# Discussion of *Earth system data cubes unravel global multivariate dynamics*, by Miguel Mahecha et al.

Edzer Pebesma*, Marius Appel*

Nov 12, 2019

We congratulate the authors of this paper with a very nice contribution describing the ESDC and its application, an activity that has involved a substantial conceptual development as well as implementation work, that has triggered a number of scientific papers already, and that now helps furthering the discussion what spatiotemporal data cubes are. We are mostly interested in having this latter discussion.

## 1 Pre-grid all the data, or do this on-the-fly?

The ESDC pre-grids all the data, and (l456) "One of the most commonly expressed practical concerns is the choice of a unique data grid". Given that an ESDC is defined on a single grid, but integrates a lot of different datasets, each of these datasets have to be forged into the target grid. This involves resampling, statistical downscaling, interpolation and/or aggregation of data both in space and time, as well as handling coordinate reference systems and possibly different calendars (Gregorian, 365day/noleap, 360 day) into one. It is unclear whether the ESDC can do all this, and also if it can give an idea about the errors introduced by doing so, or whether it can give users otherwise recommendations what a good grid is?

Other systems, e.g. Google Earth Engine or Sentinel Hub work from the raw data (which can be in many different coordinate reference systems, e.g. Sentinel-2 distributed over 120 UTM zones), and create user-defined data cubes on the fly. This has the advantage that low-resolution computations can be done relatively fast and interactively, which helps data exploration and model development, and that the effect of different target resolutions can be easily evaluated. With the ESDC, once data are cubed, users no longer have the possibility to go back to the original data.

We think this issue is important, and missed a discussion on this matter in the paper. Given that large, operational systems exist that do not store pre-

*Institute for Geoinformatics, University of Münster, Germany

cubed datasets but that work with data cube *views* (Pebesma et al., 2019) we believe that this may be at least a viable option.

## 2 Is latitude the same as time?

In Line 598, the paper states that "The ESDL is probably the most radical data cubing approach", and refers to some grey literature that also claims that data cubes should treat all dimensions identically, irrespective their semantics. We believe however that space and time are inherently different, and need different treatment. The example (e.g. fig 3) shows that all longitudes are aggregated to give profiles per latitude and time, but we feel that this is a rather contrived example; in general, any transect in space could be a good candidate for reducing space to one dimension, and this would require mapping onto that transect; it is not so likely that you would want to do that on an arbitrary transect in the two-dimensional space defined by e.g. (longitude, time). In general, a large number of functions applied to space operate on both spatial dimensions (e.g., polygonal crop), and time series models are of a very different kind and are rarely applied to single spatial dimensions. The fact that you *can* do this is nice, but we do not believe it more of a big selling point than an opportunity for users to shoot themselves in the foot.

## 3 Is a lat/long grid the only way we can cube the Earth?

No, it isn't, and many datasets use other global grids, discrete global grids, or collections of grids (Equi7, or UTM zones). All this has reasons, and the Earth will never be flat so the problem will remain. A discussion on generalizing the ESDC to other grids, or collections of these, would be welcome.

## 4 Vector data cubes are missing

In the representation of the ESDC, the implicit assumption is being made that data cubes correspond to spatial raster data. This is not the case: space can be a one-dimensional set of feature geometries (e.g. points, or polygons). One of the most requested feature of data cubes is to retrieve all the cube information at a given set of points, or aggregated over a given set of polygons. This leads naturally to vector data cubes. Can ESDC answer such queries, or do the authors consider this to not be data cubes?

## 5 Where is support?

Spatial grids may refer to a collection of points on a regular grid, or to a collection of grid cells as if they were small square polygons. In the latter case,

properties can be either continuous over a grid cell (e.g., land use, soil type, geology) or may be aggregated values over the grid cell (e.g., the total amount of carbon in the grid cell, or its maximum elevation). Similarly can time be conceived as a set of time instances or intervals, with the two interval interpretations. Does the ESDC take care of some of these options, or is this all assumed to be remembered by the users?

# 6 Code and reproducibility

We are happy to learn that the software is open source, and can be used by others. Have the authors heard success stories of others installing and using the software? A few small code examples in the paper to get a taste for how simple queries to the ESDC look like would have worked well, and could be encourage those hungry for trying it out. We did not try out the Julia script but hope that other reviewers will report whether they did, and whether they were successful in reproducing the results shown in the paper.

# 7 Dropping dimensions

If a dimension has only one value, it is dropped; we can see this is useful, but it also drops the information of the dimension (e.g. the species name, or the elevation value). We see a lot of 4-dimensional NetCDF files in the wild having 1 dimension with one values, to specify the value for that dimension, which seems all but useless. You need to be able to drop it, but can the ESDC also not drop it, e.g. so that sub-cubes can be meaningfully combined along the otherwise dropped dimension?

# 8 The data model

Having a data model that maps from dimensions to $\mathbb{R}$ and NA is great; a similar approach was adopted in Appel and Pebesma (2019). For end users it also means there is no way to properly handle logical (TRUE/FALSE or NA), categorical, or e.g. time variables. Of course 8-byte doubles can encode anything, but everyone who has tried to use them for encoding categories knows the nightmare. Do end users need that? Some sort of a discussion on this issue would be welcome.

# 9 Other issues

1. Can the ESDC cope with irregular dimensions, e.g. irregularly distributed time steps?

2. lines 260-270: array databases by default store the data in a database, not in HDF5 or NetCDF, although some can be made to do so; they

may import data from HDF5 or NetCDF though. We believe the "Earth system scientists" mentioned in line 269 will soon be the old school if data cube access principles get more widely used beyond GEE and ESDC. No data scientist will want to go back to the individual files underlying a properly implemented data cube.

3. line 180: we think that spectral decomposition does not map from (time) to (time,freq), but to (freq). Eq (12) and (13) have the same problem.

4. Eq (15) para should be par?

5. line 500 ff: we believe that UDFs are quite widely spread and are implemented in SciDB, rasdaman commercial, openEO (GeoPySpark/GeoTrellis, Grass GIS), R package stars, and Python module xarray. Seeing an example of an ESDC UDF in the paper would be nice!

# References


- Marius Appel, Edzer Pebesma, 2019, On-Demand Processing of Data Cubes from Satellite Image Collections with the gdalcubes Library. Data 4(3), 92

- Edzer Pebesma, Wolfgang Wagner, Pierre Soille, Miha Kadunc, Noel Gorelick, Matthias Schramm, Jan Verbesselt, Johannes Reiche, Matthias Mohr, Jeroen Dries, Alexander Jacob, Markus Neteler, Soeren Gebbert, Christian Briese and Pieter Kempeneers, 2019. openEO analyses Earth Observation data based on user-defined raster and vector data cube views. Geophysical Research Abstracts Vol. 21, EGU2019-9737, 2019, EGU General Assembly 2019. abstract, poster.