AntiMeridian Nightmare (TAMN) software, https://snapplanet.io, [7] I. Barbarisi et al. (2021), PSA 2020: Toward The Discovery of ESA Planetary Data through 2D and 3D Interfaces (this conference), [8] PSA UI, https://psa.esa.int/psa.esa.int.