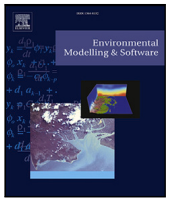
Environmental Modelling and Software 162 (2023) 105656

Contents lists available at ScienceDirect

Environmental Modelling and Software

journal homepage: www.elsevier.com/locate/envsoft



Discrete Global Grid Systems with quadrangular cells as reference frameworks for the current generation of Earth observation data cubes

Rubén Béjar *, Javier Lacasta, Francisco J. Lopez-Pellicer, Javier Nogueras-Iso

Aragon Institute of Engineering Research (I3A), Universidad Zaragoza, Spain

ARTICLE INFO

Keywords:
Discrete Global Grid System
DGGS
Earth Observation
Open Data Cube
rHEALPix

ABSTRACT

Discrete Global Grid Systems are spatial reference frameworks that associate information to multi-resolution grids of uniquely identified cells; they are proposed as mechanisms to facilitate the efficient integration of heterogeneous spatial data. They could provide an excellent reference system for Earth observation data cubes, technological infrastructures that provide analysis-ready access to Earth Observation big data, as long as they can be made compatible with them.

In this paper, we demonstrate that this is currently feasible without requiring new technological developments. We show how a Discrete Global Grid System with quadrangular cells, rHEALPix, and an existing data cube platform, Open Data Cube, can be integrated without loosing the advantages of having all the data in a Discrete Global Grid System, while keeping a straightforward access to all of the analysis tools provided by an Earth Observation Data Cube.

1. Introduction

The development of Earth Observation data cubes (EO data cubes) has been fueled by the increasingly pressing necessity to monitor the global environment. There is a growing volume, variety and velocity of big Earth observation data due to an increasing amount of Earth observation satellites equipped with instruments which have higher and higher spatial and spectral resolutions. These spatio-temporal data are made available faster and faster to the environmental research community through the implementation of better automatic processing pipelines.

In order to cope with this increasing amount of information, the EO data cubes have been proposed as a new paradigm where the access to big Earth observation data is facilitated by an infrastructure, the data cube, which provides analysis-ready spatio-temporal data to the data scientists, who can focus on their research and not on the technical issues of managing and efficiently accessing to those big data (Giuliani et al., 2019).

Discrete Global Grid Systems (DGGSs) are a relatively new framework for spatial referencing, focused on associating information to well-known and well-identified areas, cells, on Earth. Their main focus, and perhaps their strongest advantage, is also in the integration of heterogeneous spatial big data.

EO data cubes can be understood as a new technological paradigm. Their development has been focused on facilitating an efficient access to big Earth observation data, while allowing the use of existing, well-known tools for the processing of the data. The research presented in this paper addresses the problem of extending the technological capabilities of existing EO data cubes by considering other areas of improvement beyond data access performance and integration of common tools. More specifically, we demonstrate an original way to integrate DGGSs with EO data cubes.

This integration has been proposed before, but in a way that would require major technological developments. We show that this integration is feasible with the current generation of data cubes, without new technological developments, in the particular case of quadrangular DGGSs. If DGGS datasets are pre-processed within certain constraints, ingesting and processing those datasets in an existing data cube can be done without losing the characteristics of a DGGS, such as the well-known, uniquely identified and equal-area cells associated to some information.

Among the existing DGGSs, rHEALPix is a good choice for this work as its projected planar square grid is a good fit for the array-oriented tools that data cubes provide and for the raster tools that most GIS packages have. We will demonstrate that you can take an existing raster dataset, pre-process it producing another raster dataset that most common GIS applications and current data cubes can ingest and process, and that this new raster dataset is strictly under the framework provided by the DGGS: each pixel in the raster dataset corresponds

* Corresponding author.
E-mail address: rbejar@unizar.es (R. Béjar).

https://doi.org/10.1016/j.envsoft.2023.105656
Received 16 September 2022; Received in revised form 3 February 2023; Accepted 8 February 2023
Available online 15 February 2023

exactly with one cell in the DGGS, and matching pixels to cells does not require reprojecting or resampling.

In order to prove these points, we have carried out an experiment. First of all, we have transformed some datasets, both raster and vector, to rHEALPix rasters, being careful to preserve their DGGS characteristics. Then we have configured an instance of Open Data Cube loading it with both the original versions of these datasets and the ones transformed to rHEALPix. And finally, we have designed and implemented a simple geoprocess that combines those datasets in two different ways: one using the original datasets and the other one using the rHEALPix versions.

We have compared the results quantifying their differences to show that both versions of the workflow can be implemented with the same geoprocessing tools, that the results are quantitatively very similar, and that the data projected to rHEALPix does not loose its DGGS characteristics after this processing. This proves that, at least for some problems and with the appropriate procedures, some DGGSs can be used with the current generation of EO data cubes without requiring any major changes to existing software systems. We have found some issues that need to be addressed before this is immediately applicable at a large scale, but they are small and definitely fixable.

The code used to run these experiments is available in a GitHub repository: https://github.com/IAAA-Lab/rhealpix-opendatacube-docker.

In the following Section, we review some related work, focused on DGGSs, Open Data Cube, rHEALPix, and how data cubes and DGGSs are being currently combined. In Section 3 we describe the method and tools used to carry out our experiment: the datasets used as inputs, the transformation of those datasets to rHEALPix, the indexing of those datasets in Open Data Cube, and the geoprocess that calculates some results using them. In Section 4 we analyze the results produced by the geoprocess, both using a common spatial reference system and using rHEALPix, in order to quantify the differences. Section 5 discusses some of the decisions we have taken, and the main issues which have arisen, as well as some proposals for addressing or improving some of them. To finish this paper, Section 6 summarizes the main results and proposes some research lines for future work.

2. Related work

Discrete Global Grid Systems (DGGSs) divide the surface of the Earth into tessellations of cells, organized hierarchically in multi-resolution grids (Sahr et al., 2003). These grids are designed to contain and process information associated to those cells, which are fixed areas, and not as systems to support repeatable navigation (Purss, 2017). Every DGGS must also provide an algorithm to generate a unique identification, an index, for each of its cells.

Among the advantages of Discrete Global Grids over traditional GIS projections, we can point out that there are not any singularities, neither at the Poles nor elsewhere, that the spatial resolution of data is always made explicit and that their multi-resolution nature makes them good candidates for combining datasets with different spatial resolutions (Goodchild, 2018). An additional advantage is that equal area DGGSs allow to carry on spatial analyses which can be replicated consistently anywhere on Earth (Purss, 2017), even in higher latitudes where equal size pixels would create distortions for large areas (Hojati et al., 2022).

Regarding its potential to integrate heterogeneous big data, mentioned in Section 1, that would be facilitated by their always congruent multi-resolution cells and the hierarchical indexing schemes of those cells (Goodchild, 2018). We are already starting to see interesting cases of this in the case of big environmental data (Robertson et al., 2020), and indeed this kind of data is being proposed as one of the main drivers that justify the necessity for DGGSs (Hojati et al., 2022).

The rHEALPix is a cubic geodesic DGGS, compatible with the OGC proposal (Gibb, 2016), with cells that are squares once they are projected. This choice, square cells, makes it a better, at least simpler, fit for some problems than other DGGS which are based on hexagonal or triangular cells. For example, the integration of existing gridded datasets to hexagons may require different sampling and aggregation strategies for different resolutions of the DGGS (Bousquin, 2021). There are other advantages of quadrangular cells too, such as the perfect congruency between adjacent levels of the DGGS, and the widespread use of some data structures which match them perfectly (Amiri et al., 2015). On the other hand, quadrilateral cells, such as those in rHEALPix, would be less adequate to model dynamical systems where inter-cell distances are involved, as they lack uniform adjacency (Bowater and Stefanakis, 2018). In our paper, we have focused on a problem that uses existing gridded datasets, requires area calculations and does not use inter-cell distances at all, which makes it a good case for rHEALPix.

The rHEALPix DGGS has other positive attributes, such as low average angular and linear distortions, and a perfectly congruent cell structure where every cell at one resolution level is fully contained into a cell at the previous resolution level, which is something that hexagon-based grids do not support. There is a Python library that implements the rHEALPix DGGS (Gibb et al., 2013a), and its projection is supported by the well-known PROJ library (PROJ contributors, 2022), which makes it automatically available in many GIS packages and applications. The support provided by the PROJ library has allowed us to use common GIS libraries such as Rasterio (Gillies et al., 2013) more easily in our work.

Earth observation satellites and improved scientific instruments are delivering growing amounts of better quality Earth observation data, and the need to transform that increasing volume of data in information in a timely fashion has encouraged the development of data processing infrastructures, in many cases under the paradigm of the spatial data cubes where the Australian Geoscience Data Cube is one of its main examples (Lewis et al., 2017). This project is where the Open Data Cube open software project (Killough, 2018) was born, a software that we are using in this paper. The Open Data Cube intends to provide a scalable, open and free tool to exploit satellite data, and is already been used to support data cubes in Switzerland (Chatenoux et al., 2021), several African countries (Mubea et al., 2020) and other regions. Data cubes in general, with different technologies and implementations, are getting more and more attention as platforms for the efficient modeling and analysis of grid coverages that model multi-dimensional, spatio-temporal data (Baumann, 2021).

Regarding the relationship among data cubes and DGGSs, there are some advantages in using a DGGS as a base to implement a data cube infrastructure, as its “data integration engine layer”, and there are a number of initiatives where this combination is being explored (Purss et al., 2019). How that is related to the results in this paper is discussed in Section 5.

3. A workflow to process data in open data cube using rHEALPix

This section describes the method and tools that we have used to validate the hypothesis of this paper, which is the feasibility of processing rHEALPix-based datasets under a data cube paradigm, with standard geoprocessing tools, while keeping the advantages provided by the use of this DGGS and while producing results which are accurate when compared to the ones produced with more common reference systems. This method is a geodata processing workflow which uses datasets loaded into an instance of Open Data Cube, based on the cube-in-a-box Docker container (Open Data Cube, 2022), and is implemented using a Jupyter notebook (Kluyver et al., 2016). This workflow produces its results in two different ways: using the source datasets in their native reference systems and resolutions, and using versions of those datasets previously reprojected and resampled to rHEALPix. The workflow solves a simple geospatial processing task, because its purpose as a validation tool for the hypothesis mentioned above does not require it to be more complex.

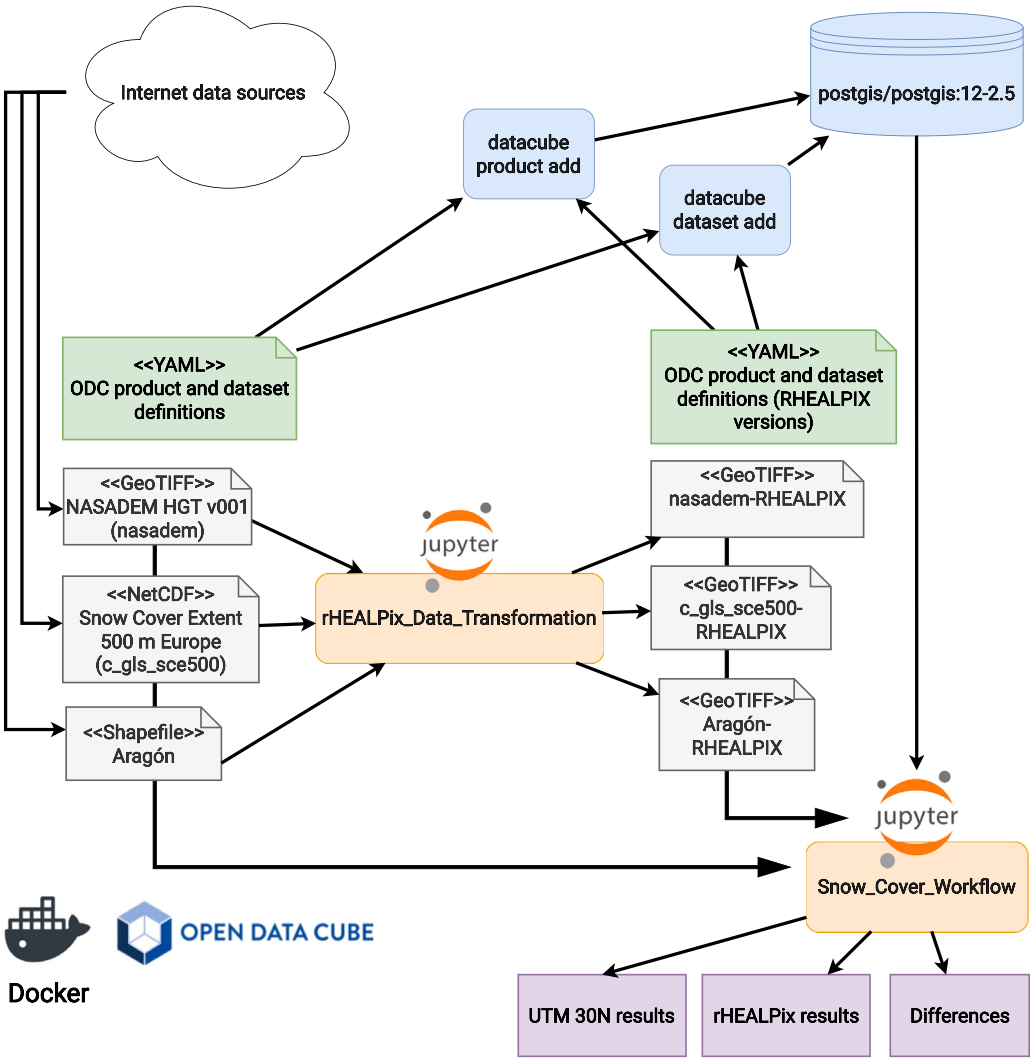


Fig. 1. System architecture and workflow design.

Both alternatives of the workflow calculate the same results:

- The average snow cover extension in a given date, April 26, 2022, in two given areas: the Pyrenees, and the part of the Pyrenees that belong to the Aragón region in Spain (a NUTS level 2 territorial unit);
- The total surface, for each of these study areas, with a snow cover of 75% or more.

In a more realistic context these, or similar, results would be calculated and accumulated in a daily basis and used in some decision-making process or analysis. For example, snow cover is a variable used in forage production simulation (He et al., 2019) or in catchment nutrient models in cold areas (Costa et al., 2020).

To delimit the study areas we have used a bounding box of the Pyrenees, and kept only those parts above 1500 m in altitude. The Aragón boundary, as a polygon, has also been necessary.

This workflow is designed as an experiment to test not only the feasibility of using rHEALPix with standard data cube and geoprocessing tools, but also the accuracy of the produced results, so we have also made a quantitative comparison of the results, described in Section 4. To keep the workflow simple and focused on testing our problem, we are making our calculations with data for a single date, but with the same tools that would be used for processing time series or data cubes with more dimensions.

Fig. 1 provides a graphical view of this workflow. In this figure we see that the original datasets, downloaded from their online sources, are transformed to rHEALPix using a Jupyter notebook. Then, all datasets, the original ones and the rHEALPix versions, plus their product and dataset metadata, YAML files required by Open Data Cube which have been created by hand for this example, are indexed in the Open Data Cube. Finally, there is another Jupyter notebook which implements the rest of the workflow, produces the intended results with both versions of the datasets and calculates the differences among those results. All this is available in a GitHub repository, https://github.com/IAAA-Lab/rhealpix-opendatacube-docker, implemented using a Docker container to facilitate its deployment and testing in different environments.

In the following subsections we will describe the steps of the workflow roughly following the pattern suggested by Apicella et al. (2022): data selection & download, pre-processing, processing, data integration and results presentation.

3.1. Data selection, download and transformation to rHEALPix

The workflow combines datasets extracted from three products: the NASADEM HGT v001 (NASA et al., 2000), the Snow Cover Extent 500 m Europe (Copernicus Service Information, 2022), and the boundary of Aragón, derived from the Administrative Units of Spain (Instituto Geográfico Nacional (ign.es), 2021). These datasets, and the YAML

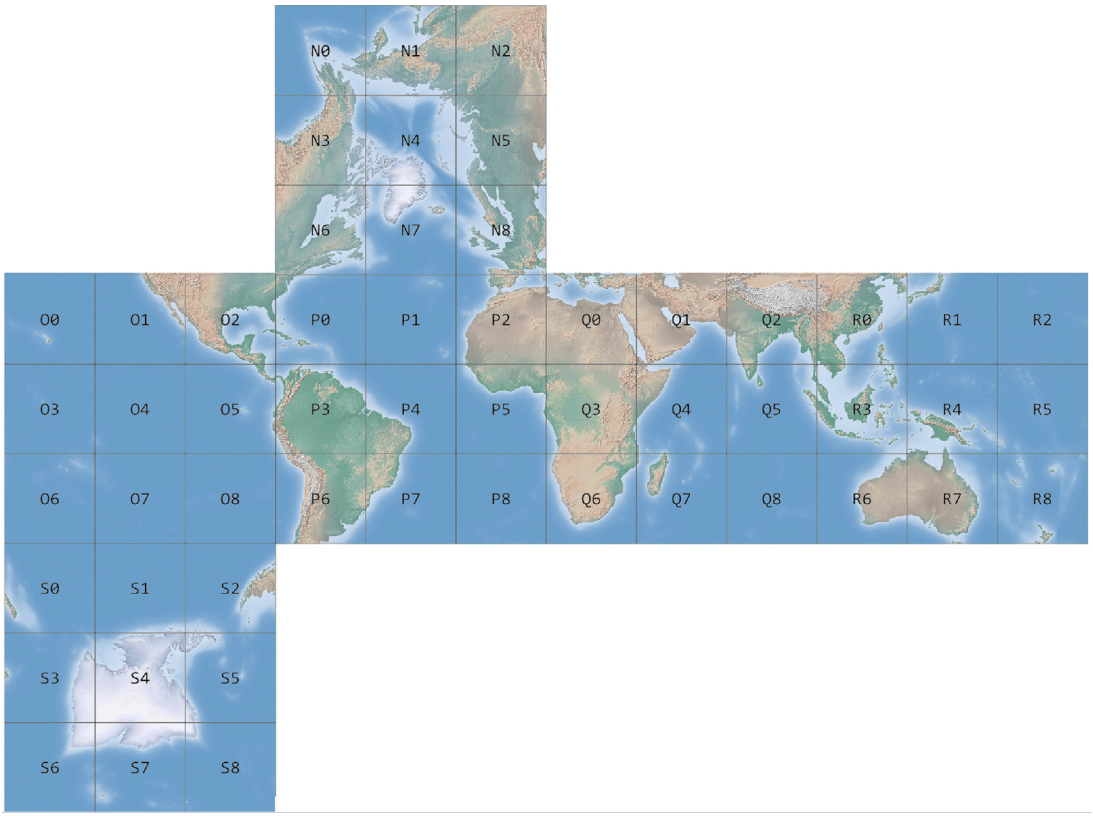


Fig. 2. (1,0)-rHEALPix resolution 1 cells with $N_{side} = 3$ and prime meridian at 10°.

metadata associated to them and their product definitions, are indexed into the Open Data Cube using its command line tools.

The rHEALPix system supports a number of parameters that can be used to produce different geodesic grids. We have used the WGS84 ellipsoid and made $N_{side} = 3$, so each cell is subdivided into 9 at each new resolution level, and we have positioned the north and south squares at positions 1 and 0: (1,0)-rHEALpix. These parameters are not very significant given our problem and study area, so other choices could have been made.

Another parameter is more relevant, and that is the planar origin. This can be shifted if, for instance, a study area is divided between two faces of the rHEALPix cube and you prefer it contained in just one. Although software which is aware of the rHEALPix system should be able to deal with data distributed in several faces of the cube, we have found some issues with this (see Section 5 for more details) so we have had to use 10° as our prime meridian. Fig. 2 shows the resolution 1 cells for the (1,0)-rHEALPix with $N_{side} = 3$ and prime meridian at 10°.

The choice of $N_{side} = 3$ establishes the available resolutions, and the cell sizes at those resolutions. We have chosen to work at the resolution level which is closest to the one that we will be using for resampling the original datasets, 500 m, and this turns out to be level 9, for a cell width, and height, of 508.72 m.

Finally, we have decided that resampling is made with the nearest neighbor strategy, for the rHEALPix transformation, and everywhere else in this paper. Although other resampling strategies could have produced slightly better results, we wanted to focus on comparing rHEALPix vs non-rHEALPix, and thus we have kept other parameters and choices as simple as possible.

To transform the raster datasets to rHEALPix we have used the Rasterio (Gillies et al., 2013) library to warp, i.e., reproject and resample, the datasets from their original coordinate reference systems to rHEALPix, and the rHEALPixDGGS library (Gibb et al., 2013a) to calculate the size of the cells to use the proper pixel size. Before the actual warping, we have made the transformation matrix perfectly aligned with the rHEALPix grid, so each pixel center corresponds with an rHEALPix cell centroid.

For the transformation of the polygon vector dataset, Aragón boundaries, we have first rasterized its features using Rasterio, in its original reference system, and then we have followed the same process as with the other raster datasets.

The Python code used for these transformations is in a notebook named `rHEALPix_Data_Transformation`. Although the rHEALPix datasets are already included in the GitHub repository, so they can be immediately indexed without using this notebook at all, it can be run to verify that the produced rHEALPix datasets are identical to the included ones.

We could have included this in the data pre-processing step, but as this is done in a separate notebook, we think that it is clearer to consider it an additional task within the data download step.

3.2. Pre-processing, processing, data integration and results presentation

Once the datasets are indexed in the Open Data Cube, the rest of the workflow is implemented in the `Snow_Cover_Workflow` notebook which has the following steps:

- Pre-processing: in the part that uses the original datasets, we establish a common reference system and a common resolution to work with, and then reproject and resample as needed. Given the datasets, area of study and objectives, we have chosen to use the Universal Transverse Mercator coordinate system, zone 30N and with the ETRS89 datum (EPSG:25830). For the common resolution, we have chosen the highest resolution of the datasets we are using, which is 500 m. We also have to create a raster mask from the Aragón vector dataset, so we can use it in the processing step.
  Pre-processing using the rHEALPix datasets is simpler, given that some steps have already implemented when loading the datasets

Table 1
Snow cover values.

Study area	Reference system	Area ≥ 75%	Mean	Std dev	Cohen's d
Pyrenees					
	UTM 30N	1269.75 km^2	28.86%	43.82%	
	rHEALPix	1253.90 km^2	28.27%	43.55%	
					−0.013
Aragón Pyrenees					
	UTM 30N	33.25 km^2	10.88%	30.72%	
	rHEALPix	32.51 km^2	11.77%	31.85%	
					0.029

into the Open Data Cube (reprojection and resampling). We do not even have to do anything special regarding the Aragón geometry, as at this point it is just another rHEALPix raster dataset available in our Open Data Cube that can immediately be used as mask.

- Processing, data integration and results presentation: these steps are simple, as the calculations themselves are not complex. We mask the snow cover dataset to keep only the cells inside our study areas, calculate the values we are looking for in our study with those masked datasets, using the tools provided by Open Data Cube, the Xarray library (Hoyer et al., 2022) and other Python geoprocessing libraries, and make a simple visualization of the results.

The comparison of the results using the original datasets and the rHEALPix ones is also included in the notebook, and described in the next section.

4. Results

We have compared the results from the two variants of this workflow, with and without using a DGGS, and quantified the differences between them. There are three sources of systematic errors that would explain these differences: clipping raster data with a vector geometry (the Aragón boundary), resampling, and reprojecting. These processes are common in geodata processing workflows, so we were not expecting large differences due to them. Our main focus is to make sure that rHEALPix and the software libraries that implement it do not introduce large systematic errors in our results, because it is not a commonly used reference system, and also to test the code that we have implemented for processing the datasets.

We summarize the differences in two tables. In Table 1 we see that the data we are calculating in this study, total surface with a snow cover of 75% or more, and the mean and standard deviation of the snow cover percentage in the given study areas, are very similar using both approaches. Although in absolute numbers it seems clear that the difference between the means is small, we have also calculated a standardized difference between them, the Cohen's d, to confirm that point.

Fig. 3 shows the areas on the Pyrenees with a snow cover of 75% or more as a raster dataset, at the top, and as a vector dataset at the bottom. Both datasets are in rHEALPix, though they have been projected to Mercator just to show them on a map. The vector dataset has been produced by taking the raster dataset we have created with the Open Data Cube tools: reading each pixel, translating the coordinates of that pixel to a resolution 9 cell of the (1,0)-rHEALPix, with $N_{side} = 3$, and writing the centroid of that cell, with its snow cover and the unique identifier of the cell, to a vector file. A sample of the contents of the vector dataset is shown in Table 2. In the provided notebook there is code to produce both datasets, the raster and the vector ones, as a GeoTIFF and a GeoPackage respectively, so that they can be fully explored and compared with any desktop GIS application.

Table 2
Pyrenees. Cells with a snow cover of 75% o more, with their identifier in the (1,0)-rHEALPix, $N_{side} = 3$, DGGS.

Row	Cell id	Geometry	Snow cover
0	N878818335	POINT (−1 166 244.489 5 095 865.234)	100.0
1	N878818343	POINT (−1 165 735.768 5 095 865.234)	100.0
…	…	…	…
5705	N886764527	POINT (−962 247.089 5 028 205.249)	85.0
5706	N886764528	POINT (−961 738.367 5 028 205.249)	100.0
5707	N886765306	POINT (−961 229.646 5 028 205.249)	99.0

Table 3
MAE, BIAS and correlation between the rHEALPix datasets, reprojected to UTM 30N, and the results obtained working directly in UTM 30N for both the Pyrenees and the Aragón Pyrenees.

Study area	Mean absolute error	BIAS	Spearman's ρ
Pyrenees	2.66	−0.40	0.95
Aragón Pyrenees	0.35	0.07	0.99

The point we want to highlight here is that we have been working with normal raster datasets, with the standard tools provided by Open Data Cube, but the DGGS nature of the data has not been lost because the input datasets were carefully reprojected, resampled and aligned before indexing them in the Open Data Cube. And thus, in our raster results, each pixel corresponds exactly with a DGGS cell, and producing a vector version of those cells, that can be processed with any vector GIS application, is simple and direct, in the sense that it does not require any reprojection, resampling or any other operation that could introduce additional systematic errors.

Table 3 shows the results of a different validation of the results. We have taken the resulting datasets from the two different approaches, UTM 30N and rHEALPix, and we have compared them on a pixel by pixel basis, calculating the Mean Absolute Error (MAE), the BIAS and their correlation. In these datasets the pixels with NODATA appear as NaN (not a number), so we have made our calculations taking this into account:

- To make the datasets comparable pixel by pixel we have reprojected and resampled, with the nearest neighbor strategy, the dataset resulting from the rHEALPix process to the CRS and resolution of the dataset produced with the other geoprocess.
- To calculate the mean absolute error (MAE), we have subtracted the value of the pixels in one dataset from the value of the corresponding pixel in the other one, taken the absolute value of this difference and obtained a NaN-aware mean.
- The calculation of the BIAS is similar, but without taking the absolute value.
- We have calculated a correlation coefficient, the Spearman's ρ, between both datasets.

Finally, Fig. 4 plots the absolute difference between both datasets on a map, for the whole Pyrenees, to show the spatial distribution of the differences.

5. Discussion

The work by Purss et al. (2019) points out that data cubes "can coexist with (and benefit from) an underlying DGGS-compliant data tiling and integration scheme". In this paper, we are contributing to prove that point by actually integrating an existing data cube with a DGGS. However, Purss et al. (2019) also point out that this integration would normally require to replace the tiling and query processing in the data cubes by DGGS/DGGS-like technology, and also that "a DGGSs structure [...] does not need to employ spatial analytics operations to perform routine search, aggregation, and decomposition tasks.". We fully agree on the second point: with the proper DGGS structure and

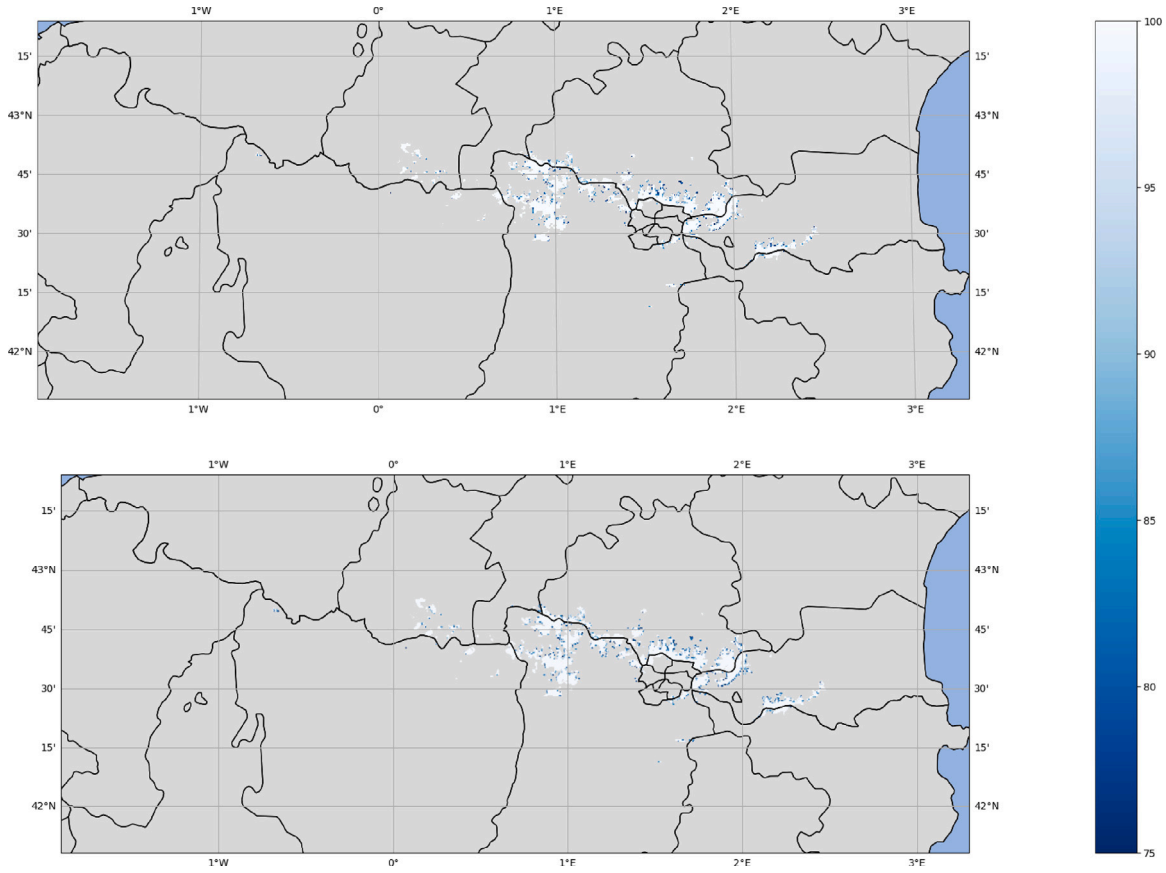


Fig. 3. Pyrenees. Areas with a snow cover of 75% or more, as a raster dataset (top), and as a vector dataset with the rHEALPix cells (bottom). Mollweide projection.

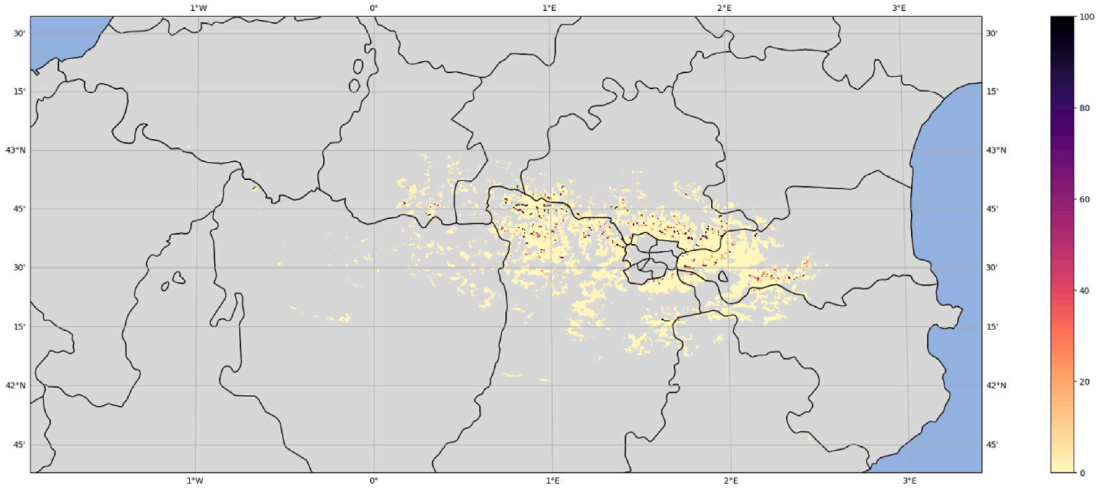


Fig. 4. Pyrenees. Absolute difference in snow cover between the UTM 30N and the rHEALPix datasets. Mollweide projection.

query tools you do not need spatial analytics operations. Nevertheless, even if you do not need them, you can still use them: our solution does use spatial analytics operations, those provided by the Open Data Cube, and this makes the integration possible without new tiling and query processing technology.

There are many spatial analytics/raster data processing tools and libraries which expect or, at least, are very optimized for the case of regular grids with rectangular, usually square, cells. Discrete global grids based on hexagonal or triangular cells do not make easy cases for these tools and libraries, but those global grids which are based on quadrangular cells are a different case: they could offer the opportunity to make use of those tools and libraries with little, if any, changes. The results presented in this paper do that, and thus they would confirm that this opportunity is real. If other research confirms this for other use cases, and also for other DGGSs and tools, we can have ahead of us a path for the adoption of DGGSs in different raster data processing communities: these communities could continue working with their current tools while having the possibility to easily make profit from some advantages of the DGGSs as they introduce them in their work. This path could also lead to increase the adoption of DGGSs based on hexagonal and triangular cells: as the quadrangular DGGSs become better known, and more used, the interest in the applications where

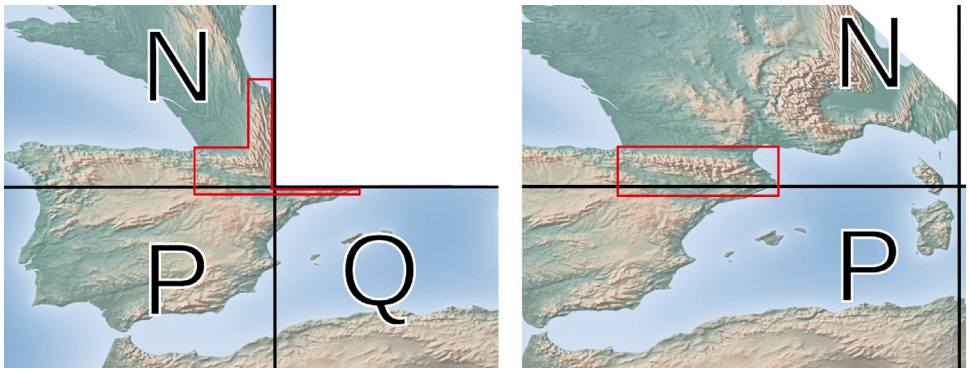


Fig. 5. Pyrenees study area (in red). Projected to rHEALPix with prime meridian 0° (left), and with prime meridian 10° (right).

hexagonal and triangular DGGSs might be better options could be increased. Finally, this path could also facilitate the transition towards data cube implementations which are more DGGS-aware as proposed by Purss et al. (2019).

Nevertheless, there are some relatively minor issues that could make it more difficult to follow the path proposed in the previous paragraph. Although we have shown that we can calculate results which are very close to those obtained with common projections using the same tools, we have found a number of problems, specific to rHEALPix and Open Data Cube, that have required solutions and workarounds that deserve some discussion.

First of all, rHEALPix accepts a number of parameters that modify both the grids and the projection of the data. As briefly mentioned in Section 3.1, we have shifted the prime meridian to 10°, so our area of study falls in adjacent faces of the rHEALPix cube. This can be seen in Fig. 5, where the area of study is approximately depicted in red: with the default prime meridian, on the left, the study area intersects the N, P and Q faces and, worse, although it is a bounding box that can be expressed as two pairs of geodetic coordinates, once projected to the plane with rHEALPix, it is not a rectangle anymore because this area intersects non-contiguous faces. On the right side we can see that with the prime meridian at 10° the study area is contiguous and a rectangle after projecting to the plane. This should not be a major issue for a software which is developed to deal with rHEALPix datasets, but the `load` operation of Open Data Cube failed with the default prime meridian. Shifting the prime meridian solved the problem in our case, but a long term solution would require a more robust rHEALPix data selection by extent in Open Data Cube.

This brings another issue to the table: the PROJ software library supports shifting the prime meridian for rHEALPix projections, with the `lon_0` parameter, but not the prime parallel. In our case, shifting the prime meridian has been enough to be able to work with Open Data Cube and Rasterio as required, but the capability of shifting the planar origin in both axes could be necessary, or at least convenient.

Besides this minor thing, the support provided by PROJ to the rHEALPix projection has allowed us to use common GIS software, such as Rasterio, easily in this work. However, with a DGGS, the projection is not everything, you need to take into account at least the allowed resolutions and cell identifiers, so the rHEALPixDGGS library was also necessary. For this paper we have combined both libraries, plus Fiona (Gillies et al., 2011–) to deal with vector geometries, to reproject and resample the used datasets. This code, besides supporting our experiment, can be useful for others working with this DGGS, but it lacks the completeness, robustness and documentation that a library to reproject datasets to rHEALPix should have. We propose this as future work in Section 6.

Our rHEALPix datasets are produced as rasters, in the GeoTIFF format, before indexing them into Open Data Cube and processing them with the array-oriented tools provided there. Nevertheless, as described in Section 4, implemented in the Jupyter notebook and shown in Table 2, they can be easily transformed into a vector dataset where each pixel corresponds to a single cell in the DGGS, without requiring any reprojection or resampling, and without losing any information at all (neither spatial, nor thematic). As a collection of unique cell identifiers where each one corresponds to one fixed area on Earth with some associated variables (e.g., snow cover value), the results produced within the data cube can be immediately exported to a CSV file, processed as a DataFrame, loaded into a SQL or Non-SQL database, with or without spatial capabilities, etc. For the small example shown in this paper, this result proves, mainly, that DGGSs and data cubes can be made compatible, at least for some DGGSs, right now. However, in a big data context where this is done in a consistent way for many datasets, with many different workflows, all of them updated often, and with many other kinds of datasets, spatial and non-spatial, available, this becomes much more interesting, because it underlines the major expected advantage of using DGGSs: they facilitate the integration of heterogeneous spatial big data. In Section 6, we propose as future work the design of spatial data processing pipelines that integrate data cubes, SQL and Non-SQL databases, big data frameworks and other data storage and processing infrastructures, all of them mediated by DGGGs. We have already seen some steps towards this kind of infrastructures (Robertson et al., 2020), and we expect that incorporating data cubes in the way we propose here can provide further advances.

The validation of the proposal presented in this paper is done with a specific variable, snow cover, and comparing rHEALPix with a UTM projection on a specific study area. There is not anything special about any of those choices. A DGGS defines a spatial structure with uniquely identified well-known cells, but the contents of those cells can be any kind of information. And the DGGSs are focused on the information associated to the cells, so designing them with small area distortions is important (an "information grid not a navigation grid" as succinctly put in Purss (2017)). UTM has been chosen because it is a common choice for the chosen study area, and the rHEALPix is a global projection with a small areal distortion, see Gibb et al. (2013b, Appendix B), so it should perform properly in any other study area.

To end this section, there are two technical decisions of our work that deserve some discussion. First, as described in Section 4, when we have compared cell by cell the spatial datasets resulting from working in UTM 30N and in rHEALPix, we have calculated the Spearman's ρ, instead of the perhaps more common Pearson correlation coefficient (Pearson's r). This has been done because the Spearman's ρ makes less assumptions on the distribution of the data: it only requires a monotonic relationship, so it is often less biased than the Pearson's r. The monotonicity condition has been checked with a scatterplot of the data, which is included in the Jupyter notebook.

And second, to index a product, and its datasets, in Open Data Cube, there are certain metadata files, in the YAML format, which are required. For this paper, these files have been created manually as we just needed to show that it was feasible to index the rHEALPix datasets in Open Data Cube. In a more realistic scenario, these files should

be produced automatically at the same time that the reprojection and resampling of the original datasets is done, something that could also be done by that library that we are proposing as future work.

6. Conclusion and future work

In this paper we have demonstrated that a DGGS based on quadrilateral cells, rHEALPix, can be used to process raster datasets with current data cube technology, in this case Open Data Cube, without any changes to that software and with the usual, array-oriented tools which are commonly used in data cubes and producing results which are very similar to those produced using other projection systems. A careful pre-processing of the raster datasets, taking into account the characteristics of rHEALPix, allows us to process the raster datasets as if they were projected with any common projection system, without losing their DGGS-based structure. This way, we can produce, easily and without losing any spatial or thematic information, the final results as values associated to uniquely identified cells.

In larger information infrastructures, these uniquely identified cells can be associated to values produced in other, completely unrelated spatial processes, helping thus to create a view of the Earth where all kind of information is associated to a congruent, multi-resolution grid of cells, each corresponding a well-known, fixed area on the Earth surface, and thus getting closer to that Digital Earth vision that DGGSs intend to facilitate.

We have discussed some issues that we have found in our work. The rHEALPix DGGS has a number of characteristics which are still not completely supported by current spatial software libraries. We have shown that some workarounds have been necessary, and that we have needed to combine a number of different libraries to achieve the intended results. Although some improvements in the management of rHEALPix, and surely other DGGSs too, are required in common spatial libraries, we think that a robust, generic implementation of some of the code developed for the experiments in this paper, well-documented and easy to install, would be a useful addition to the toolbox of any spatial data analyst willing to work with rHEALPix.

The experiment conducted in this paper suggests that a deeper integration of rHEALPix and the Open Data Cube is feasible without too much work. The indexing of rHEALPix products and datasets in an instance of the ODC just requires a proper configuration regarding the coordinate reference system definition and the valid cell resolutions. And regarding the Python API that allows the processing of the datasets, a first step is using it carefully, for instance avoiding resampling operations if they would produce cell resolutions outside of the allowed resolutions of the DGGS, and including external libraries for rHEALPix specific operations when needed; this is what we have done in this experiment. A next step, that we propose as future work, would be to design and implement a set of rHEALPix-aware, and rHEALPix-safe, operations and integrate them in the ODC Python API.

This paper has presented a small example intended to prove the feasibility of our approach and to quantify potential systematic errors. However, most benefit from DGGSs will come when they are used to facilitate integrating heterogeneous big spatial data. Introducing DGGS processing steps in big data processing pipelines, and finding out how to make those steps as efficient and automatic as possible, is necessary to validate this point if we expect DGGSs to develop their full potential. We intend to work on this problem in our next steps.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Code and data created and/or used are available in a GitHub repository, referenced in the text.

Acknowledgments

This paper is part of the R&D project PID2020-113353RB-I00, supported by the Spanish Ministry of Science and Innovation (MCIN/AEI/10.13039/501100011033/) and the project T59_20R supported by the Aragon Regional Government.

We also want to acknowledge the suggestions provided by the reviewers and the editors of the journal, as they have undoubtedly helped us to improve this paper.

References

Amiri, A.M., Samavati, F., Peterson, P., 2015. Categorization and conversions for indexing methods of discrete global grid systems. ISPRS Int. J. Geo-Inf. 4 (1), 320–336. http://dx.doi.org/10.3390/ijgi4010320, URL: https://www.mdpi.com/2220-9964/4/1/320.

Apicella, L., De Martino, M., Quarati, A., 2022. Copernicus user uptake: From data to applications. ISPRS Int. J. Geo-Inf. 11 (2), http://dx.doi.org/10.3390/ijgi11020121, URL: https://www.mdpi.com/2220-9964/11/2/121.

Baumann, P., 2021. A general conceptual framework for multi-dimensional spatio-temporal data sets. Environ. Model. Softw. 143, 105096. http://dx.doi.org/10.1016/j.envsoft.2021.105096, URL: https://www.sciencedirect.com/science/article/pii/S1364815221001390.

Bousquin, J., 2021. Discrete Global Grid Systems as scalable geospatial frameworks for characterizing coastal environments. Environ. Model. Softw. 146, 105210. http://dx.doi.org/10.1016/j.envsoft.2021.105210, URL: https://www.sciencedirect.com/science/article/pii/S1364815221002528.

Bowater, D., Stefanakis, E., 2018. The rHEALPix Discrete Global Grid System: considerations for Canada. Geomatica 72 (1), 27–37. http://dx.doi.org/10.1139/geomat-2018-0008.

Chatenoux, B., Richard, J.-P., Small, D., Roeoesli, C., Wingate, V., Poussin, C., Rodila, D., Peduzzi, P., Steinmeier, C., Ginzler, C., et al., 2021. The Swiss data cube, analysis ready data archive using earth observations of Switzerland. Sci. Data 8 (1), 1–11.

Copernicus Service Information, 2022. Snow cover extent 500 m Europe. https://land.copernicus.eu/global/products/sce.

Costa, D., Baulch, H., Elliott, J., Pomeroy, J., Wheater, H., 2020. Modelling nutrient dynamics in cold agricultural catchments: A review. Environ. Model. Softw. 124, 104586. http://dx.doi.org/10.1016/j.envsoft.2019.104586, URL: https://www.sciencedirect.com/science/article/pii/S1364815219306474.

Gibb, R.G., 2016. The rHEALPix discrete global grid system. In: IOP Conference Series: Earth and Environmental Science. In: IOP Conference Series: Earth and Environmental Science, vol. 34, 012012. http://dx.doi.org/10.1088/1755-1315/34/1/012012.

Gibb, R., Car, N.J., Raichev, A., 2013a. rHEALPixDGGS: a Python package that implements the rHEALPix Discrete Global Grid System (DGGS). https://github.com/manaakiwhenua/rhealpixdggs-py.

Gibb, R., Raichev, A., Speth, M., 2013b. The rHEALPix Discrete Global Grid System. Technical Report, Landcare Research NZ Ltd., URL: http://dx.doi.org/10.7931/J2D21VHM.

Gillies, S., et al., 2011–. Fiona is OGR's neat, nimble, no-nonsense API. URL: https://github.com/Toblerity/Fiona.

Gillies, S., et al., 2013. Rasterio: geospatial raster I/O for Python programmers. https://github.com/rasterio/rasterio.

Giuliani, G., Camara, G., Killough, B., Minchin, S., 2019. Earth observation open science: Enhancing reproducible science using data cubes. Data 4 (4), http://dx.doi.org/10.3390/data4040147, URL: https://www.mdpi.com/2306-5729/4/4/147.

Goodchild, M.F., 2018. Reimagining the history of GIS. Ann. GIS 24 (1), 1–8. http://dx.doi.org/10.1080/19475683.2018.1424737.

He, W., Grant, B., Smith, W., VanderZaag, A., Piquette, S., Qian, B., Jing, Q., Rennie, T., Bélanger, G., Jégo, G., Deen, B., 2019. Assessing alfalfa production under historical and future climate in eastern Canada: DNDC model development and application. Environ. Model. Softw. 122, 104540. http://dx.doi.org/10.1016/j.envsoft.2019.104540, URL: https://www.sciencedirect.com/science/article/pii/S1364815218305747.

Hojati, M., Robertson, C., Roberts, S., Chaudhuri, C., 2022. GIScience research challenges for realizing discrete global grid systems as a Digital Earth. Big Earth Data (published online), 1–22. http://dx.doi.org/10.1080/20964471.2021.2012912.

Hoyer, S., Roos, M., Joseph, H., Magin, J., Cherian, D., Fitzgerald, C., Hauser, M., Fujii, K., Maussion, F., Imperiale, G., Clark, S., Kleeman, A., Nicholas, T., Kluyver, T., Westling, J., Munroe, J., Amici, A., Barghini, A., Banihirwe, A., Bell, R., Hatfield-Dodds, Z., Abernathey, R., Bovy, B., Omotani, J., Mühlbauer, K., Roszko, M.K., Wolfram, P.J., 2022. Xarray. http://dx.doi.org/10.5281/zenodo.598201.

Instituto Geográfico Nacional (ign.es), 2021. Límites municipales, provinciales y autonómicos. CC-BY 4.0 https://centrodedescargas.cnig.es/CentroDescargas/busquedaSerie.do?codSerie=LILIM.

Killough, B., 2018. Overview of the Open Data Cube initiative. In: IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 8629–8632.

Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Avila, D., Abdalla, S., Willing, C., 2016. Jupyter Notebooks – a publishing format for reproducible computational workflows. In: Loizides, F., Schmidt, B. (Eds.), Positioning and Power in Academic Publishing: Players, Agents and Agendas. IOS Press, pp. 87–90.

Lewis, A., Oliver, S., Lymburner, L., Evans, B., Wyborn, L., Mueller, N., Raevksi, G., Hooke, J., Woodcock, R., Sixsmith, J., Wu, W., Tan, P., Li, F., Killough, B., Minchin, S., Roberts, D., Ayers, D., Bala, B., Dwyer, J., Dekker, A., Dhu, T., Hicks, A., Ip, A., Purss, M., Richards, C., Sagar, S., Trenham, C., Wang, P., Wang, L.-W., 2017. The Australian Geoscience Data Cube — Foundations and lessons learned. Remote Sens. Environ. 202, 276–292. http://dx.doi.org/10.1016/j.rse.2017.03.015, URL: https://www.sciencedirect.com/science/article/pii/S0034425717301086, Big Remotely Sensed Data: tools, applications and experiences.

Mubea, K., Killough, B., Seidu, O., Kimani, J., Mugambi, B., Kamara, S., 2020. Africa regional data cube (ARDC) is helping countries in Africa report on the Sustainable Development Goals (SDGS). In: IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium. pp. 3379–3382. http://dx.doi.org/10.1109/IGARSS39084.2020.9324156.

NASA, JPL, USGS, OpenTopography, Microsoft, 2000. NASADEM HGT v001. https://planetarycomputer.microsoft.com/dataset/nasadem.

Open Data Cube, 2022. Cube in a box. https://github.com/opendatacube/cube-in-a-box.

PROJ contributors, 2022. PROJ coordinate transformation software library. http://dx.doi.org/10.5281/zenodo.5884394, https://proj.org/.

Purss, M. (Ed.), 2017. The OpenGIS Abstract Specification - Topic 21: Discrete Global Grid Systems Abstract Specification, OGC 15-104r5. Open Geospatial Consortium.

Purss, M.B., Peterson, P.R., Strobl, P., Dow, C., Sabeur, Z.A., Gibb, R.G., Ben, J., 2019. Datacubes: A discrete global grid systems perspective. Cartogr.: Int. J. Geogr. Inf. Geovis. 54 (1), 63–71.

Robertson, C., Chaudhuri, C., Hojati, M., Roberts, S.A., 2020. An integrated environmental analytics system (IDEAS) based on a DGGS. ISPRS J. Photogramm. Remote Sens. 162, 214–228.

Sahr, K., White, D., Kimerling, A.J., 2003. Geodesic discrete global grid systems. Cartogr. Geogr. Inf. Sci. 30 (2), 121–134. http://dx.doi.org/10.1559/152304003100011090.