HHS Public Access

Author manuscript

J Geogr Syst. Author manuscript; available in PMC 2019 October 01.

Published in final edited form as:

J Geogr Syst. 2018 October ; 20(4): 343–361. doi:10.1007/s10109-018-0277-2.

IPUMS-Terra: Integrated Big Heterogeneous Spatio-Temporal Data Analysis System

David Haynes, Ph.D.,

Family Medicine, University of Minnesota, 717 Delaware Street SE, Suite #166, Minneapolis, MN 55455, dahaynes@umn.edu

Alex Jokela, and

Institute for Social Research and Data Innovation, University of Minnesota

Steven Manson

Department of Geography, Environment, and Society, University of Minnesota

Abstract

Big Geo Data promises tremendous benefits to the GIS Science community in particular and the broader scientific community in general, but has been primarily of use to the relatively small body of GI-Scientists who possess the specialized knowledge and methods necessary for working with this class of data. Much of the greater scientific community is not equipped with the expert knowledge and techniques necessary to fully take advantage of the promise of big spatial data. IPUMS-Terra provides integrated spatiotemporal data to these scholars by simplifying access to thousands of raster and vector datasets, integrating them and providing them in formats that are useable to a broad array of research disciplines. IPUMS-Terra exemplifies a new class of National Spatial Data Infrastructure because it connects a large spatial data repository to advanced computational resources, allowing users to access the needle of information they need from the haystack of big spatial data. The project is trailblazing in its commitment to the open sharing of spatial data and spatial tool development, including describing its architecture, process development workflows, and openly sharing its products for the use general use of the scientific community.

Keywords

IPUMS-Terra; spatial-temporal data; spatio-temporal analysis; spatial data infrastructure

1 Introduction

IPUMS-Terra is a spatial data infrastructure project that focuses on the integration of heterogeneous big data. It does so by developing new methods and technologies capable of handling the diversity, volume, and complexity of scientific spatial data in the era of big data. IPUMS-Terra focuses on creating an integrated data repository with population and environmental data by collecting, preserving, documenting, and harmonizing datasets for a diverse community of researchers.

The term big data became popular in the early 2000's (Laney, 2001). However, geospatial datasets have always been large, requiring additional computational strategies to support their analysis (Armstrong, 2000; Ding & Densham, 1996). What separates big data from large data are the additional foundational characteristics termed the v's of big data: volume, velocity, and variety (Laney, 2001). With respect to the IPUMS-Terra application and spatial data integration more generally, we would offer two additional characteristics: value and veracity. The IPUMS-Terra's big data repositories reflect these characteristics.

The IPUMS-Terra data collection is big data, as it draws on allied datasets from the Institute of Social Research and Data Innovation (ISRDI), including the Integrated Public Use Microdata Series *International* projects (IPUMS-*I*) and the National Historical Geographic Information System (IPUMS-NHGIS). IPUMS-*I* contains over 300 censuses with half a billion person records with a total data volume of approximately three terabytes (Minnesota Population Center, 2017). NHGIS data collection is approximately 400 gigabytes, containing decennial census data from 1790 to present describing housing, population, and economic characteristics from the United States (Minnesota Population Center, 2011). Additionally, the IPUMS-Terra's raster data collection is approximately half a terabyte and provides users access to over 4,000 global rasters on agriculture, climate, and land cover datasets.

IPUMS-Terra's integration across three data structures: microdata, vector data, and raster data is truly innovative, particularly in how it handles and processes data types with varying characteristics. Microdata are fixed-width files on individuals and households, typically derived from surveys collected by government agencies. Vector data are geographic boundaries of areas, such as census units or administrative regions. Raster data are gridded and referenced representations of aerial imagery and satellite sensor data of the earth's surface. They have typically been translated into earth features such as elevation or land cover.

IPUMS-Terra deals to varying extents with the 'v's of big data. However, it currently does not consume and process data in real time, velocity. Historically, each of the IPUMS data projects has been developed and maintained as separate data repositories. A motivation for the development of IPUMS-Terra was to connect these repositories and data structures to promote heterogeneous big data integration. The velocity component of IPUMS-Terra is slow, but it is constantly in motion.

Validity and value are paramount characteristics of big data in IPUMS-Terra. ISRDI focuses on providing curated datasets that promote compatibility and comparability for researchers. The center disseminates approximately one terabyte of data each month to thousands of researchers, resulting in hundreds of publications each year (Kugler & Fitch, 2018). The development of this Spatial Data Infrastructure (SDI) project is important because it provides simplified access to big data through its big data integration processes. In this paper, we focus on the development, standardization, and generalization of processes that benefit researchers through IPUMS-Terra.

2 Big Integrated Spatial Data Infrastructure

Big data integration or data fusion is a fast expanding area of research that stands to benefit many scientific research areas through the creation of new datasets and making existing ones more readily available to a broader array of users. New datasets are developed primarily by integrating existing heterogeneous sources while broadening the use of existing ones often involves making them compatible with existing data and analytical processes (Abdalla, 2016; Butenuth et al., 2007). Data fusion techniques have a long history in remote sensing (i.e., feature extraction and value estimations) (Sohn & Dowman, 2007). These techniques are increasingly being applied in industry, academia, health, and the public sectors, but have not yet been substantially automated within a SDI (Deschamps, Greenlee, Pultz, & Saper, 2002; Jiang et al., 2016).

Big data integration has three benefits for the scientific community. First, data integration breaks data silos, allowing researchers from different domains to access and use new data sources. Second, data integration promotes interdisciplinary research through the sharing of datasets. Much of the research conducted in scientific communities is domain specific, in which each discipline develops its models and algorithms for analyzing a particular data structure. The GIS community is no exception, in that many researchers are often experts in either raster or vector data analyses. This singular focus does not lend itself to interdisciplinary or team science research. Integrated data allows researchers to collaboratively apply techniques to the same data. Third, data integration is an opportunity to develop new data products and new models that will utilize heterogeneous datasets from other disciplines.

Integrated big data are of no use without a suitable infrastructure to provide access to these data and, ideally, the tools for analyzing them. Currently, there are few frameworks that support big data integration. In the case of spatial data infrastructure, we have seen the historical development of two classes of SDI which are broadly characterized as geospatial repositories and geospatial computation.

Geospatial repositories or geoportals were the first type of SDI. The term was coined in the 1990s by the US National Research Council (National Research Council, 1993). To date, there are numerous geoportals and they are characterized by their spatial extent (e.g., regional or national scales) and the type of data they contain (Maguire & Longley, 2005). The primary functions of geoportals are geoservice cataloging and data discovery. A shortcoming of geospatial repositories is that while they contain large volumes of data, their geospatial services are typically unable to be applied to big data because they do not adequately scale (Jaeger et al., 2005).

The second class of SDI focuses on geocomputation and can be subdivided into cloud-based computation platforms and geocomputation platforms. Geocomputation in the cloud is a standard way of scaling out geospatial operations (Evangelidis, Ntouros, Makridis, & Papatheodorou, 2014; Yue, Zhou, Gong, & Hu, 2013). Yue et al. (2010) describes a Software as a Service (SaaS) approach for scaling out the GIS software package Geographic Resources Analysis Support System (GRASS) on the cloud. The flexibility of the cloud-

based approach is that it allows for the development of highly customizable geoprocessing tools. A drawback to this approach is that the user must develop the entire process. If the analytical methods employed by the SaaS are not scalable, the resulting workflow will have suboptimal performance. Additionally, the highly specific and complex nature of these models makes them difficult to use for all but the most highly trained and motivated researchers.

Geocomputation platforms such as the CyberGIS Gateway and Geospatial Building Blocks (GABBs) are improvements on the cloud geocomputing approach. These portals provide advanced cyberinfrastructure geospatial tools and services for big spatial data (CyberGIS Gateway, 2016; Geospatial Data Analysis Building Blocks, 2016). The groups provide a number of application-specific parallel spatial libraries through their platforms such as ppsyal and the AgMIP (*GABBS*, 2017, *cybergis toolkit*, 2017). A downside to these platforms is that users are often required to bring data to these platforms for processing, which entails data handling and transfer times. What is more useful for a broader array of users are platforms that grant researchers access to large-scale datasets combined with a computational environment for processing these data, eliminating the need to move data themselves.

IPUMS-Terra offers data and computation via a hybrid cyberinfrastructure – essentially, the combination of a geospatial repository and geocomputation system – designed to allow researchers to focus on their work without dealing with storing, manipulating, or analyzing data. The core functionality of IPUMS-Terra is more similar to that of a spatial data warehouse in that the platform's purpose is analytical (Shekhar, Lu, Tan, Chawla, & Vatsavai, 2001). Data warehouses are renowned for their Online Analytical Processing (OLAP) capabilities, a series of tools capable of analyzing large datasets, performing trend analyses, forecasting, data mining and report generation (Bédard, Merrett, & Han, 2001; Viswanathan & Schneider, 2011). Spatial data warehouses utilize Spatial Online Analytical Processes (SOLAP) to provide knowledge about spatial data (Di Martino, Bimonte, Bertolotto, & Ferrucci, 2009; Rivest, Bédard, Proulx, & Nadeau, 2003).

Conceptually, SOLAP platforms leverage large geospatial data repositories and geocomputational platforms to provide spatial analytical tools for big spatial data (Rivest, Bédard, & Marchand, 2001). IPUMS-Terra is a SOLAP and it benefits the scientific community by enabling geospatial data-driven analysis, exploration, and discovery, without requiring them to have expert knowledge about spatial data analysis.

3 Challenges of Big Integrated Spatial Data Infrastructure

The development of an integrated spatial data infrastructure is not without challenges. A major issue with big data is identifying an architecture that supports all the characteristics of the data structures within the repository. For many projects, this requires utilizing new architectures, such as column-store or Hadoop-like platforms, which do not rely on traditional database indices that become outdated as the data continues to change. For IPUMS-Terra, a hybrid SDI, challenges arise primarily from geocomputation and geocataloging. Geocomputation challenges include the development of services and

architectures that support heterogeneous data processing, while geocataloging challenges focus on the integration of semantic content and metadata throughout the data access system to support data knowledge.

A significant challenge in providing geospatial services is identifying architectures that support heterogeneous data integration. Recently a number of authors have provided reviews on the open source big data platforms for spatial data (Eldawy & Mokbel, 2015; Haynes, Manson, & Shook, 2017; Olasz, Thai, & Kristóf, 2016). Their assessments indicate that essentially all of these platforms focus on implementing one geospatial data structure at the cost of the other (e.g., raster versus vector data models), which places a tremendous burden on the geospatial researcher because it forces the researcher to transform data between models and deal with several different platforms.

Because of the limitations imposed by available spatial data platforms, IPUMS-Terra does not currently utilize any high-performance geospatial system. Instead, it utilizes PostgreSQL with PostGIS for analyzing spatial data as it provides robust support for both vector and raster datasets for geospatial analysis. The system provides access to multiple raster formats including Network Common Data Format (NetCDF) and Georeferenced Tagged Information File (GeoTIFF). However, an ongoing challenge is the development of generalizable methods for processing raster datasets in PostgreSQL. The PostGIS raster datatype was only released in 2012 and does not currently support all the datatypes of the Geographic Data Abstraction Library (GDAL), arguably one of the most commonly adopted and powerful spatial toolkits. In response to these challenges, we provide strategies for incorporating large volumes of raster datasets and explain techniques that reduce the complexity of raster analysis.

A second challenge in developing IPUMS-Terra as a SOLAP platform, is the integration of semantics to support data access, data sharing, and data knowledge. A major goal of this project is to promote the access of geospatial data and methods to users who do not necessarily have the domain-specific skills necessary to access these datasets. Conversely, we give our primary user community of GIScientists access to valuable demographic datasets they, likewise, would not have the domain-specific knowledge to effectively use. For both groups of users, it is crucial that variable concepts and computational methods are represented in a way that makes them accessible to various user communities.

The approach that we have taken to the challenge of semantics is limited in comparison to much of the literature focusing on the standards often proposed for geosemantics (Lieberman & Goad, 2008; Percivall, 2010; Reitsma, Laxton, Ballard, Kuhn, & Abdelmoty, 2009). Instead of attempting to create or use a general semantic framework, our approach primarily focuses on interoperability of the data structures within IPUMS-Terra (Friis-Christensen, Schade, & Peedell, 2005; Vaccari, Shvaiko, & Marchese, 2009). Each data structure has a unique metadata framework developed around it to support its use within the system. This approach of utilizing metadata is similar to those put forth by those working on data integration and semantic enabled spatial data infrastructures, in which metadata is used to inform the processes that are applicable for each variable (Janowicz et al., 2010). Additionally, data and associated metadata that is moved into the system becomes

searchable through our user interfaces. To this end, we have adopted standardized metadata standards, which allow our metadata to be placed into other systems, such as DataOne (DataOne, 2014).

4 IPUMS-Terra

IPUMS-Terra addresses a number of the core challenges of big integrated spatial data infrastructure. The project provides researchers access to new datasets and uses spatial integration to transform them into data structures they already know(Figure 1a & b). Users are typically familiar with one data structure for analysis but may want to access and integrate data found in other formats (Figure 1a). Through IPUMS-Terra guided workflows this process is straightforward. IPUMS-Terra provides access to three different data structures: microdata that is stored in hierarchical flat files, geographic boundaries that are stored as PostGIS geometry types, and raster data that are stored as PostGIS raster types. In order to provide these integrated data, IPUMS-Terra offers six different forms of geotransformations (Figure 1b). IPUMS-Terra utilizes geolocations to perform transformations between existing datasets to allow users access to multiple data structures (Figure 1b).

4.1 IPUMS-Terra Use Case

IPUMS-Terra users can request customized, integrated datasets that contain variables originating from the following data structures: raster, area-level, and microdata. The user receives the dataset in a structure that meets their needs. For example, let's examine the relationship between population growth and natural resource consumption in Brazil from 1990–2010. Traditionally, the researcher would spend many weeks finding and generating all the components of the datasets necessary to complete this analysis. First, they would need to identify and gather the Brazilian decennial census datasets. These datasets are not often found in an easily accessible form and require expertise in demography to properly collect. Second, the researcher would join the censuses by their administrative units to the corresponding geographic boundary files, with the caveat that these files are often missing or incomplete for many regions of the world and that matching spatial units on a map to tabular census data can be complicated. Third, they would gather satellite imagery, or land cover products derived therefrom, and identify the pixel values of interest and perform a reclassification. This series of operations require expertise in remote sensing and geospatial analysis. Lastly, the researchers would develop zonal statistics based on the conflated raster and vector boundaries. Table 1 provides an example of variables that are within the IPUMS-Terra system that would be suitable for this type of analysis.

Each of these steps poses a variety of barriers that require expertise to overcome and that limit the ability of researchers to conduct analyses. Our previous work has identified many of the technical issues as well as the solutions that are implemented in IPUMS-Terra that overcome the challenges of data integration and data processing (Haynes et al., 2017; Haynes, Ray, Manson, & Soni, 2015; Kugler, Manson, & Donato, 2017; Kugler et al., 2015). The power of IPUMS-Terra is that it provides a one-stop shop, where users can make a customized extract utilizing all the data within our system.

An additional benefit is that when using IPUMS-Terra each of these variables is accompanied with detailed metadata that will help facilitate understanding of researchers from differing research domains. Through IPUMS-Terra the researcher can easily identify the data requirements necessary for the analysis and allow IPUMS-Terra to do the data processing. We describe approaches later for developing integrated geospatial workflows.

4.2 IPUMS-Terra Architecture

In addition to handing core tasks of data integration, IPUMS-Terra also offers additional tools for data discovery and access through its application layer (Figure 2). The application layer consists of three different applications that are designed to address the needs of different user communities. The first and main application is the IPUMS-Terra data finder (https://data.terrapop.org/), which allows for the creation of a customized subset of data, called an extract. The data finder has three different guided workflows (i.e., microdata, areal level, and raster) that allow the user to specify from the beginning the data format they would like to use. The second application, TerraClip (https://data.terrapop.org/terraclip) provides raster-only data outputs for users. The third application is TerraScope (https://data.terrapop.org/terrascope) and its primary purpose is data visualization that lets users better understand the spatial and temporal relationships that exist within their data. TerraScope also allows for area-level only data extracts to be generated.

The second layer is the integration layer that translates web requests to actionable queries the underlying system can perform (Figure 2). The integration layer has two components: 1) a Ruby on Rails framework and 2) a geospatial services platform. The Ruby on Rails framework is the primary data integration engine. When a user creates an extract, the Ruby on Rails framework is tasked with assembling the required datasets. The geospatial services platform offers Web Mapping Services and Web Coverage Services, which are primarily used with TerraScope, the data visualization application.

The third and final layer is the data processing layer. The system architecture used for processing is designed to meet the storage and the data analytical needs necessary for all data types. Currently, microdata are stored as flat files. Microdata files are read and tabulated using a custom Java multi-threaded application. The spatial datatypes raster and vector are stored in a PostgreSQL database using the PostGIS extension. PostGIS allows for the storage and analysis of both vector and raster data types. Haynes et al. (2017) discusses some of the obstacles encountered with high-performance computing using both spatial data types.

5 IPUMS-Terra Solutions for Big Data Integration

The development of IPUMS-Terra, a SDI that provides streamlined access to integrated spatial data, is not without its challenges. In this section, we highlight some of the advancements that we have developed when dealing with different data structures that have allowed us to forge an integrated system. We detail these areas below:

1. data intake, where a wide array of microdata, area, and raster data are taken into the IPUMS-Terra system;

2. data manipulation, where data are transformed into new data and among data types; and
3. data discovery and user access.

5.1 IPUMS-Terra Data Intake

IPUMS-Terra provides access to three different data structures: microdata, vector geographic boundaries, and raster datasets. In order to provide access to each of these data structures, they must be brought into the system with appropriate metadata that will allow them to be queried later. Our platform stack is a mix of Ruby on Rails and Python scripts that uses a series of custom-built connectors to move data into the database.

One example of these connectors is Mound, a standalone piece of software written using components from Ruby on Rails. It reads input data or metadata that is in Yet Another Markup Language (YAML) format and loads it into the database. The provided YAML contains specific database foreign-key relationships and supports a belongs-to-many relationship, which can be expressed in more human readable, data-centric terms. By combining Mound with Ruby on Rails models we can define the database schema of IPUMS-Terra and infer data relationships that correctly match database records with related foreign keys or intermediary join table entries. Source code for Mound can be found at https://github.com/mnpopcenter/mound.

5.1.1 Microdata Intake—IPUMS-Terra leverages existing data and metadata from IPUMS *International* (IPUMS-*I*). These two projects are integrated by taking a "snapshot" of the IPUMS-*I* database and metadata and then incorporating them into a new IPUMS-Terra database. IPUMS-*I* processed data are stored by country and year in fixed-width files in a compressed format, while the metadata is stored in MySQL databases. These metadata must be transformed and stored into IPUMS-Terra's PostgreSQL database. The transformation from MySQL to PostgreSQL is achieved by using a tool called jmysqltopostgres (https://github.com/maxlapshin/mysql2postgres). The tool is written in Ruby but utilizes the Java implementation of Ruby, JRuby. The Java implementation allows for greater flexibility and provides access to PostgreSQL's "COPY" command which is necessary for large batch data loading to PostgreSQL. The tool issues a "SELECT" statement to the MySQL database; the result of the select operation is loaded into the PostgreSQL database using a single "COPY". Using this tool results in a duplication of IPUMS-*I* metadata. However, it allows us the opportunity to replicate the IPUMS-*I* Ruby on Rails' ActiveRecord model objects, which are used to represent key microdata elements, such as country, sample, and variable.

5.1.2 Area Data Intake—IPUMS-Terra area data – namely boundary variables and aggregated area-level census data – are loaded using metadata associated with the boundary files. The process begins by initially cross-matching all IPUMS-*I* countries with the appropriate census samples and geographic levels associated with the country. Since all datasets within IPUMS-Terra have explicit temporal definitions, their metadata too must have explicit temporal definitions. This allows countries like South Sudan and Sudan to exist in our system without confusion. Additionally, the presence of aggregate variables like "total

population" vary for each country. Therefore, the variable "total population" may exist for one or more time instances, but the variable "education attainment" may exist once or not at all for a country. In this way, the metadata facilitates the linkage between spatiotemporal geographic boundaries and their associated aggregate variables. A more detailed discussion of the boundary harmonization process can be found in Kugler (2015).

Once a defined country object, which consists of one or more geographic levels (e.g., state and county), is identified we can begin data loading, which takes place in two steps. The first step of data loading is to load the geographic boundaries. The second step loads the variables associated with the boundary. We use a combination of the Ruby language libraries of ActiveRecord and geoRuby for loading geographic files into the database. The geographic boundary files are transformed from Environmental Systems Research Institute (ESRI) shapefiles into Well Known Text (WKT), which can then be stored in an IPUMS-Terra PostgreSQL database.

Area variables (i.e., total population or education attainment) currently are primarily derived from IPUMS-*I* microdata. We use custom written software to derive aggregations, based upon rules found in a Hierarchical YAML file. Once the aggregations have been calculated they are loaded into IPUMS-Terra's database using a "COPY" statement.

5.1.3 Raster Data Intake—Raster data intake also begins with retrieval of metadata for input into the database. Metadata loading is used for determining the types of operations that are possible for a given raster dataset. The intake includes raster variable name, raster spatial metadata, and a list of statistical operations that are appropriate for this dataset. We utilize the existing raster loading tool for PostGIS, raster2pgsql, which produces a tiled representation of the raster in a binary format that can be loaded into PostgreSQL. As we want to minimize the amount of intake necessary in building databases, we have developed a secondary tool in Ruby. This tool takes these preprocessed raster files and stores them in compressed binary format files. Then during the database building process, we are able to access these binary files and load them into the database without having to manually recompile them, reducing loading time by approximately 50%.

Due to the heterogeneous nature of the raster datasets, we take advantage of PostgreSQL's schema concept to better define analysis concepts. Land cover and related rasters reside in a landcover schema, climate rasters in the climate schema, and so on. This segmentation allows for easier management of the system's many rasters. During the database schema building, all the necessary stored procedures for analyzing raster datasets are loaded into the database.

The initial release of the IPUMS-Terra application initially over-simplified the raster data type because the rasters at that time only existed at single points in time. Unlike all our other data structures there was no initial need to have explicit spatiotemporal representations for the raster datasets as they covered the entire globe. We expanded the system as we began to work with time-series raster datasets such as the Moderate Resolution Imaging Spectroradiometer (MODIS) and Climatic Research Unit Time Series (CRU TS). In order to fully utilize both data collections, the metadata collection was redeveloped to support

temporal granularities. The current system is now configured to support vast amounts of sensor-derived data, which are represented by our existing classes of rasters. However, we are aware that there are other raster classes that are not yet represented in our system and our goal is to develop and maintain a flexible generalizable infrastructure that accurately represents and allows for the processing of raster spatial data.

5.2 IPUMS-Terra Data Processing

Once data are loaded into IPUMS-Terra, the data management and manipulation challenges begin. The benefit of the system to researchers is that they can choose data from various data structures and have the data transformed and delivered to them in the format of their choosing. This flexibility for researchers places enormous pressure on the framework to perform complex transformations, given the many unknowns regarding the size and density of the dataset that it is working with for a given user. We present three of the transformations that exemplify the complexity of big data integration and offer innovative approaches to dealing with complex big spatial data.

5.2.1 Microdata Tabulation to Area-level—All area level datasets, which originate as IPUMS-*I* microdata samples (i.e., Brazil 2001) must be tabulated. Tabulation of microdata into area level data begins by defining area level variables based upon their subcomponent microdata variables. Each resulting value must be determined and classified into either of two classes: count and percent. These classifications are used to define variable specific rules for automatic tabulation. Rules are represented as arrays of strings, which are similar to Structured Query Language (SQL) queries (Figure 3). Source code for the tabulator can be found at https://github.com/mnpopcenter/microdata-tabulator.

The tabulator is designed for flexibility and can be used to express operators traditionally used in the SQL “WHERE” statement. The tabulator can utilize operators which specify equality and inequality as well as ranges of values. Specifying multiple lines of rules produces an implicit “AND”; for rules that require more expressive predicate logic, the tabulator also supports parenthetical expressions as well as the “OR” operator.

The YAML description in Figure 2, looks syntactically similar to a SQL “WHERE” statement. However, the tabulator performs its analysis upon the fixed width files not within the database. Therefore, in addition to defining the variable through rules for calculation, the tabulator also employs rules that define the domain in which it operates. The domain is defined by a comma separated list of values that denote the representative start and stopping locations within the microdata file. The length of value to extract can also be defined by these rules.

The last set of rules that must be defined are the geographic tabulation levels, which are represented as a geographic configuration file. Microdata is aspatial with each line representing a person or household. However, each microdata record contains a ten-digit geographic identifier which will be used in the tabulation process. The geographic identifier is similar to the Federal Information Processing Standard (FIPS) code used by the United States government. The identifier contains nested levels of geographic specificity which can be extracted by knowing the range of values needed. The tabulator reads microdata, line by

line, storing the results in memory by geographic level and geographic unit. We provide tabulations for three geographic levels: 0, or national scale; 1, or what is commonly termed by demographers as administrative level one (i.e., state or province); and 2, the administrative level two, which maps onto counties, wards, municipios, and other smaller scale units.

5.2.2 Raster Data GeoTiff Summary—IPUMS-Terra's raster datasets are large and complex. The current application provides spatial analysis capability for over 4,000 rasters containing data on agriculture, land cover, and climate. The metadata becomes an essential tool for clearly identifying the data analyses that are relevant for each data type. Raster summarizations are one of the services/data transformations that we provide through our system. Raster summarization is a method that provides an overview of a raster dataset for a given vector region. IPUMS-Terra identifies which raster summarizations are appropriate for the given raster dataset (Figure 4).

To accurately process and disseminate spatial data analyses, we utilized procedures that are stored within the database. Stored procedures are written in PostgreSQL's supported procedural languages plpgsql or plpython. By way of example, we briefly discuss the stored procedures written in plpgsql that are currently used for conducting zonal statistics. The IPUMS-Terra framework contains multiple classes of rasters (e.g., float, categorical, binary, and custom) in Figure 4. All raster summarization stored procedures operate at two levels and use common table expressions (CTE) to improve the performance (Figure 5).

The top-level functions focus on vector dataset verification. PostGIS is very lenient in that it will load corrupt boundaries (e.g., boundaries with topology errors such as intersecting vertices) into its system that have the potential for failing during subsequent function calls. In practice, we have found that the function ST_Clip has high topological requirements and does not tolerate any topological errors. It is also possible to introduce new topological errors when converting a vector dataset to a new projection. The top-level function's purpose is to project boundaries if necessary, then verify and remove any topological errors and then pass these clean boundaries to a secondary procedure which performs the zonal statistics.

The secondary processing functions have been developed to improve performance for specialized datasets. The development of highly tuned functions allows us to eliminate errors and improve performance when conducting raster summary statistics. We provide pseudo code and a description of two of these secondary data processing functions. The processing functions allow different analysis products to be derived from a categorical landcover raster. In the example, we demonstrate how different analytics and be performed on the same landcover raster. Additional stored procedures can be found https://github.com/mnpopcenter/ipums-terra-spatial-functions.

The first stored procedure, Figure 6, returns the mode category and the number of unique categories, which is computationally intensive because the system must count all pixels and their values that fall within the boundaries. Additionally, PostgreSQL does not have a modal function, therefore we use a windowing function from PostgreSQL to return the largest category for each geographic unit.

The second stored procedure, Pseudo code 2, utilizes two-raster datasets, to determine the percent area of a particular landcover value(s). We utilize the ST_Reclassification function of PostGIS to extract the specific landcover values of interest. In this case, we have not used the multiband capabilities of PostGIS for storing our raster datasets, instead, each of these datasets are stored as separate tables. The first dataset is the landcover or data raster and the second is the area reference raster. The landcover dataset would be the same dataset as what was used in Stored Procedure 1, whereas the area reference raster is a floating-point raster and each pixel value contains the area value of the earth's surface. This calculation utilizes both the area reference raster and the data raster in the analysis. By employing the ST_MapAlgebra function we are able to determine results on demand for any given user input. A benefit of this approach is that when new reclassification variables are added in the raster metadata, the underlying stored procedures perform the calculations without any changes to the calculation stored procedures.

5.2.3 IPUMS-Terra NetCDF Data—The integration of the Climate Research Unit Time Series (CRU TS) into IPUM Terra presented new challenges emblematic of more general challenges facing SDI, namely issues of data conversion and size. We encountered two problems integrating the CRU TS data into our system. NetCDF files cannot be directly read into PostGIS and while it is possible to convert each NetCDF file into a GeoTIFF, the in-database file size would be prohibitively large, with over 1300 raster bands for one climate variable. The spatial resolution of the CRU TS dataset is also an issue with a relatively coarse resolution of 0.5 decimal degrees. This coarse resolution also limits the efficacy of raster summarizations. Most GIS systems, including PostGIS, rasterize the polygons to the resolution of the raster cells when performing raster summarization. In doing this, many small geographic units are eliminated. In an effort to preserve geographic detail, we developed a polygon template of the CRU TS dataset. When the CRU TS template is intersected with the boundary polygons, every boundary polygon is assigned a portion of a CRU TS pixel. Polygons that intersect multiple CRU TS pixels have their resulting value determined through proportional allocation.

The volume and temporal granularity of the CRU TS dataset encouraged new architecture development. The CRU TS dataset is loaded into a large denormalized table, with functional indices on the following columns: pixel_id, id, year, month, and year-month combination. The pixel_ids are unique identifiers that represent the same geographic location across all variables and time points and are designed to be joined to the CRU TS template. When a user makes a request from the CRU TS dataset, IPUMS-Terra conducts a single spatial overlay query between the geographic boundary and the template. This query identifies all CRU TS pixel_ids that intersect the boundary of interest. These pixel_ids are joined to the larger CRU TS table for further calculations, which invoke secondary processing functions. This approach is similar to GeoTIFF Raster Summarization approach.

The development of the large denormalized table facilitates the raster summarization process, but it does not work well for researchers who would like to receive the CRU TS data as a GeoTIFF. In order to deliver data in this format, we utilize the Web Coverage Service (WCS) of GeoServer. GeoServer's WCS supports time slicing and cropping of NetCDF files, which can be integrated into a stored procedure call from PostgreSQL. While

the data is delivered in the proper format, the result is not ideal as a single dataset is stored in two separate locations to support separate delivery products. Currently, we are investigating other alternatives.

5.3 IPUMS-Terra Data Delivery

While the data loading and data processing steps of an integrated system are technically challenging, the system is of no use if the user does not get their data. The User Interface (UI) functions like a wizard or workflow, guiding the user along the path and building a customized JavaScript Object Notation (JSON) file which is submitted to the system for processing. The result is a customized data extract in any of the three main formats: microdata in a CSV, aggregate data in CSV with associated ESRI Shapefile, and raster data as a GeoTIFF. These data structures fit the needs of a broad variety of members of our user community. Extract sizes from IPUMS-Terra can be large depending on the type and amount of data sources. For example, a microdata extract for France 2011 is very large and would constitute what we would call a “tall” extract with approximately 20 million-person records. This extract would be several gigabytes with the standard eight microdata default variables of country, year, sample, household serial number, form type, person number, person weight, and residence status. With the addition of variables like age, sex, and income, the extract size increases and grows wider. Adding additional contextual variables (which are derived from raster environmental datasets) exacerbates this process and potentially creates an unwieldy extract. We recommend that users request extracts with only the variables they need to improve usability. Typical user extracts are tall and narrow (e.g., microdata) or short and wide (area-level) and even those extracts can be several gigabytes.

A related challenge lies in raster summarization. Raster summarizations were initially degrading the performance of our system as they are computationally expensive. Haynes (2017) discusses why query performance times of raster analyses with PostgreSQL vary greatly and we have implemented a data caching system to reduce additional calculations. The caching key is generated from the following variables: geographic level, temporal time point or range, and the raster variable and raster operation (e.g., minimum value, maximum value). When any raster analysis is run the resulting value or values will be produced and stored with that key. The values are subsequently cached in a raster summarization table that can be keyed off from the geographic level’s identification id, a raster variable identification number, the summarization operation, and in some cases the raster band number. The caching mechanism works by first checking this caching table for any matches based on the aforementioned keys, and if nothing is found, the associated summarization stored procedure is called, and the values from that are then stored in the cache table.

6 Conclusion

The next step in the big data revolution is the integration of disparate data types, models, and sources. IPUMS-Terra focuses on the integration of heterogeneous data to promote the understanding of population and environment. IPUMS-Terra is a next-generation spatial data infrastructure project that demonstrates the value of linking big data computation to big data storage. Much of the value of this project lies in the benefit it provides to the broader

research community by allowing fast and easy access to geospatial data and geospatial analytical tools.

Integrating big data is a complex and challenging process as it involves aspects of computation and semantic integration. We have found that metadata, in many cases, is the connective tissue of this project. By integrating metadata in almost every stage of this project we have reduced complexity and redundancy of data while improving conceptual understanding of the system and data processing performance for the user. A critical challenge that still remains is developing better methods for storage, retrieval, and analysis of these datasets. In particular, there are no comprehensive high-performance computing solutions that natively deal with both raster and vector data formats. The lack of an integrated framework results in degraded performance when analyzing raster and vector datasets.

In sum, IPUMS-Terra is a new class of spatial data infrastructure whose goal is to provide fast and convenient access to big integrated data. The framework we have developed to support this project is a flexible stack of open-source tools and methods that we freely share with the research community. IPUMS-Terra continues to work with other research initiatives and a range of user communities to address pertinent research challenges concerning the integration of heterogeneous spatial data and the development of next-generation infrastructures and tools to support heterogeneous data integration.

7 Acknowledgments

We would like to thank the editor and peer reviewers for providing feedback on this article. Additionally, we would like to thank members of the Institute of Social Research and Data Innovation and the data project teams IPUMS-I and IPUMS-Terra for their help in this process. Special thanks to the IT-Core for assisting with this development process. The research in this article is supported by National Institutes of Health Award 5T32CA163184.

References

AgMIP. (2017). Retrieved from https://mygeohub.org/groups/gabbs/project_page

Armstrong MP (2000). Geography and computational science. Annals of the Association of American Geographers, 90(1), 146–156.

Bédard Y, Merrett T, & Han J (2001). Fundamentals of spatial data warehousing for geographic knowledge discovery. Geographic Data Mining and Knowledge Discovery, 2, 53–73.

CyberGIS Gateway. (2016). Retrieved November 1, 2016, from http://sandbox.cigi.illinois.edu/home/

cybergis-toolkit. (2017). Retrieved from https://github.com/cybergis/cybergis-toolkit

DataOne. (2014). DataONE. Retrieved August 1, 2017, from https://www.dataone.org/news/dataone-welcomes-its-25th-member-node-minnesota-population-center

Deschamps A, Greenlee D, Pultz T, & Saper R (2002). Geospatial data integration for applications in flood prediction and management in the Red River Basin. In Geoscience and Remote Sensing Symposium, 2002. IGARSS'02. 2002 IEEE International(Vol. 6, pp. 3338–3340). IEEE.

Di Martino S, Bimonte S, Bertolotto M, & Ferrucci F (2009). Integrating google earth within olap tools for multidimensional exploration and analysis of spatial data. In International Conference on Enterprise Information Systems (pp. 940–951). Springer.

Ding Y, & Densham PJ (1996). Spatial strategies for parallel spatial modelling. International Journal of Geographical Information Systems, 10(6), 669–698.

Eldawy A, & Mokbel MF (2015). The era of big spatial data. In Data Engineering Workshops (ICDEW), 2015 31st IEEE International Conference on (pp. 42–49).

Evangelidis K, Ntouros K, Makridis S, & Papatheodorou C (2014). Geospatial services in the Cloud. Computers & Geosciences, 63, 116–122. 10.1016/j.cageo.2013.10.007

Friis-Christensen A, Schade S, & Peedell S (2005). Approaches to solve schema heterogeneity at European Level. In Proceedings of 11th EC-GI & GIS Workshop, ESDI: Setting the Framework, Alghero, Sardinia, Italy.

Geospatial Data Analysis Building Blocks. (2016). Retrieved September 15, 2016, from https://purr.purdue.edu/projects/geodibbs

Haynes D, Manson S, & Shook E (2017). Terra Populus' Architecture for Integrated Big Geospatial Services. Transactions in GIS, 21(3), 546–559. 10.1111/tgis.12286

Haynes D, Ray S, Manson SM, & Soni A (2015). High performance analysis of big spatial data. In Big Data (Big Data), 2015 IEEE International Conference on (pp. 1953–1957). IEEE.

Jaeger E, Altintas I, Zhang J, Ludäscher B, Pennington D, & Michener W (2005). A Scientific Workflow Approach to Distributed Geospatial Data Processing using Web Services. In *SSDBM* (Vol. 3, pp. 87–90).

Janowicz K, Schade S, Bröring A, Keßler C, Maué P, & Stasch C (2010). Semantic Enablement for Spatial Data Infrastructures. Transactions in GIS, 14(2), 111–129. 10.1111/j.1467-9671.2010.01186.x

Jiang P, Winkley J, Zhao C, Munnoch R, Min G, & Yang LT (2016). An intelligent information forwarder for healthcare big data systems with distributed wearable sensors. IEEE Systems Journal, 10(3), 1147–1159.

Kugler TA, & Fitch CA (2018). Interoperable and accessible census and survey data from IPUMS. Scientific Data, 5, 180007 10.1038/sdata.2018.7 [PubMed: 29485621]

Kugler TA, Manson SM, & Donato JR (2017). Spatiotemporal aggregation for temporally extensive international microdata. Computers, Environment and Urban Systems, 63, 26–37. 10.1016/j.compenvurbsys.2016.07.007

Kugler TA, Van Riper DC, Manson SM, Haynes II DA, Donato J, & Stinebaugh K (2015). Terra Populus: Workflows for integrating and harmonizing geospatial population and environmental data. Journal of Map & Geography Libraries, 11(2), 180–206.

Laney D (2001, February 6). 3D Data Management: Controlling Data Volume, Velocity, and Variety. Retrieved August 21, 2017, from http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf

Lieberman J, & Goad C (2008). Geosemantic Web Standards for the Spatial Information Infrastructure In van Oosterom P & Zlatanova S (Eds.), Creating spatial information infrastructures: Towards the spatial semantic web (pp. 119–128). CRC Press.

Maguire DJ, & Longley PA (2005). The emergence of geoportals and their role in spatial data infrastructures. Computers, Environment and Urban Systems, 29(1), 3–14.

Minnesota Population Center. (2011). National Historic Geographic Information System (Version 2.0). Minneapolis, MN: University of Minnesota Retrieved from http://www.nhgis.org

Minnesota Population Center. (2017). Integrated Public Use Microdata Seris, International (Version 6.5). Minneapolis, MN: University of Minnesota Retrieved from http://www.nhgis.org

National Research council, M. S. (1993). Toward a coordinated spatial data infrastructure for the nation. National Academies Press.

Olasz A, Thai BN, & Kristóf D (2016). A NEW INITIATIVE FOR TILING, STITCHING AND PROCESSING GEOSPATIAL BIG DATA IN DISTRIBUTED COMPUTING ENVIRONMENTS. ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences, 3(4).

Percivall G (2010). Progress in OGC Web Services Interoperability Development In Di L & Ramapriyan HK (Eds.), Standard-Based Data and Information Systems for Earth Observation (pp. 37–61). Berlin, Heidelberg: Springer Berlin Heidelberg 10.1007/978-3-540-88264-0_4

Reitsma F, Laxton J, Ballard S, Kuhn W, & Abdelmoty A (2009). Semantics, ontologies and eScience for the geosciences. Computers & Geosciences, 35(4), 706–709. 10.1016/j.cageo.2008.03.014

Rivest S, Bédard Y, & Marchand P (2001). Toward better support for spatial decision making: defining the characteristics of spatial on-line analytical processing (SOLAP). GEOMATICA-OTTAWA-, 55(4), 539–555.

Rivest S, Bédard Y, Proulx M-J, & Nadeau M (2003). SOLAP: a new type of user interface to support spatio-temporal multidimensional data exploration and analysis. In Proceedings of the ISPRS Joint Workshop on Spatial, Temporal and Multi-Dimensional Data Modelling and Analysis, Quebec, Canada (pp. 2–3).

Shekhar S, Lu C, Tan X, Chawla S, & Vatsavai R (2001). A visualization tool for spatial data warehouses. Geographic Data Mining and Knowledge Discovery, 73.

Sohn G, & Dowman I (2007). Data fusion of high-resolution satellite imagery and LiDAR data for automatic building extraction (Vol. 62). 10.1016/j.isprsjprs.2007.01.001

Vaccari L, Shvaiko P, & Marchese M (2009). A geo-service semantic integration in spatial data infrastructures. IJSDIR, 4, 24–51.

Viswanathan G, & Schneider M (2011). On the requirements for user-centric spatial data warehousing and SOLAP. In International Conference on Database Systems for Advanced Applications (pp. 144–155). Springer.

Yue P, Gong J, Di L, Yuan J, Sun L, Sun Z, & Wang Q (2010). GeoPW: Laying Blocks for the Geospatial Processing Web. Transactions in GIS, 14(6), 755–772. 10.1111/j.1467-9671.2010.01232.x

Yue P, Zhou H, Gong J, & Hu L (2013). Geoprocessing in Cloud Computing platforms – a comparative analysis. International Journal of Digital Earth, 6(4), 404–425. 10.1080/17538947.2012.748847

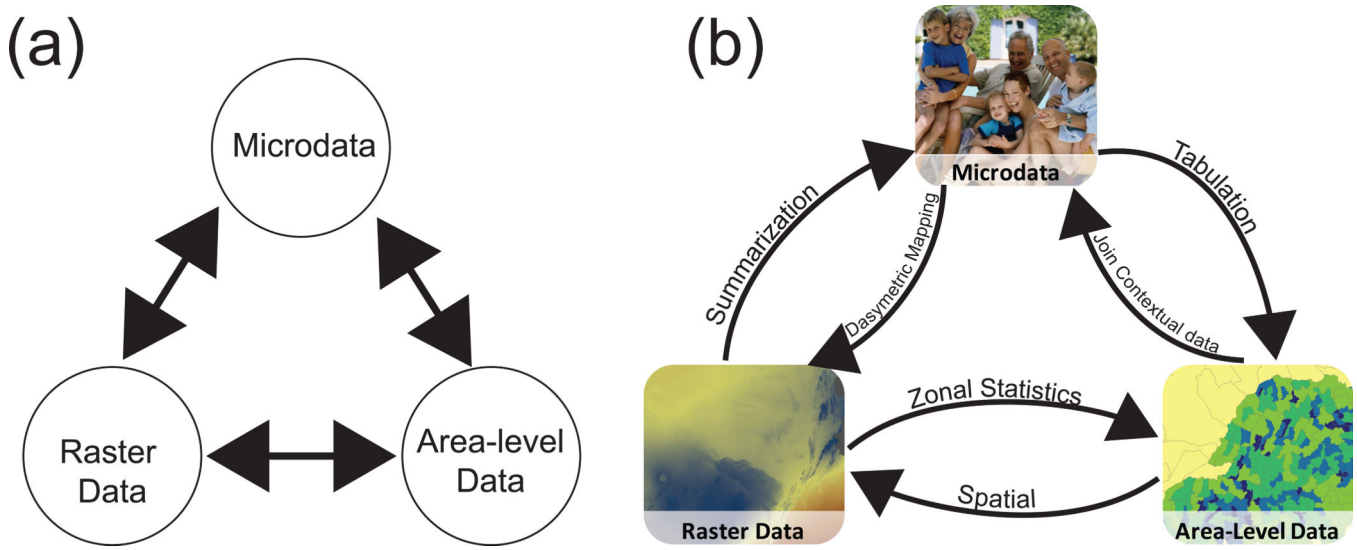

Fig. 1.
IPUMS-Terra Data Structures and Data Transformations

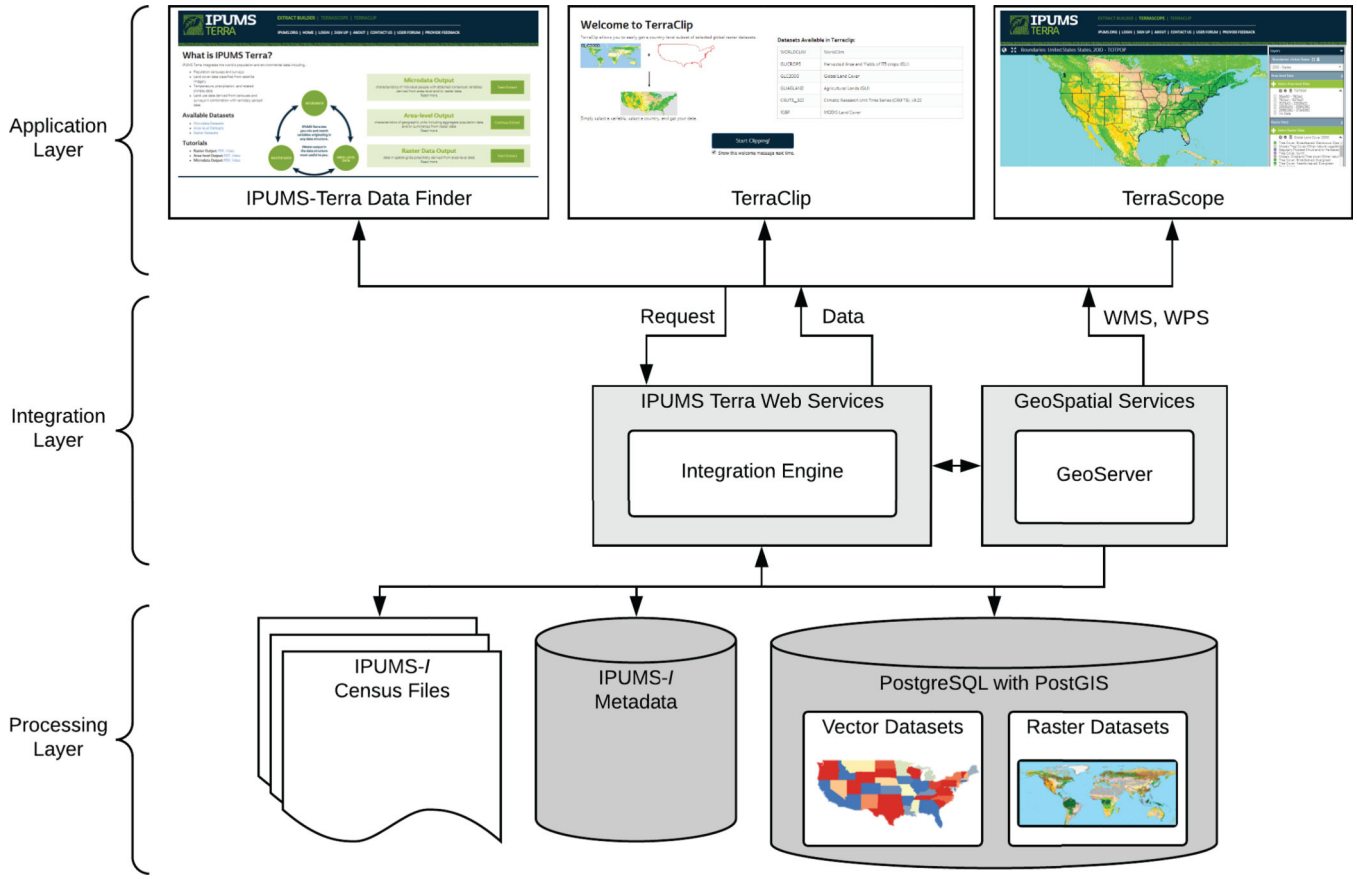

Fig. 2.
IPUMS-Terra System Overview

```
name:  “Percent of households with a television”
code:    TELEVISION
description:    “This variable reporst the percentage of households
                        in the geographic unit that have a television”

data_table_id: TELEVISION
operation: Percent
query_variable: RECTYPE
query_on: Household
numerator:
        where_person:
                -[]
        where_household:
                - [GQ, “=” 10]
                - [TV, “BETWEEN”, 20, 54]
denominator
        where_person:
                - []
        where_household:
                - “([GQ, = , 10]) AND ([TV, BETWEEN, 10, 54])”
```

Fig. 3.
IPUMS-TERRA YAML syntax for microdata tabulation

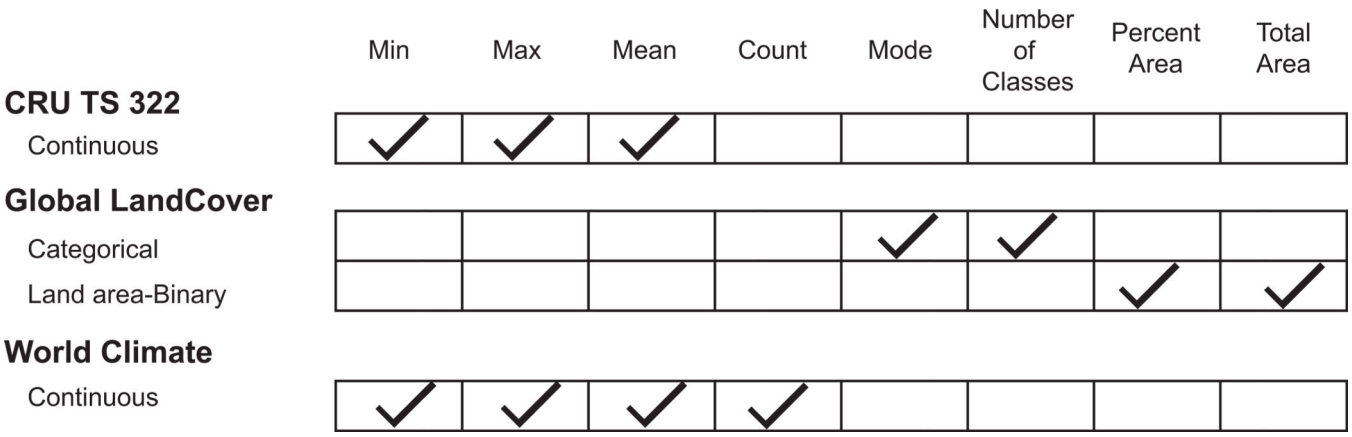

Fig. 4.
IPUMS-Terra Raster Summary Operations

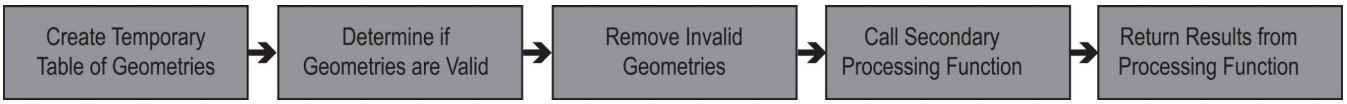

Fig. 5.
PostgreSQL Stored Procedure Workflow for Raster Summaries

Stored Procedure 2 Binary Reclassification Raster Summary Statistics

```
1.  TerraPop_Binary Summarization(sample geographic_id, raster_variable_id)
2.   Select cleaned boundaries
3.   Lookup reclassification values from raster_variable_id        (e.g., 3: 1)
4.   Reclassify and Clip categorical raster to geometry           Reclassified
                                                                  Raster
5.   Clip area reference raster to geometry                       Area Raster
6.   Use ST_MapAlgebra to multiply reclassified raster by area
     raster and perform ST_SummaryStatsAgg on resulting dataset
7.  Return percent area, reclassified area
```

Fig. 6.
Pseudo Code for Categorical Summaries

Stored Procedure 1 Categorical Raster Summary Statistics

```
1.  TerraPop_Categorical Summarization(sample geographic_id, raster_variable_id)
2.   Select cleaned boundaries
3.   Clip raster to geometry
4.   Use ST_ValueCount to identify all pixel values and count
5.   Aggregate pixels by geographic unit and pixel value
6.   Count number of categories
7.   Use windowing function to identify modal category
8.  Return mode, number_of_categories
```

Fig. 7.
Pseudo Code for Binary Summary

Table 1

IPUMS-Terra Sample Extract for Brazilian Resource Consumption

Area Level Variables	**Raster Summarized Variables**
POPTOTAL: Total Population	EVRGRNBRDLF: Percent area of evergreen broadleaf forest
AGRIC: Percent of workforce in agriculture	URBAN: Percent urbanized area
URBANIZ: Percent of population living in urban areas	CROPLAND: Percentage area devoted to cropland
UNEMP: Percent of population unemployed	