ATLAS: A visualization and analysis framework for geospatial datasets

Ricardo Barros Lourenço[1,3]
Nathan Matteson[1,2]
Alison Brizius[1]
Joshua Elliott[1]
Ian Foster[1,3]

[1] Computation Institute, Argonne National Laboratory and University of Chicago
5735 South Ellis Avenue – 60637 – Chicago – IL, USA
{ joshuaelliott, abrizius }@uchicago.edu

[2] School of Design, College of Computing and Digital Media, DePaul University
243 South Wabash Avenue – 60604 – Chicago - IL, USA
nmatteso@cdm.depaul.edu

[3] Computer Science Department, University of Chicago
Ryerson Laboratory – 1100 East 58th Street – 60637 – Chicago - IL, USA
{ rlourenco, foster }@cs.uchicago.edu

Abstract. Due to the internationalization of agricultural markets and the relevance of global change drivers (climate, population, consumption, and regulation), food security and land-use change dynamics must be evaluated at the global scale. The effects of food insecurity and environmental impacts, however, are largely experienced locally and confronted by decision-makers at national or regional scales. The ATLAS (Agro-Economic Dynamics and Trade-offs of Land Use and Sustainability) viewer is designed to support data management, retrieval, analysis and visualization to enable users to explore interactions across these scales. We are beginning with visualizations of pSIMS model outputs and will be extending ATLAS for use with many more models and other multi-scale data products.

Keywords: data visualization, climate modelling, cloud computing, visualização de dados, modelagem climática, computação em nuvem

1. Introduction

The Center for Robust Decision-making on Climate and Energy Policy (RDCEP) is located at the Computation Institute, which is a joint venture of the Argonne National Laboratory and the University of Chicago. The Center brings together experts in economics, physical sciences, energy technologies, law, computational mathematics, statistics, and computer science to undertake a series of tightly connected research programs aimed at improving the computational models needed to evaluate climate and energy policies, and to make robust decisions based on outcomes.

RDCEP is funded by a grant from the National Science Foundation (NSF) through the Decision Making Under Uncertainty (DMUU) program as described in NSF (2009). The primary goal of DMUU is to "advance fundamental understanding of decision making under uncertainty for climate change and related long-term environmental risks". In addition, the Center is expected "to provide new knowledge about how public officials, firms in the private sector, other groups, and/or individuals can incorporate existing knowledge about climate change and related long-term environmental risks into their decisions".

As part of RDCEP activities, the center generates and consumes geospatial information in large scale, often dealing with datasets that have global coverage in high spatial and temporal

resolutions. Examples of those datasets include pSIMS climate impacts ensemble model outputs as defined in Elliott et al. (2014); climate forcing datasets such as the ones described in Ruane, Goldberg and Chryssanthacopoulos (2015); soil simulation model outputs as described in Shangguan et al. (2014); among others. These datasets are calculated in High Performance Computing (HPC) environments, and the output of those models is stored in the Network Common Data Form (NetCDF).

The NetCDF container was initially defined by Rew and Davis (1990), and has become a popular standard for representing climate data, in part because it was designed as a multidimensional data structure able to be manipulated in parallel. The raster data is represented as sets of multidimensional arrays of primitive types, which are assigned variable names, physical dimensions, and other possible attributes. The most paramount aspect of NetCDF files is the ability to use dynamic dimensions, generating datasets that have uneven spatial distribution of data, while avoiding memory allocation of spaces that don't contain data, thus reducing the file size.

Despite its robustness as a geospatial data container, the NetCDF format is not the main standard for the majority of Geographical Information Systems (GIS) users. These users often prefer using the ESRI Shapefile format, which is proprietary and closely related with ESRI ArcGIS platform. Integrating NetCDF files into the research workflow does require a practical knowledge of the format as well as some programming skills—attributes that are not always present in the the users of such data.

Having these computational characteristics, and also the typical user group of the climate products developed at RDCEP (such as public policy stakeholders, economists, teachers, and a wide range of students) taken in account, ATLAS was initially developed to be a user friendly visualization interface for data stored in NetCDF files. The scope of this framework is to provide thematic maps and a limited scope of analytical tools, allowing its users to retrieve a variety of datasets and analyze their patterns—thus facilitating the integration of such data into other research with minimal effort, and without getting into the myriad of jargon and technical aspects present in the GIS learning process.

2. Methodology

The development of ATLAS is conceptually related with the research conducted at RDCEP, and this project was initially designed to graphically represent three different datasets that are in the core of its research. The datasets differ much in terms of their content, and due to this the internal data organization is different too. The primary effort in terms of project methodology was to define a common platform able to integrate such varied data without loosing information; without restricting the users' ability in retrieve information; and in a reasonably efficient manner with regard to information storage. To fully define our methodology it is necessary to define upfront the main datasets and just then the technologies able to deal with this data variety.

2.1. Dataset Description

We describe the GeoJSON container and MongoDB NoSQL database components of our dataset representations.

2.1.1. pSIMS Model Run

The pSIMS model run, as defined in Elliott et al. (2014), and using a different setup defined in Elliott et al. (2015), generated multiple crop model ensembles as NetCDF Files containing

three main variables[1], being the spatial pairs latitude and longitude in a global one-degree grid in both dimensions; and time, being split in annual intervals from 1948 to 2012. Each file has a single attribute but with global coverage, being a result of a permutation of parameters of:

- Climate models;
- Historical climate forcing datasets;
- Simulation Scenarios in terms of simulation configuration and irrigation setting;
- Mandatory attributes (Crop yields; Applied irrigation water);
- Optional attributes (Total above-ground biomass yield; Actual growing season evapotranspiration; Actual planting date; Days from planting to anthesis; Days from planting to maturity; Nitrogen application rate; Nitrogen leached; Nitrous oxide emissions; Accumulated precipitation, from planting to harvest; Growing season incoming solar; Sum of daily mean temperature, from planting to harvest);
- Crop type;
- Time step and time range.

2.1.2. AgMERRA Model Run

The AgMERRA model run, as defined in Ruane, Goldberg and Chryssanthacopoulos (2015), generated climate forcing data stored in NetCDF files where the three variable names are: latitude and longitude in a global quarter-degree grid in both dimensions; and time defined in daily values ranging between 1980 and 2010. With these three variables used as input references, there are other attributes defined, such as: Precipitation flux; Maximum and minimum surface air temperature; Surface downwelling shortwave radiation flux; Wind speed at 10 meters; Relative humidity average approximated by average temperature; Relative humidity at time of maximum temperature; and cropland percentage. These attributes are dynamic. They are distributed across several files which are split in terms of spatial coverage and depend on the three initial variables as primary keys in a database.

2.1.3. GSDE Model Run

The GSDE model run, as defined in Shangguan et al. (2014), is a high resolution soil model. It is distributed in two sets of information:

1. The first set has a spatial grid resolution of 30 seconds, with eight depths of soil simulation as main variables. The related attributes are: Additional property; Available water capacity; Drainage class; Impermeable layer; Nonsoil class; Phase1; Phase2; Reference soil depth; Obstacle to roots; Soil water regime; Topsoil texture.
2. The second set has spatial grid resolution of five minutes, with eight depths of soil simulation split in two different files, with the top four depths separated from the bottom four. For these pairs of files, the following attributes are represented: Total carbon; Organic carbon; Total nitrogen; Total sulfur; $CaCO_3$; Gypsum; pH(H_2O); pH(KCl); pH($CaCl_2$); Electrical conductivity; Exchangeable calcium; Exchangeable

[1]In the context of this work, we use the definition established in Rew and Davis (1990), which names *attribute*, as a variable dependent on other independent variable(s).

magnesium; Exchangeable sodium; Exchangeable potassium; Exchangeable aluminum; Exchangeable acidity; Cation exchange capacity; Base saturation; Sand content; Silt content; Clay content; Gravel content; Bulk density; Volumetric water content at -10 kPa; Volumetric water content at -33 kPa; Volumetric water content at -1500 kPa; Amount of phosphorous using the Bray1 method; Amount of phosphorous by Olsen method; Phosphorous retention by New Zealand method; Amount of water soluble phosphorous; Amount of phosphorous by Mehlich method; Exchangeable sodium percentage; Total phosphorus; Total potassium.

The files are just split in terms of attributes, being much larger than the AgMERRA or pSIMS NetCDF files.

2.2. Data Transformation and Analysis

2.3. Web Interface

The initial prototype of ATLAS was created in 2014. Its focus was on accessing individual NetCDF files, using decompression of those files on-the-fly. The prototype worked on a small subset of the pSIMS model runs, without needing to integrate other datasets. During that period, a web interface was developed using the spatial capabilities of the d3 Javascript library as defined at Bostock, Ogievetsky and Heer (2011) communicating with a python backend able to open and manipulate this subset of NetCDF files. This interface provided a visualization schema that allowed interaction with GeoJSON messages, which were generated by directly transforming a NetCDF into a GeoJSON on demand.

However, this model has been proven insufficient to deal with a wider range of data, especially because for every user query, the target file would be opened and transient GeoJSON messages generated. Once a user closes his browser, this transient information is lost, and a new request requires re-processing at the back-end level. With multiple users accessing the website, this would generate a processing overhead. Considering these issues, one straightforward solution is to use a geospatial database. Another option would be to use an entire WebGIS service, but to maintain interactivity with each 'pixel' of the data, we use d3 to produce SVG vector graphics from GeoJSON messages, instead of the sort of tile server typically found in traditional WebGIS implementations.

2.4. Back-end

2.4.1. GeoJSON Container

In our application, we transform the input NetCDF file into multiple GeoJSON messages, as defined in Butler et al. (2016). This is done by discretizing each point present in the input NetCDF file, generating an output GeoJSON message with a polygon, often a squaroid (considering that all input data is encoded using WGS84 standard), for each spatial coordinate. Data for each coordinate is represented as a n-dimensional values key in the GeoJSON's properties object, the dimensionality being dependent on the characteristics of the ingested dataset. When visualizing the data, these polygons are rendered by d3 as multiple contiguous polygons. For example, Figure 1 shows a pSIMS register, in which is possible to verify how a GeoJSON point is adapted for the ATLAS environment.

2.4.2. MongoDB NoSQL Database

Another issue arose when evaluating the original prototype, around the choice of a database. The most used database in the open source GIS world is Postgres with the PostGIS extension.

```
{
    "_id" : ObjectId("56649b87a54d75221dd3ac45"),
    "geometry" : {
        "type" : "Polygon",
        "coordinates" : [
            [
                [-124.0,45.0],[-123.5,45.0],[-123.5,44.5],[-124.0,44.5],[-124.0,45.0]
            ]
        ]
    },
    "type" : "Feature",
    "properties" : {
        "source" : "papsim_wfdei.cru_hist_default_firr_aet_whe_annual_1979_2012.nc4",
        "centroid" : {
            "geometry" : {
                "type" : "Point",
                "coordinates" : [
                    -123.75,
                    44.75
                ]
            }
        },
        "value" : {
            "start" : [
                NumberInt(1979)
            ],
            "step" : NumberInt(1),
            "values" : [298.6,292.7,311.6,305.6,291.1,346.5,323.4,337.3,328.2,310.8,298.1,316.1,326.0,289.5,332.5,
                302.6,281.7,310.7,272.9,320.7,271.6,370.3,303.9,286.7,295.7,298.2,307.7,307.6,309.9,321.0,273.3,
                290.7,314.2,null]
        },
        "timestamp" : "2015-12-06T14:33:11.030204",
        "simulation" : "aet_whe"
    }
}
```


Figure 1: GeoJSON register as stored in a MongoDB collection in ATLAS

However due to the data volume of this project, which uncompressed is in the order of hundreds of terabytes, and the Postgres's current lack of support for operating as a parallel and distributed service, we looked for other options. By 2015, MongoDB already supported geospatial indexing and querying; it also has the ability to use shards of nodes in order to scale out computational power in a cluster using cloud computing.

Complementing the predefined GeoJSON message defined in the previous section, we use MongoDB to store these messages in collections that are generated for each model run inserted. Chodorow (2013) defines MongoDB as a NoSQL document database, that stores registers in JSON nested structures, called collections, which are single message binaries encoded as BSONs. It is possible to perform a wide range of spatial queries on such collections, provided they use spatial indexing. Currently in our work we use MongoDB's '2dsphere' indexing, because of our data is compliant to WGS84. Currently, the framework is under active development. A proper evaluation of databases, and general back-end infrastructure will be an objective of future work.

When a user initially opens the ATLAS website, they are presented with a list of available datasets. Upon selecting the desired dataset, the backend collects the browser's viewport information, and by an inverse process using d3, it is possible to define a bounding box from the user's viewport. That information is encoded to a spatial query, also using a bounding box, that retrieves all the GeoJSON available for that region, within a certain context as shown in Figure 2. As the GeoJSON is converted to SVG in the client, the array of values contained in the key 'values' is bound to each 'pixel.' This provides for a range slider in the UI allowing the user to 'move' the visualization along other dimensions (e.g., time or depth) with almost no latency. Because the data is not compressed, this generates almost no overhead at the user side.

3. Results and Discussion

ATLAS users are able to retrieve data and assess its basic characteristics, as one would in traditional thematic cartography. Users can choose from divergent (Figure 2) or sequential

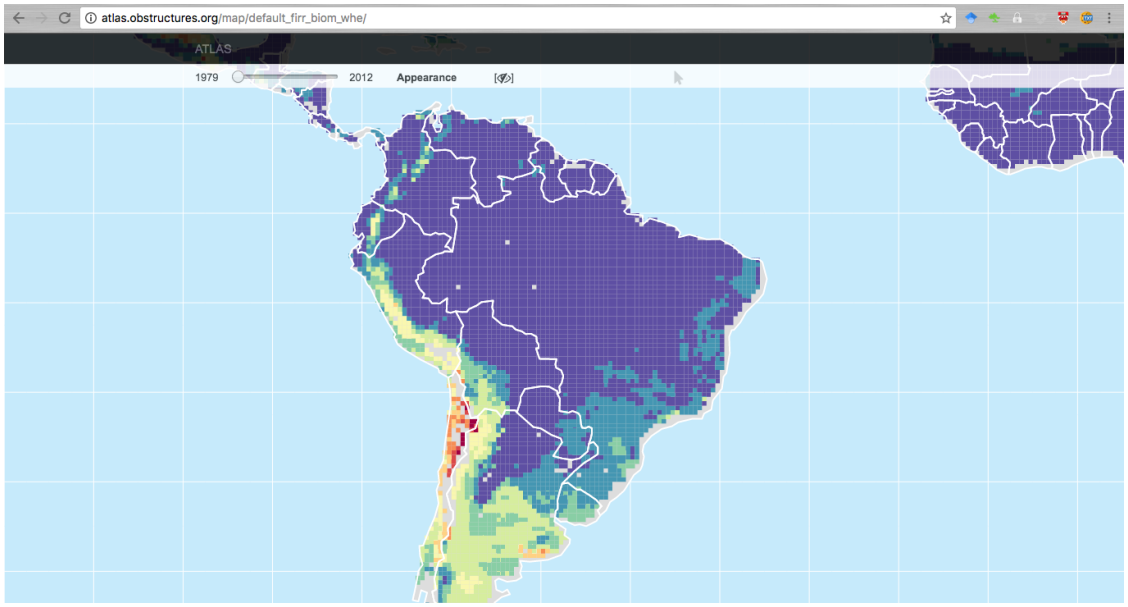

Figure 2: Example of landing page for a pSIMS Wheat Biomass under Irrigated conditions. The sliding bar allows the user dynamically look into a time range between 1979 and 2012

(Figure 3) color ramps, and adjust the number of color bins (Figure 3). ATLAS also enables users to chose a preferred smoothing filter as shown in Figure 4. Notice that the smoothing is applied to the visualization, not the source data. All these operations are performed on the fly, with latency levels that do not exceed three seconds.

4. Conclusions

The ATLAS framework is indeed a work still in development. However, it allows the diverse community of RDCEP users to easily visualize geospatial information stored in NetCDF files, and users not experienced with geoprocessing tasks are able to perform their own simple data analysis, thereby avoiding confusion and delays. In this work the usage of a NoSQL database was also important in helping to integrate a variety of different simulation products, each with a unique data model, in the same environment, and dealing with intensive simultaneous processing workloads without losing speed.

Future work on this project will involve the development of statistical and geostatistical features, towards allowing the users to perform Exploratory Spatial Data Analysis (ESDA) in the ATLAS environment. Our intention is to incorporate some of the features available in the PySal library as defined in Rey and Anselin (2010). Further, we plan to define a more efficient compression schema, as well working on an encoding method able to perform spatial similarity analysis, including dimensionality reduction methods discussed in Samet (2006), such as Single Value Decomposition (SVD) and Fast Fourier Transform, but also Fast Wavelet Transform, which is able to represent non-harmonic signals, often present in environmental datasets.

5. Acknowledgments

This work was funded by NSF Decision Making Under Uncertainty Program under award N$^{o.}$ 0951576. Ricardo Barros Lourenço also acknowledges the financial support of CAPES Foundation/Ministry of Education of Brazil under grant 88888.075449/2013-00, between September, 2014 and June, 2016.

Figure 3: Example of landing page for a pSIMS Wheat Biomass under Irrigated conditions. The color ramp was changed from divergent to sequential, being also the user able to customize the number of color bins

References

BOSTOCK, M.; OGIEVETSKY, V.; HEER, J. D3 data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, IEEE Educational Activities Department, Piscataway, NJ, USA, v. 17, n. 12, p. 2301–2309, dez. 2011. ISSN 1077-2626. Available from Internet: <http://dx.doi.org/10.1109/TVCG.2011.185>.

BUTLER, H. et al. *The GeoJSON Format*. [S.l.], 2016. Available from Internet: <https://tools.ietf.org/html/rfc7946>.

CHODOROW, K. *MongoDB: the definitive guide*. [S.l.]: " O'Reilly Media, Inc.", 2013.

ELLIOTT, J. et al. The parallel system for integrating impact models and sectors (psims). *Environmental Modelling & Software*, Elsevier, v. 62, p. 509–516, 2014.

ELLIOTT, J. et al. The global gridded crop model intercomparison: data and modeling protocols for phase 1 (v1. 0). *Geoscientific Model Development*, Copernicus GmbH, v. 8, n. 2, p. 261–277, 2015.

NATIONAL SCIENCE FOUNDATION. *Decision Making Under Uncertainty Collaborative Groups (DMUU)*. Arlington, VA, jul. 2009. Available from Internet: <https://as102.http.sasm3.net/pubs/2009/nsf09544/nsf09544.htm>.

REW, R.; DAVIS, G. Netcdf: an interface for scientific data access. *IEEE computer graphics and applications*, IEEE, v. 10, n. 4, p. 76–82, 1990.

REY, S. J.; ANSELIN, L. Pysal: A python library of spatial analytical methods. In: *Handbook of applied spatial analysis*. [S.l.]: Springer, 2010. p. 175–193.

RUANE, A. C.; GOLDBERG, R.; CHRYSSANTHACOPOULOS, J. Climate forcing datasets for agricultural modeling: Merged products for gap-filling and historical climate series estimation. *Agricultural and Forest Meteorology*, Elsevier, v. 200, p. 233–248, 2015.

SAMET, H. *Foundations of multidimensional and metric data structures*. [S.l.]: Morgan Kaufmann, 2006.

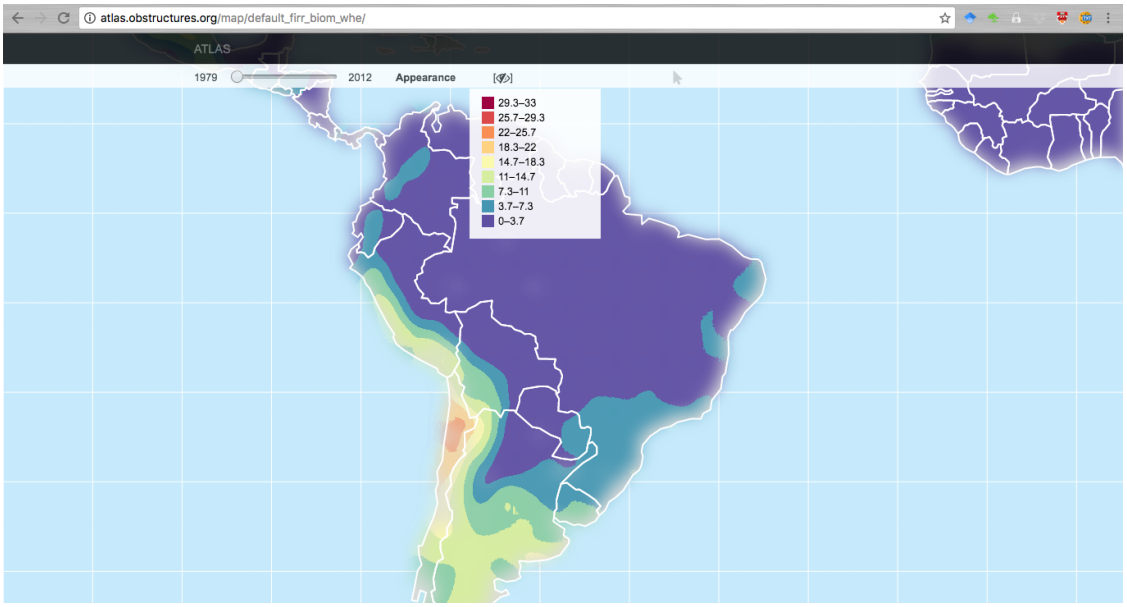

Figure 4: Example of landing page for a pSIMS Wheat Biomass under Irrigated conditions. The cell edges were smoothed using an SVG gaussian blur available in D3 library

SHANGGUAN, W. et al. A global soil data set for earth system modeling. *Journal of Advances in Modeling Earth Systems*, Wiley Online Library, v. 6, n. 1, p. 249–263, 2014.