Datenbank Spektrum (2017) 17:233–243
https://doi.org/10.1007/s13222-017-0266-5

SCHWERPUNKTBEITRAG

VAT: A Scientific Toolbox for Interactive Geodata Exploration

Christian Beilschmidt[1] · Johannes Drönner[1] · Michael Mattig[1] · Marco Schmidt[2] · Christian Authmann[1] · Aidin Niamir[2] · Thomas Hickler[2,3] · Bernhard Seeger[1]

Received: 10 June 2017 / Accepted: 8 September 2017 / Published online: 21 September 2017

Abstract Data-driven research requires interactive systems supporting fast and intuitive data exploration. An important component is the user interface that facilitates this process. In biodiversity research, data is commonly of spatio-temporal nature. This poses unique opportunities for visual analytics approaches. In this paper we present the core concepts of the web-based front end of our VAT (Visualization, Analysis and Transformation) system, a distributed geo-processing application. We present the results of two user studies and highlight unique features, among others for the management of time and the generalization of data.

Keywords Visualization · Biodiversity · Scientific Workflows

1 Introduction

Recently, research has become increasingly data-driven. Researchers often form new ideas by exploring large databases and identifying interesting patterns, instead of collecting data with a concrete hypothesis already in mind. Visual analytics plays an important role in this approach. It provides the necessary tools for researchers in order to facilitate effective data exploration. In biodiversity research a large fraction of the data is inherently spatio-temporal, i. e. the position of objects can be represented in a coordinate system at a certain point in time. This makes it especially appealing for a visual analytics approach, as spatial data can naturally be visualized on a map but also in a tabular view and by using different plots.

While data-driven research offers many new scientific opportunities, it also poses challenges for users regarding data integration, cleansing, filtering and lineage. First, there is a multitude of publicly available heterogeneous data which users want to combine, possibly also with their own data. There are different types of data, e. g. vector and raster data, and different reference systems for space and time. Second, data often appear as time series and compu-

This is an extended version of the paper „Interactive Data Exploration for Geoscience“ [5] selected for the special DASP issue *Best Workshop Papers of BTW 2017*.

✉ Christian Beilschmidt
beilschmidt@mathematik.uni-marburg.de

Johannes Drönner
droenner@mathematik.uni-marburg.de

Michael Mattig
mattig@mathematik.uni-marburg.de

Marco Schmidt
marco.schmidt@senckenberg.de

Christian Authmann
authmanc@mathematik.uni-marburg.de

Aidin Niamir
aidin.niamir@senckenberg.de

Thomas Hickler
thomas.hickler@senckenberg.de

Bernhard Seeger
seeger@mathematik.uni-marburg.de

[1] Department of Mathematics and Computer Science, University of Marburg,
Hans-Meerwein-Straße 6, 35032 Marburg,
Germany

[2] Senckenberg Biodiversity and Climate Research Centre
(BiK-F), Senckenberganlage 25, 60325 Frankfurt am Main,
Germany

[3] Department of Physical Geography, Goethe University,
Altenhöferallee 1, 60438 Frankfurt am Main, Germany

tations need to take this into account in order to produce valid results. Third, the size of individual data sets poses challenges as high resolution raster images easily exceed hundreds of gigabytes. It is not feasible to store and process such data on regular desktop hardware using standard software. Fourth, specific subsets of data often have quality issues. An appropriate visualization has to support identifying errors and relevant subsets. Fifth, the composition of complex workflows makes it generally difficult for users to keep track of the data lineage. In particular, this poses challenges in correctly citing the source data. We are not aware of a single system addressing all these challenges for biodiversity science.

Our Visualization, Analysis and Transformation system (VAT) [3, 4, 6] aims to support such an interactive data exploration for biodiversity data. VAT consists of a back end for low-latency geo processing called MAPPING (Marburg's Analysis, Processing and Provenance of Information for Networked Geographics) and a web front end called WAVE (Workflow, Analysis and Visualization Editor). The main purpose of VAT is to pre-process data and to export results for further analysis in custom tools. The user interface is of utmost importance in order to effectively enable data-driven science on large and heterogeneous data for scientific users with little background in information technology.

We develop VAT as a part of the GFBio[1] project [9], a national data infrastructure for German biodiversity research projects. GFBio's overall goal is to provide long-term data access in order to facilitate data sharing and re-usage. VAT provides important added-value services for exploring and processing the data of the GFBio data centers. It is also in use in the Idessa[2] project that deals with sustainable rangeland management in South African savannas. Here, VAT is used as a toolbox for implementing a web-based decision support system for farmers to avoid land degradation.

This paper presents the overall architecture of VAT and our design decisions. Our main contributions are: we present an interface for exploratory workflow creation, effective data generalization and previews, linked time series computations, and automatic provenance and citation tracking. We verified the validity of our approach in two user studies.

The rest of the paper is structured as follows. Sect. 2 briefly discusses the motivation for building a new system by presenting other work in this area. Sect. 3 describes the architecture of VAT and the underlying data model. The main part in Sect. 4 gives a general overview of our web-frontend and discusses several aspects of the system in more detail. Sect. 5 presents a short non-trivial scientific use case accomplished with VAT to give a more practical idea of its capabilities. Sect. 6 describes the results of our two user studies that concern the design and implementation phase, respectively. Finally, Sect. 7 concludes the paper.

2 Related Work

There is an ongoing scientific interest in interactive map applications [16, 2] for which the processing time needs to be sufficiently low. In general, current visual analytic approaches [18] meet this requirement, but lack the ability to track workflow provenance and modify configurations.

Typically, data processing in geo sciences is either done using scripting languages like R or Geographic Information Systems (GIS) like QGIS[3]. Writing R programs requires knowledge of the language and the required packages. The development takes time and the processing speed is limited. GIS offer a graphical user interface that requires less programming skills. However, desktop GIS have the following deficiencies. First, they suffer from slow processing as they are limited to local resources and do not exploit modern hardware sufficiently well [4]. Second, workflow builders for GIS do not support an exploratory usage. Third, to the best of our knowledge there exists no GIS that treats every data set as a time series and produces derived data sets with valid temporal information dynamically on demand. Existing GIS like GRASS [11] have extensions that allow processing temporal data sets. However, to match their existing processing model, they process the data upfront rather than on demand. Moreover, they store intermediate steps on disk as a new data set which leads to a high consumption of storage space and slow response times for exploratory data analysis.

Web-based applications allow for ubiquitous access and are able to provide more processing power than desktop applications. There are specialized applications like Map Of Life [13] that aim to solve specific use cases very well. More general functionality is provided by cloud-based GIS like CartoDB[4] and GIS Cloud[5]. However, the processing capabilities of these systems is still limited as they mainly focus on map creation rather than scientific processing tasks.

In contrast to that, Zhang et al. [20] are developing a web-based GIS for exploratory usage incorporating GPU-accelerated processing. Their work focuses on extensive indexing support for evaluating point in polygon predicates. They base their front-end on the Google Map API[6] for performance and ease of use but focus mainly on the back-end part. We, in fact, consider a good user interface to be

[1] https://www.gfbio.org

[2] http://www.idessa.org

[3] http://www.qgis.com

[4] https://www.carto.com

[5] https://www.giscloud.com

[6] https://developers.google.com/maps/

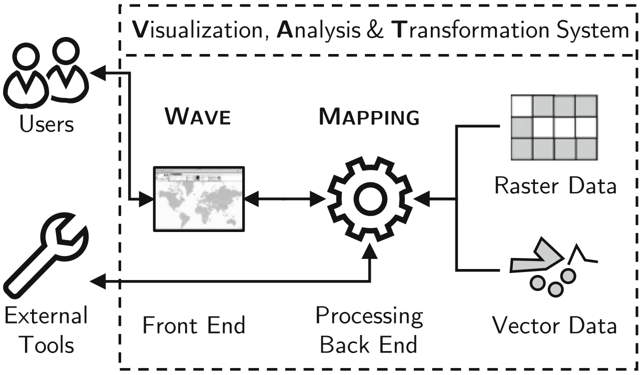


Fig. 1 A condensed view of the system architecture

equally important as a high-performance back end. Thus, we spent particular effort in accomplishing a high level of usability. We conducted extensive user studies to validate our user interface design and identify opportunities for improvements.

Workflow systems like Taverna [19], Kepler [1] and Pegasus [14] build workflows upfront, with the goal of executing them on multiple data sets. This is contrary to our desired exploratory approach of incremental workflow creation. Additionally, they do not allow web-based exploratory workflow creation, offer little geo functionality and limited processing throughput [4].

Another desirable aspect of scientific workflow systems is the automated generation of citations. While none of the previously discussed systems provides this feature, Buneman et al. [8] tackled this task for database systems in general. The idea is to subdivide the input data into so-called citable units. This means for specifying the citations, this is the most fine-grained level to look at. They use common query rewriting techniques to transform the query into views such that each view corresponds to a citable unit. If the view is still in the re-written query, the data's metadata is part of the citation. While the problem of query re-writing is NP-hard in general, there are applicable heuristics to use. As VAT uses custom operators for each processing step, the citation tracking is built-in there. This means there is no need for rewriting queries afterwards. We detail this concept in Sect. 4.

3 VAT Architecture and Data Model

VAT consists of a web-based front end (WAVE) and a high-performance distributed back-end (MAPPING) with an index node and multiple workers. Users interact with the back-end either through the web interface or using external tools (Fig. 1). In both cases the interaction is based on a combination of custom APIs and well-known OGC[7] protocols. The basic design of VAT aims at supporting interactive users that explore data and incrementally add new processing steps.

In VAT, all data has a spatial and temporal context. This context describes the validity of an item in terms of a *location* and a *time interval*. In the same manner, a query specifies a spatial bounding box and a temporal interval. The system returns all results that are valid within these constraints. As time is an integral part of our data model, we consider all data sets as time series.

Our system supports two very different data types: raster and vector data. A raster is a uniform grid of numeric values. Cells contain either continuous values, e. g. temperature measurements, or discrete values, e. g. a classification based on land usage. In our data model all cells of a raster have the same temporal validity. A raster data set consists of multiple rasters with disjoint temporal validity. Vector data is modelled as in the simple feature model [15]. Here, a data set consists of a set of features with attributes. To achieve clean semantics for our operators, we only allow homogeneous collections of features. Thus, a data set consists either of points, lines or polygons. Each feature is optionally a *multi*-feature consisting of multiple points, lines or polygons. The attributes for each feature are a key-value map. VAT supports textual and numeric values.

The system processes data as *workflows*. A workflow consists of all inputs and processing steps and describes how they are connected. A query to VAT thus consists of a spatio-temporal query rectangle and a workflow. The interactive data processing in VAT means that a final workflow is built incrementally. Thus, consecutive queries typically make use of previous results. In order to avoid expensive and redundant re-computations the intermediate results of a workflow are stored in a cache. Subsequent queries that match, contain or overlap cached results can thus be answered faster.

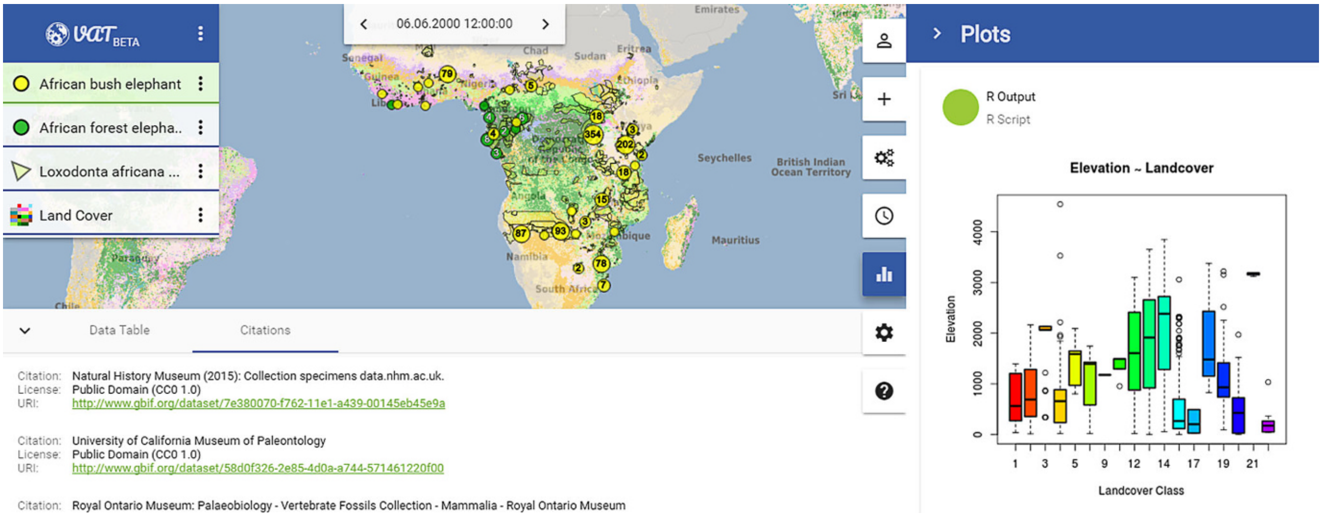
4 WAVE Overview and In-Depth Presentation

This section presents VAT's capabilities on the basis of the web-based user interface WAVE. It first gives an overview of the general design approach and interaction design. Then, it discusses several individual aspects including the connectivity to GFBio, the processing of data and lineage tracking, and the visualization and export of data.

4.1 General Overview

The fundamental idea for WAVE is to offer an intuitive web-based user interface for interactive exploration of biodiver-

[7] Open Geospatial Consortium, http://www.opengeospatial.org



Citation: Natural History Museum (2015): Collection specimens data.nhm.ac.uk.
License: Public Domain (CC0 1.0)
URI: http://www.gbif.org/dataset/7e380070-f762-11e1-a439-00145eb45e9a

Citation: University of California Museum of Paleontology
License: Public Domain (CC0 1.0)
URI: http://www.gbif.org/dataset/58d0f326-2e85-4d0a-a744-571461220f00

Citation: Royal Ontario Museum: Palaeobiology - Vertebrate Fossils Collection - Mammalia - Royal Ontario Museum

Fig. 2 This screenshot gives an overview of the main parts of WAVE. There is a layer list on the left side, a list of citations in the bottom and a tool view (which currently displays a plot) on the right side. Additionally, there is a temporal reference component on top

sity data. Users select data from a repository and upload custom files. They perform operations for filtering or enriching data by combining them with other sources of information. Finally, they export data for further analysis in their preferred custom tools on a different system. The complete workflow of computations is always accessible and allows a reproduction of results anytime later. It describes the dataflow from source collections over processing operators to a final result. A JSON representation makes the workflows storable and shareable. WAVE builds a workflow on-the-fly in the background when a user performs actions on selected data. It thus keeps track of all the applied processing steps.

The central part of the data visualization is a panel with a map that consists of multiple layers of geographic data and shows them in their spatial and temporal context (Fig. 2). The map supports different coordinate systems and allows for panning and zooming. It is linked to a data table that contains further information about non-spatial attributes. All data, either from sources or results of computations, are represented as separate layers. Users can also specify the order in which layers are drawn on the map. Layers refer to a workflow that is part of the query processed in MAPPING. They also serve as inputs for operators in order to create a new workflow. Beside layers, a workflow can also output plots, e. g. histograms or scatter plots.

Layers and plots are linked to a component that allows the selection of the temporal reference. Here, users specify the point in time that is transferred to MAPPING as part of the spatio-temporal context. A change of the temporal reference triggers a re-computation of all layers and plots. WAVE supports a video mode for which the user specifies a time interval. Then, the computation slides over the interval, continuously producing the outputs of layers and plots. This effectively visualizes the changes in the data over time.

Adding a new layer consists of either selecting data from a source or applying an operator on one or more existing layers. In order to allow users to easily combine their own data with important environmental information, WAVE offers an interface to access a repository of raster and vector data hosted by MAPPING. In addition, users may upload their own data represented in the popular CSV data format. Operators allow the selection of appropriate data sets as inputs. One of the benefits of WAVE is that there is a check whether inputs fit to operators. If not, it tries to transform the data, e. g. adapt to the coordinate system, to make it compatible. By automatically applying such transformations, users can easily integrate heterogeneous data sets in their scientific workflows. All such implicit processing steps are reflected in the workflow and as such made visible to the user.

Fig. 2 shows a project that contains four layers and one plot. The layers are of different types, point and polygon collections and a raster time series. A user can configure the global reference time in a component on the top. The bottom displays a list of citations and contains a switch to display a data table. On the right there is a tool bar with multiple options from adding source data, over applying operators to displaying plots. The boxplot shows classified data with respect to elevation data from one of VAT's repositories.

Users can work in multiple projects, each of them offers its specific data sets, workflows, layers and plots. The auto-save feature of WAVE ensures that projects are always up-to-date. A flexible rights management allows sharing projects in a team.

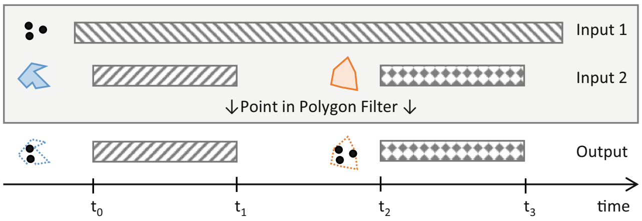


Fig. 3 Temporal point-in-polygon filter

4.2 Connectivity to GFBio

VAT is an integral component of the GFBio portal. This portal is the main entry point of the GFBio project, a federation of German biodiversity data centers. In GFBio, users can select the most relevant data center for their data, create data management plans and submit their data for long term archival. The existing data sets are indexed and searchable by other users to facilitate sustainable data re-use.

A GFBio portal account allows users to log into VAT and access the data of the GFBio archives. The system lists the existing data sets and allows adding them as layers to the map. The user also has access to her/his past searches of the GFBio data index. Complete data sets as well as individual data units discovered by the GFBio search on the meta-data are thus accessible to the user. VAT gathers the data from the corresponding data center and presents them to the user. The user is then able to process the data with the provided operators. It thus offers an important service to the GFBio portal to explore the raw data, while the portal itself only operates on metadata.

4.3 Citations and Provenance

Citations are very important for scientific work. They allow researchers to classify, comprehend and reproduce published results. They are also important for the publishers of those results, as they facilitate assessing the impact of their work. Aside from scientific results, also raw data has to be properly cited for the above reasons. Today, more and more organizations and journals encourage publishing data as a citable publication to facilitate data sharing. A recent article [8] stressed the importance of citations in scientific data management.

Our system aggregates all citations of the involved data sets. It exploits the automatically tracked workflows (Sect. 4.1) to retrieve information from all incorporated data sources. We call the combination of (1) citation, (2) license and (3) a URI (e.g. link to a landing page) provenance information. All source operators are responsible for collecting the provenance information for the outputs, given a specific input. Processing operators combine the provenance information of their inputs via a duplicate eliminating union operation. This behavior can be altered for specific operators.

The provenance information for a layer is always accessible by the user in WAVE. When the user exports a layer, a ZIP archive is created. In addition to the actual data it contains two files. One file contains the workflow representing the computation of the layer, including all applied transformations. The other contains all the provenance information of the data sets involved in the computation. This provides a comprehensible description of the result and ensures reproducibility.

4.4 Temporal Operations and Aggregation

Dynamic time-series processing based on workflows is a core feature of VAT in contrast to other GIS-like systems. Our system supports the specification of a date and time as a global temporal reference. Recall that each query consists of a workflow with a spatio-temporal context, in particular a time interval that expresses the validity. The temporal reference slices a time series result such that it only contains elements that are valid at the given point in time. This facilitates an exploratory, on-demand processing of workflows where VAT only generates the results for actually requested data.

For example, a user may add the WorldClim[8] mean annual temperature data set as a raster layer to a project. This data set is a time series that contains monthly climate variables. By means of the temporal reference the system is now able to choose the valid raster for the month of the currently selected point in time and add it to the map. Consequently, operations that include this data set also incorporate the correct raster with respect to the temporal reference. In comparison, traditional GIS oblige a user to manually add the correct raster from the data set beforehand. This is obviously a cumbersome and error-prone task.

The temporal validity of a data object is defined as an interval from a start time to an end time in which the object is incorporated into computations. The start and end times refer to a user-defined temporal reference system, e. g. the Gregorian calendar. When data with different validities are combined, data objects may have to be split into multiple items with different validities. Fig. 3 shows an example of a point-in-polygon filter, where the output is a time series with two separate objects due to different validities of the polygons.

A problem arises when users want to combine data with non-overlapping temporal validity. One example is to compare measurements from today with measurements from the same day of the last year. In order to support this important type of temporal operations, WAVE offers a temporal shift

[8] http://www.worldclim.org

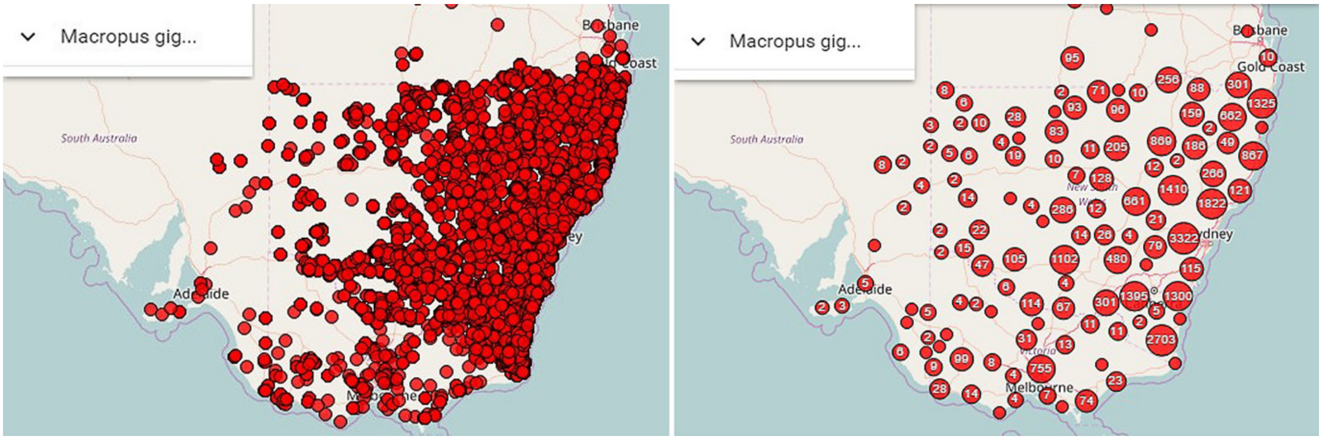


Fig. 4 Aggregating a set of kangaroo observation points in eastern Australia to improve density information

operator to change the temporal validity of objects. A shift can either be absolute or relative. It is applied to a query first before any other processing takes place. The result has to adapt its temporal validity to the temporal operation. In the previous example, after retrieving the measurements from last year, we have to set their validity to the current year in order to compare them with each other.

When a user changes the temporal reference, WAVE triggers a re-computation of all views. Thus, there will be an update of the map, the layers, the connected table and the associated plots. The incorporation of temporal functionality in the user interface is still object of our future research. We currently only allow the specification of the temporal reference. Harmonizing the validity of different data sets is not yet possible. We will perform a user study in order to find an appropriate and intuitive way to extend our user interface for such kind of operations.

4.5 Data Generalization and Exploration

There are two major objectives when interactively visualizing data for the user: One is to compute results in a near-realtime fashion for the user's exploration experience. The other is to provide an abstraction of the data that facilitates detecting interesting patterns. Data generalization can address both concerns.

The generalization of raster data is possible by aggregating multiple adjacent cells and representing them in a lower resolution. This requires less storage but comes at the expense of losing information. However, the amount of visible cells is naturally limited by the amount of pixels on the user's screen. Thus, it is sufficient to output raster images in this resolution for previews. Moreover, it is also sufficient to use source rasters and intermediate results of queries in this resolution instead of restricting the aggregation only to the results. This allows us to compute preview results with low latency. Users can afterwards trigger the computation in full resolution to produce scientifically valid results. In addition, the user can increase the accuracy of the data processing and the data visualization incrementally by simply zooming into interesting areas.

A popular approach to generalizing vector data is rasterization. This allows reducing the data to a fixed grid and makes a lot of operations much easier to compute, e. g. checking if a point is contained in a rasterized polygon. One drawback is the loss of attribute information that was attached to each feature, and another one is the loss in precision, e. g. by introducing a line width. Another approach is to apply simplification techniques that lead to fewer coordinates. Beside of being expensive, this also causes semantic changes of queries. For lines and polygons we will use standard techniques like Douglas-Peucker [17] but aim for extending them to consider topographic constraints in future work.

WAVE offers an approach to generalizing big point sets to speed up their visualization and to identify cluster patterns. Displaying each point with its associated attributes exceeds the capabilities of current browsers on modern hardware even for sizes of less than one million points. Additionally, the size of transferring the data in the GeoJSON standard format stresses the internet connection of mobile devices. An example of this are 23 039 kangaroo (Macropus giganteus) occurrence points from GBIF[9], cf. the left hand side of Fig. 4. The uncompressed size with 20 common attributes is ∼15 MB even for this relatively small data set. Furthermore, it is hard to recognize in the raw data that there is a very dense population of kangaroos in the south of Canberra. WAVE uses an adapted tree implementation of the hierarchical method developed by Jänicke, et al. [12] to aggregate point data for the purpose of visualization. This allows combining nearby data points dependent on the zoom level and map resolution. By zooming in, the user gets a

[9] Global Biodiversity Information Facility: http://www.gbif.org

smaller excerpt of the map in more detail such that clusters break up and more details reveal. We represent the clusters as non-overlapping circles with logarithmically scaled area based on the number of included points. We assessed the quality, and the advances and drawbacks of such a visualization technique, that corresponds to a novel constraint clustering problem, in previous work [7]. Additionally, the circles contain the number of points as labels, see the right-hand side of Fig. 4.

4.6 The Data Table

WAVE provides a tabular view bidirectionally linked to a selected layer of the map. Like the map itself, the table reacts to changes on the temporal reference. The table rows correspond to the visible, aggregated point data on the map. This allows us to limit the transferred data from MAPPING to the aggregated values only. Therefore, the table shows exactly the same data points as the map in the same resolution. Because of the geographic aggregation of points we also need to aggregate all other attributes of the data set. For numeric attributes, we use the mean and standard deviation that can be computed in linear time. By zooming in, the amount of points in a cluster decreases and so does the standard deviation. This means the information gets more exact by diving into the data. For textual attributes, WAVE keeps a small number of representative points (typically three to five) for each cluster. Among the many options for selecting representative points, we decided to use the points closest to the cluster center. The reason for our choice is that this information improves in accuracy when zooming in.

Some textual attributes contain links to multimedia data. An example is a hyperlink to a photo of a data set object. WAVE is able to detect multimedia links in a table cell and provides image views, or audio and video players whenever relevant to access the data at its source. Thus, users are able to investigate the different aspects of a data set directly in our system.

4.7 Plotting and R-Connectivity

Plots allow visualizing properties of data sets and correlations among attributes that are neither visible on the map nor recognizable in the data table. VAT natively provides histograms and scatter plots. They also follow our core concept of time-linkage, i. e. a change of the global time in the user interface results in a re-computation of the plots.

In addition to the native plots, we support an interface to R for the following reasons. First, experienced users are able to bring in their existing scripts, e. g. for sophisticated data statistics or to generate plots. Second, we use our R interface to exploit the powerful visual components by providing a transparent interface that makes the functionality accessible to users without knowledge of R. Thus, it is not necessary to re-implement it again.

The coupling of VAT with R is done in the following way. The MAPPING R component communicates with an RServer[10]. We use remote procedure calls to execute code, send input data and return the finished script output. Within the called procedures, we make use of RInside [10] to handle the impedance mismatch of the C++ and R data types. This is especially complex when transforming our GPU-friendly, columnar data structures to R data frames and vice versa. Furthermore, we extend the R environment by specialized functions that make it possible to access the input data delivered by VAT. The output data types are either raster or vector data, plots or simply text. For security reasons, we make use of sandboxing to prevent executing malicious and system-threatening code on VAT.

4.8 Data Export

There are basically two reasons for exporting data with VAT. The first is when the data exploration yielded interesting data with respect to the field of research and the toolbox of VAT resulted in enriched data or insightful plots for a publication. The second is when VAT's general purpose tools reached the limits and a specific modelling software is necessary to finish a research task. Both cases make it necessary to get data out of the system into popular formats. For raster data, the user can specify the data resolution of the export in order to get accurate results (in comparison to the concept of preview resolutions for fast data exploration).

Currently, VAT supports exporting vector data in CSV or GeoJSON format and raster data as GeoTIFF. The exported data is bundled together with provenance information in form of a workflow description and a list of citations (Sect. 4.3). While both are currently exported in JSON representation, we plan to extend it e. g. to BibTeX in the future. The composition of these three files form an export unit. VAT packs it into a single ZIP file for downloading.

5 Description of a Use Case in WAVE

The following use case exemplifies the capabilities of VAT. A public version of the system is available[11] and allows conducting this scenario. We consider the scientific task of discovering differences in the distribution of two tree species: the toothbrush tree (*Salvadora persica*) and the kola nut (*Cola nitida*). While the toothbrush tree is an evergreen tree or shrub growing in hot and dry conditions, the

[10] https://github.com/bgweber/RServer

[11] https://vat.gfbio.org

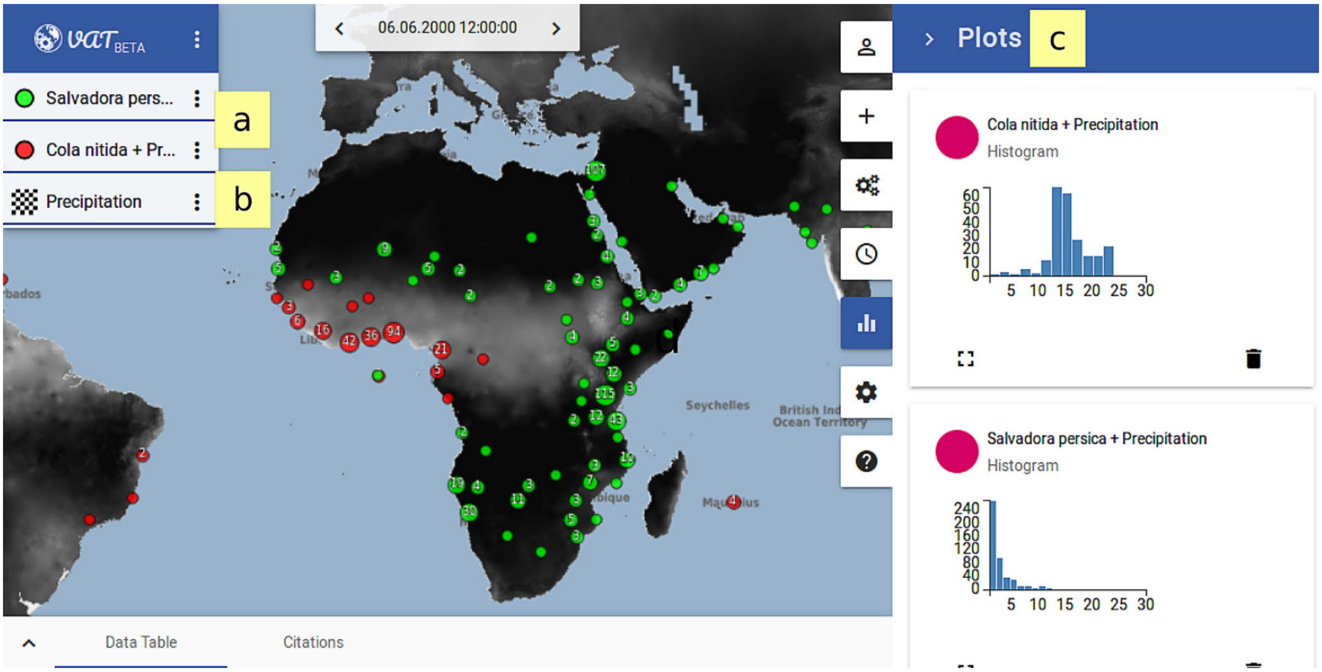


Fig. 5 The final stage of the use case where markers indicate the different steps that led to the results

kola nut is a tropical tree from West African rain-forests. The use case includes processing and visualizing a combination of raster and vector data, comparative plotting and time shifting.

In a first step occurrence data from both species is added to WAVE. Often users want to combine their own data with existing public data sets. Thus, the use case contains one user data set and two data set from one of the repositories provided by VAT. The user data in CSV representation is imported using the import wizard from WAVE's data menu. After specifying the data format, the data set is available in the private repository of the user and added as a layer to the map. Then, the species occurrence wizard from WAVE's data menu is used. Here, the kola nut is looked-up and the resulting GBIF data is added as layer to the map.

Fig. 5 shows the map with layers (a) for both species. Their distributions are matching their habitat descriptions. The kola nut occurrences are in rain-forest areas while toothbrush trees are located in hot and dry regions. To confirm the visual impression, the user adds environmental data to the project.

In the environmental data repository of WAVE's data menu, monthly precipitation data is available from the WorldClim data set. Adding this as a layer allows to assess the precipitation at the occurrence points by visual inspection. Now the *raster-value-extraction* operator from the operator menu allows to add the raster value at each occurrence location to the occurrence points. This creates a new layer where all points have an additional attribute *precipitation* that gives the average monthly precipitation at the location of the point.

Fig. 5 shows layers for both species and the precipitation (b). To compare the two data sets the histograms of the precipitation at the occurrence locations are generated using the histogram operator from WAVE's operator menu. Here, both histograms use the same configuration (e.g. a range of 0 mm to 300 mm and 20 buckets) to achieve comparable results.

Fig. 5 shows the histograms (c) of precipitation values at the occurrence locations of kola nut and toothbrush tree. We observe that kola nut occurrences are more often located in areas with high precipitation rates and toothbrush trees in dry areas. The precipitation data from WorldClim represents monthly aggregates. WAVE's time menu can be used to traverse the data month-by-month by changing the displayed point in time. Changing the reference time in WAVE, the layers on the map, the values in the data-table and the plots are re-generated automatically. For sophisticated species distribution modelling on the user's personal computer, the enriched layers are exported as CSV files using the export dialog found in the layer menu.

6 User Evaluation

To evaluate the design and usability of WAVE we conducted two user studies with biodiversity experts from the Senckenberg Biodiversity and Climate Research Institute (BiK-F) in Frankfurt am Main, Germany. The first user study

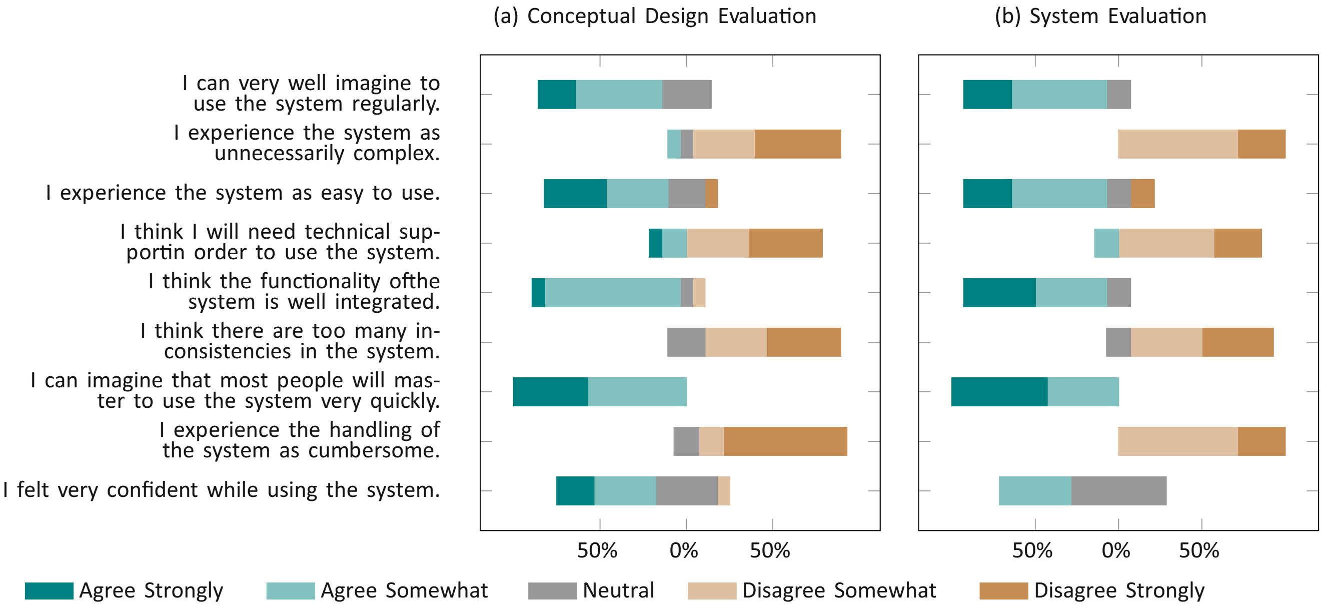


Fig. 6 These bar diagrams show the results of the two user evaluations

took place in the design phase of WAVE and was conducted to gain insights about an appropriate user interface design prior to the product development. The second user study was designed to evaluate the actual implementation of WAVE and used the use case provided in the previous section.

6.1 Conceptual Design Evaluation

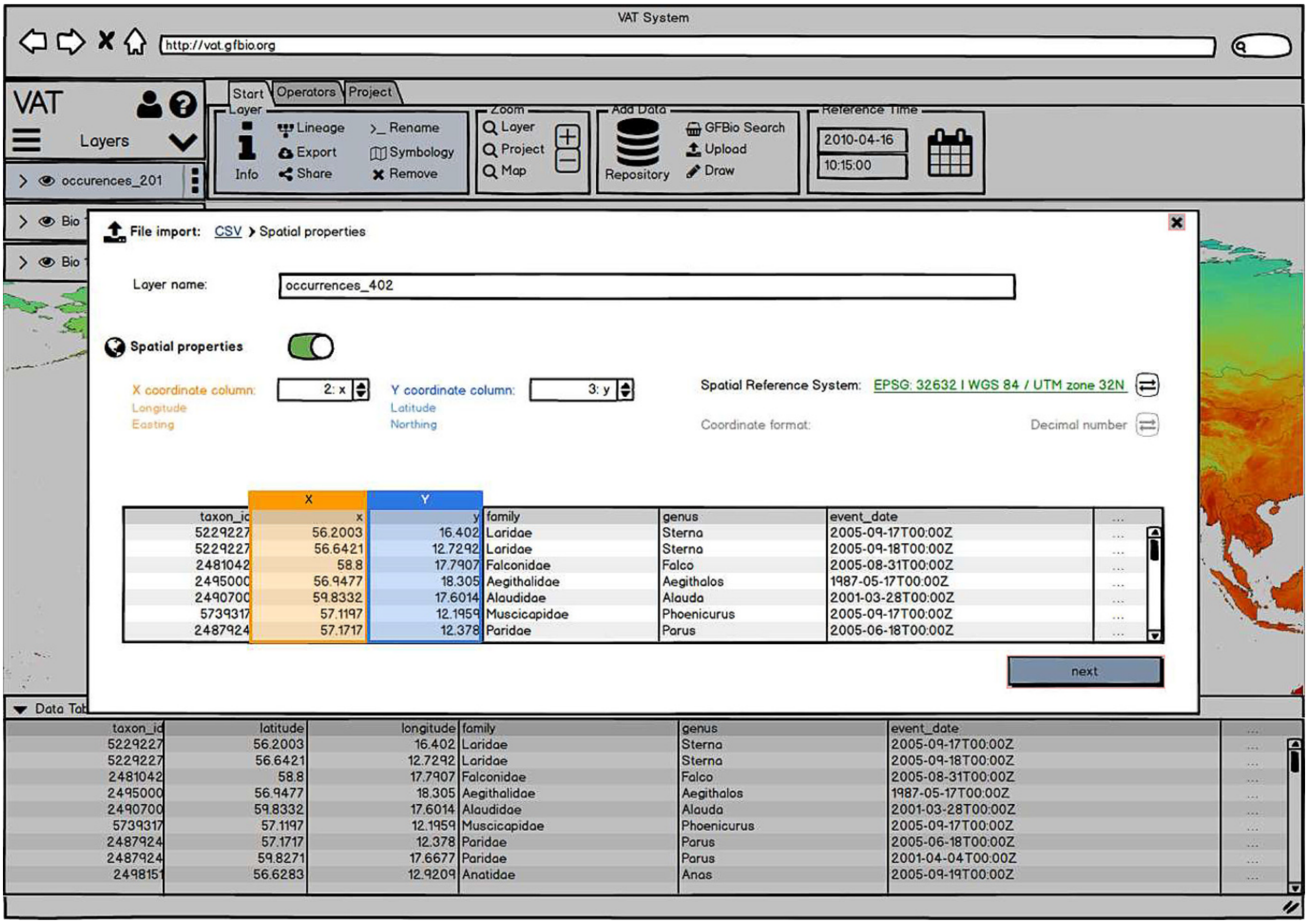
The conceptual design evaluation included 15 potential users in the design process of creating an effective user interface. For this, we developed a paper prototype (Fig. 7) which covered an almost identical use case to the one described in the previous section. The use case focuses on bird occurrences instead of trees but other than that uses the same operations. This allowed us stepping back from any implementation details and focusing on concepts on a sketch board. The advantage was that it is very inexpensive to discard doubtful concepts. And in conclusion this led to rapid concept development with domain experts.

The user study consisted of two parts. The first was an introduction of the use case and a 20 minute time span to solve a specific task. The users had to work on the task independently without any system introduction or explanation. We observed the behavior and timed steps of certain sub tasks. The second part included a questionnaire of nine fixed questions and an additional field for free text comments. The participants had ten minutes time for this feedback. The questions aimed at different impressions about the system usage. As answers we used a symmetric typical five-level Likert scale with a neutral element (Fig. 6a).

Together with the additional comments (which we excluded here for space reasons) we did not find any reason for major changes in our design proposal. Nevertheless, we identified minor weaknesses and were able to get a better understanding of how users work with our system. One interesting fact to mention was the expectation of the users to interact with the application like in desktop GIS. This included right-clicking on elements to perform actions. This was a strong contrast to our previous experience in web application development. Additionally to the feedback from the first user study we were asked by early testers to move the toolbar from the top to the side to more efficiently exploit the typical wide-screen aspect ratio of modern desktop and mobile displays (Fig. 5).

6.2 System Evaluation

The second user study included 7 participants. While the evaluation design was the same as in the first evaluation, the participants used VAT and not a mockup. The equal design allow a direct comparison of the results from both studies. First, the use case focused on toothbrush tree and kola nut was introduced (Sect. 5). Then, again, the users worked independently without introduction or explanation of WAVE to solve the tasks. In comparison to the first study, we extended the time span to 30 minutes. None of the users had previous experience with WAVE. We monitored the screens of the participants and their mouse movement. Additionally, we asked them to express their thoughts (think-aloud technique) and denoted them in a detailed fashion. After finishing their tasks, the participants filled out the same survey we used in the first evaluation to assess VAT (Fig. 6b).



File import: CSV › Spatial properties

Layer name: occurrences_402

Spatial properties

X coordinate column: 2: x
Longitude
Easting

Y coordinate column: 3: y
Latitude
Northing

Spatial Reference System: EPSG: 32632 | WGS 84 / UTM zone 32N

Coordinate format: Decimal number

	X	Y				
taxon_id	x	y	family	genus	event_date	...
5229227	56.2003	16.402	Laridae	Sterna	2005-09-17T00:00Z	...
5229227	56.6421	12.7292	Laridae	Sterna	2005-09-18T00:00Z	...
2481042	58.8	17.7907	Falconidae	Falco	2005-08-31T00:00Z	...
2495000	56.9477	18.305	Aegithalidae	Aegithalos	1987-05-17T00:00Z	...
2490700	59.8332	17.6014	Alaudidae	Alauda	2001-03-28T00:00Z	...
5739317	57.1197	12.1959	Muscicapidae	Phoenicurus	2005-09-17T00:00Z	...
2487924	57.1717	12.378	Paridae	Parus	2005-06-18T00:00Z	...

next

taxon_id	latitude	longitude	family	genus	event_date	...
5229227	56.2003	16.402	Laridae	Sterna	2005-09-17T00:00Z	...
5229227	56.6421	12.7292	Laridae	Sterna	2005-09-18T00:00Z	...
2481042	58.8	17.7907	Falconidae	Falco	2005-08-31T00:00Z	...
2495000	56.9477	18.305	Aegithalidae	Aegithalos	1987-05-17T00:00Z	...
2490700	59.8332	17.6014	Alaudidae	Alauda	2001-03-28T00:00Z	...
5739317	57.1197	12.1959	Muscicapidae	Phoenicurus	2005-09-17T00:00Z	...
2487924	57.1717	12.378	Paridae	Parus	2005-06-18T00:00Z	...
2487924	59.8271	17.6677	Paridae	Parus	2001-04-04T00:00Z	...
2498151	56.6283	12.9209	Anatidae	Anas	2005-09-19T00:00Z	...

Fig. 7 The CSV upload dialog of the paper prototype

In summary, the results were very positive. The participants identified no major issue that would have led to a conceptual change of WAVE. Minor disturbances in the workflow could be explained by the deviations from their daily used tools. The temporal functionality that is a novel feature in this domain was very much appreciated. As the provided data sets were highly appreciated, most participants asked us to extend the repository with more specialized data sets. Some users mentioned that they will most likely use VAT in the future and think that we should start providing VAT functionality for other biodiversity projects, e. g. for landing pages of data sets.

7 Conclusion and Future Work

We presented the VAT system, an application for the interactive exploration of biodiversity data. It facilitates data-driven research by combining a fast processing backend with visual analytics techniques in our web-based frontend WAVE. VAT offers several features including the management of temporal information, the efficient generalization of data and the automatic tracking of lineage information and citations. Two user studies verified the validity of our approach.

The easy access to a rich repository of geo data combined with tools for the flexible join of different data sets also provides unique opportunities for the database community. VAT and GFBio are excellent sources for benchmarking innovative geo processing techniques on real-world data.

In our future work we will extend the temporal operators and provide support for trajectory data. This will allow the detection of spatio-temporal correlations among objects and the calculation of what-if scenarios with respect to environmental variables.

Acknowledgements This work has been supported by the Deutsche Forschungsgemeinschaft (DFG) under grant no. SE 553/7-2 and by the Bundesministerium für Bildung und Forschung (BMBF) under grant no. 01LL1301.

References

1. Altintas I, Berkley C, Jaeger E, Jones MB, Ludäscher B, Mock S (2004) Kepler: An Extensible System for Design and Execution of Scientific Workflows. In: SSDBM, pp. 423–424. IEEE Computer Society. https://doi.org/10.1109/SSDM.2004.1311241
2. Andrienko N, Andrienko G, Gatalsky P (2003) Exploratory Spatio-Temporal Visualization: An Analytical Review. J Vis Lang Comput 14(6):503–541
3. Authmann C, Beilschmidt C, Drönner J, Mattig M, Seeger B (2015) Rethinking Spatial Processing in Data-Intensive Science. In: Datenbanksysteme für Business, Technologie und Web (BTW 2015) - Workshopband, 2.–3. März 2015, Hamburg, Germany. BTW Workshops, LNI, vol. 242, Bonn, Germany, pp. 161–170
4. Authmann C, Beilschmidt C, Drönner J, Mattig M, Seeger B (2015) VAT: A System for Visualizing, Analyzing and Transforming Spatial Data in Science. Datenbank-Spektrum 15(3):175–184
5. Beilschmidt C, Drönner J, Mattig M, Schmidt M, Authmann C, Niamir A, Hickler T, Seeger B (2017) Interactive Data Exploration for Geoscience. In: Datenbanksysteme für Business, Technologie und Web (BTW 2017), 17. Fachtagung des GI-Fachbereichs Datenbanken und Informationssysteme (DBIS), 6.–10. März 2017, Stuttgart, Germany, Workshopband. BTW (Workshops), LNI, vol. P-266, Bonn, Germany, pp. 117–126
6. Beilschmidt C, Drönner J, Mattig M, Seeger B (2017) VAT: A System for Data-Driven Biodiversity Research. In: EDBT, pp. 546–549. http://OpenProceedings.org
7. Beilschmidt C, Fober T, Mattig M, Seeger B (2017) Quality Measures for Visual Point Clustering in Geospatial Mapping. In: W2GIS vol. 10181. LNCS, Cham, Switzerland, pp. 153–168
8. Buneman P, Davidson S, Frew J (2016) Why Data Citation is a Computational Problem. Communications of the ACM 59(9):50–57. https://doi.org/10.1145/2893181
9. Diepenbroek M, Glöckner F, Grobe P et al (2014) Towards an Integrated Biodiversity and Ecological Research Data Management and Archiving Platform: The German Federation for the Curation of Biological Data (GFBio). In: GI-Jahrestagung, LNI, vol. 232, pp. 1711–1721. GI
10. Eddelbuettel D (2013) Seamless R and C++ Integration with Rcpp. Springer, New York, USA
11. Gebbert S, Pebesma E (2017) The GRASS GIS Temporal Framework. Int J Geogr Inf Sci 31(7):1273–1292
12. Jänicke S, Heine C, Stockmann R, Scheuermann G (2012) Comparative Visualization of Geospatial-temporal Data. In: GRAPP/IVAPP, pp. 613–625. SciTePress
13. Jetz W, McPherson J, Guralnick R (2012) Integrating Biodiversity Distribution Knowledge: Toward a global Map of Life. Trends Ecol Evol 27(3):151–159
14. McLennan M, Clark SM, McKenna F, Deelman E et al (2013) Bringing Scientific Workflow to the Masses via Pegasus and HUBZero. In: IWSG, CEUR Workshop Proceedings, vol. 993. http://CEUR-WS.org
15. Open Geospatial Consortium Inc. (2011) OpenGIS Implementation Standard for Geographic information – Simple feature access – Part 1: Common architecture
16. Roth RE (2013) Interactive maps: What we know and what we need to know. J Spatial Inf Sci 2013(6):59–115
17. Shi W, Cheung C (2006) Performance Evaluation of Line Simplification Algorithms for Vector Generalization. Cartogr J 43(1):27–44
18. Steed CA, Ricciuto DM, Shipman G et al (2013) Big Data Visual Analytics for Exploratory Earth System Simulation Analysis. In: Computers and Geosciences vol. 61. Elsevier, New York, USA, pp. 71–82
19. Wolstencroft K, Haines R, Fellows D et al (2013) The Taverna workflow suite: designing and executing workflows of Web Services on the desktop, web or in the cloud. Nucleic Acids Research **41**(Webserver-Issue), pp. 557–561
20. Zhang J, You S, Gruenwald L (2017) Towards GPU-Accelerated Web-GIS for Query-Driven Visual Exploration. In: W2GIS, *Lecture Notes in Computer* vol. 10181. Science, Cham, Switzerland, pp. 119–136