Info Vis

Article

Information Visualization
2017, Vol. 16(3) 232–256

DOI: 10.1177/1473871616667767
journals.sagepub.com/home/ivi
SAGE

A systematic view on data descriptors for the visual analysis of tabular data

Hans-Jörg Schulz[1], Thomas Nocke[2], Magnus Heitzler[3] and Heidrun Schumann[1]

Abstract

Visualization has become an important ingredient of data analysis, supporting users in exploring data and confirming hypotheses. At the beginning of a visual data analysis process, data characteristics are often assessed in an initial data profiling step. These include, for example, statistical properties of the data and information on the data's well-formedness, which can be used during the subsequent analysis to adequately parametrize views and to highlight or exclude data items. We term this information *data descriptors*, which can span such diverse aspects as the data's provenance, its storage schema, or its uncertainties. Gathered descriptors encapsulate basic knowledge about the data and can thus be used as objective starting points for the visual analysis process. In this article, we bring together these different aspects in a systematic form that describes the data itself (e.g. its content and context) and its relation to the larger data gathering and visual analysis process (e.g. its provenance and its utility). Once established in general, we further detail the concept of data descriptors specifically for tabular data as the most common form of structured data today. Finally, we utilize these data descriptors for tabular data to capture domain-specific data characteristics in the field of climate impact research. This procedure from the general concept via the concrete data type to the specific application domain effectively provides a blueprint for instantiating data descriptors for other data types and domains in the future.

Keywords

Metadata, data profiling, initial data analysis, climate impact research

Introduction

Over the last two decades, visualization has matured into an important tool for data analysis. The scientific literature encompasses a plethora of visualization techniques that support *exploratory analysis* (i.e. hypothesis generation) and *confirmatory analysis* (i.e. hypothesis testing). Yet, these visual analysis techniques require certain constraints to be met by the input data to ensure their applicability and usefulness—for example, a certain quantity and quality of data. As a given input dataset rarely carries information about these aspects beyond the raw data, comprehensive data analysis methodologies start with an *initial analysis*[1] or *data profiling*[2] that aims to assess these data characteristics before going into the exploratory or confirmatory phase. The outcomes of such an assessment are what we term *data descriptors*.

We define a data descriptor as any objective data characterization that captures data properties with a particular focus on the data's subsequent visual analysis. The objectiveness of the descriptor is of

[1]Department of Computer Science, University of Rostock, Rostock, Germany
[2]Potsdam Institute for Climate Impact Research, Potsdam, Germany
[3]ETH Zurich, Zurich, Switzerland

Corresponding author:
Hans-Jörg Schulz, Department of Computer Science, University of Rostock, Albert-Einstein-Straße 22, 18059 Rostock, Germany.
Email: hjschulz@informatik.uni-rostock.de

importance to not bias this base information on which the remainder of the visual analysis workflow rests. The term "data descriptor" was chosen to reflect this objectiveness and to set it apart from interpretive information construing the data. It thus shares the intention of similar concepts, such as *metadata* and *semantic data*.

Data descriptors explicitly encode a dataset's characteristics, such as irregularities (e.g. format violations or extreme values) and regularities (e.g. data types or constant data values). These make it possible for subsequent visual analysis techniques, to check the found irregularities against required quality constraints and to adapt their parametrization to the found regularities. A common example for the latter is the parametrization of a meaningful and effective color scale according to known properties of the data. These properties can range from simple information about the data's type[3] to its spatial frequency[4] or background knowledge about its semantics.[5,6]

Taking this knowledge about a dataset into account when visualizing the data can be vital. An impressive instance of how visualization fails, when the facts that are known about a dataset are not taken into account, was just recently given in the IEEE VIS 2016 tutorial on "Perception and Cognition for Visualization" by Bernice Rogowitz. Her example shows how a poorly chosen and inadequately parametrized color scale hides the known properties of the Higgs Boson dataset instead of showing them (see http://root.cern.ch/rainbow-color-map).

Yet, useful data properties and metrics are scattered across different levels of detail, subsuming various heterogeneous information about data and spanning different subdomains of analysis. For example, data descriptors can relate to such diverse aspects of a dataset as its provenance, its storage schema, its uncertainties, or its descriptive statistical measures. From these few examples, it is easily understandable that these aspects are rarely considered and described in concert and taken into account only as they become relevant for a particular computational analysis or visual mapping.

This article aims on one hand to bring these scattered approaches for describing data together in a systematic form. And on the other hand, it aims to illustrate how these approaches can be used to support the visual analysis process. To form such a systematic understanding of data descriptors, this article makes the following contributions:

- It gathers a wide variety of data descriptors from different fields of visual analysis in a *generic classification* that is not restricted to a particular data type or application domain.
- This classification is then *instantiated for tabular data*, which is one of the most common types of data across various application domains, and a pipeline for gathering descriptors from tabular data is presented.
- To exemplify its use, this instantiation is then *adapted for climate impact research*, which has a high demand for data descriptors due to the heterogeneity of the various involved disciplines and their diverse data standards and implementations.

This systematic view on data descriptors will provide a solid and comprehensive base for their application and further investigation. In this way, our article gives a blueprint for how descriptors for other types of data—for example, textual data, image/video data, or graph/network data—can be systematically established and adapted to their respective application domains. The structure of this article follows this overall direction from the generic to the specific, starting with the definition and classification of data descriptors in the following section.

A classification of data descriptors

To concretize our introductory remarks, we define a *data descriptor* as objective data about data that is available to a visual analysis system. *Objectivity* captures the important aspect of independence of any a priori assumptions about the data and of any preconceived goal or path of analysis. Note that this does not imply that the data itself must be objective, if it ever can be[7]—only its description. As it is hard to exactly delimit objectivity in technical terms, we use two indicators that a data description is objective: *invariance* (i.e. the same dataset stemming from the same data source will always result in the same description) and *independence* (i.e. the description only depends on the dataset and its source and no external parameters). If these indicators are fulfilled by a data description, we deem it sufficiently probable that the description is not distorted or biased by an outside influence. Finally, the description's *availability* to a visual analysis system is important, as it means that the description is machine-readable, ruling out, for example, solely verbal or diagrammatic descriptions.

While other fields have already embraced the idea of leveraging "data about data"[8]—be it out of convenience or out of necessity—the visualization community has just started to explore this direction. Examples include the Metadata Mapper[9] that utilizes data descriptors to map data between different visual analysis components, as well as the Knowledge-based Visual Analytics Reference Architecture[10] that captures

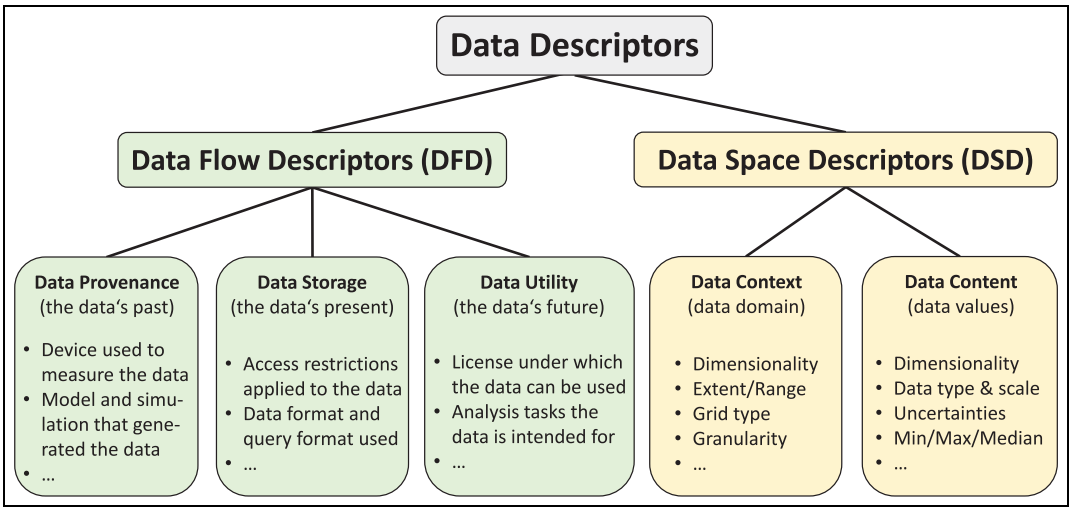

Figure 1. Our proposed classification of data descriptors by the data aspects they describe. The different classes of descriptors are arranged from general (top) to specific (bottom) and exemplified with a few common descriptors each.

analytical results (i.e. knowledge about the analyzed data and the analysis process) and feeds them back into future analyses. As applications like these make use of a few selected data descriptors to reach their particular goals, a general overview of data descriptors for visual analysis remains an open point of research.

For giving such an overview, we assume without loss of generality the described data to be *self-contained* and *homogeneous*. In cases in which this assumption does not hold—that is, for datasets that link to external data of unknown properties (not self-contained) or that are combined of a number of individual datasets of different structure and content (heterogeneous dataset) or both—descriptors can hardly be applied across the whole dataset. In these cases, the dataset can be partitioned into self-contained homogeneous subsets, then to be characterized individually by data descriptors appropriate for each of them.

For such a self-contained homogeneous dataset, our classification shown in Figure 1 categorizes data descriptors according to the aspect of the data they are describing. As a first distinction, data can either be looked at from a temporal perspective, that is, the *data flow*, or from a structural perspective, that is, the *data space*. Current literature on metadata, data properties, data models, or any related term or notion hardly ever considers data flow descriptors (DFDs) and data space descriptors (DSDs) in concert. This is most certainly due to the fact that DFDs are mainly used in database applications and information management scenarios, whereas DSDs are mainly used in data analysis and data mining approaches. Yet, describing these two aspects of data together is common in other fields—for example, for describing multimedia not only content-wise, but also in terms of who produced it and for which audience,[11] or for describing web-based resources together with their utility if limited by legal or other conditions.[12] Hence, the following sections give examples of data descriptors for both aspects, their common notations and models, as well as how they are used for visual analysis purposes.

DFDs

DFDs give details about where the data came from (*data provenance*), where it is now (*data storage*), and where it can go from there (*data utility*). Data provenance information can range from a simple model number and firmware version of the device used to record the data to a full-fledged protocol of all analysis and data processing steps it has already undergone. Information about the current storage of the data captures mainly if and how the data can be retrieved and thus be used in its current state. Such information can include, for example, details on the database schema (i.e. across how many tables does the data spread) or to how many queries per minutes the server restricts the access. Finally, the data utility information can detail the uses for which a dataset is licensed or it can specify for which purpose the data were collected.

Apart from a few papers contributing to the current research challenge of data provenance, literature from the visualization community is rather sparse on DFDs. The reason for this may be that DFDs are often less formal than DSDs or they make use of non-standard notations in which the description is given. Due to the diversity of data, standards for its description exist mainly for domain-specific cases, such as the ISO

standard 19115-1:2014[13] for geographic information or the full-metadata format (FMF)[14] for data from scientific experiments. The following sections highlight data descriptors that have been proposed for the data aspects of provenance, storage, and utility.

Data provenance. Since data are ubiquitous in these days, it is not uncommon anymore to have at least minimal provenance information indicating who authored or curated a dataset and which version of the dataset one is looking at. With more knowledge about the origin and lineage of data, it may also be possible to judge the trustworthiness of the data and of the processes that generated it,[15] or even to independently reproduce the data.[16]

A number of taxonomies for provenance descriptors exist. The taxonomy which captures the widest range of provenance aspects is the one by Simmhan et al.[17,18] It contains not only such essential aspects as the data and the process that has produced it (*subject of provenance*) but also such diverse descriptors as the representation, the storage, the dissemination, and the use of the provenance information. Other taxonomies tend to focus more on the conceptual and technical aspects of collecting and managing provenance information—such as the ones by Glavic and Dittrich[19] and by Da Cruz et al.[20] Regardless of their particular focus, each of these taxonomies can be thought of as a logical continuation and further subdivision of the class of provenance descriptors in Figure 1.

For recording and storing provenance information, either models from the database community[21] or from the domain of scientific workflows[22,23] are more suitable—depending on whether the provenance information captures a series of data states or the sequence of processing steps that produced them. For interchangeability of provenance information beyond the data and software ecosystems of a particular domain, the Open Provenance Model[24] has been developed.

In visual analysis, provenance is still largely an open research challenge.[25] Research results in this direction deal with capturing either the generation of the visualization—that is, the visualization process itself,[26,27] or the interaction history with the generated visualization—that is, the knowledge discovery process.[28,29] Along the lines of the current survey by Ragan et al.,[30] these two strategies can be called "follow the data" and "follow the user," respectively. As an outcome of both, the generated visualization or the discovered knowledge can then be annotated with the information about how they were yielded. In particular in highly exploratory scenarios that are characterized by a frequent back and forth in which many different alternatives are tried, provenance information can become quite large and unwieldy. In these cases, the visual exploration of this provenance information poses a challenge in itself that is addressed by dedicated tools like *Tableau Behavior Graphs*[29] or *AVOCADO*.[31]

Data storage. For retrieving and querying data, information about its storage is needed. This information does not only entail descriptions of the data's organization, such as the data structure or data schema, but also about the access mechanism with which to read and manipulate the data.

Descriptors that commonly hold such information can be given for different aspects of a data storage—for example, for a logical or for a physical perspective,[32] as well as for stored data or for stored (business) rules/processes.[32,33] The latter include descriptors that detail the behavior of a data storage through usage statistics and security settings, which can help to optimize data access patterns or to understand access limitations, respectively.

Notations for information about the storage of a dataset are clearly centered on describing the logical model of data organization, with the relational model describing data as tables being one of the most prominent examples.[34,35] Other models include graph-based descriptions of data organization[36] or data being organized in a multi-dimensional space.[37,38] As most models come with their own notation, more abstract metamodels, like *information spaces*,[39] and a diverse set of standards, like the ISO/IEC 10027:1990[40] Information Resource Dictionary System (IRDS) or the W3C[41] Resource Description Framework (RDF) have been developed. They can be used to describe the specifics of different forms of data organization in a uniform way.

In visualization, information about the data storage is used in some cases for finding correspondences between data items and using them for visual highlighting[42] or for visual linking[43] in multiple coordinated views. Other authors utilize RDF-encoded information about the data organization to automatically establish mappings of data attributes to visual attributes without prior knowledge of the data sources and their schemas.[44] While these approaches all work on data item level, others use information about datasets as a whole to visualize the landscape of all datasets in a particular data repository. For example, descriptors containing information about each dataset's size and server speeds can be used to graph an entire such *data landscape* to make an informed decision about which dataset to use for an analysis at hand.[45]

Data utility. The utility of data, that is, its intended and imaginable uses—which may not be the same, is rarely considered in the literature. Some of the data's utility (or lack thereof) may be inferred from its provenance, as for example outdated financial data cannot be used as a basis for day trading or stale patient records cannot be utilized to plan a medical procedure. Other utility aspects may be inferred from storage descriptors, as for example many database servers limit the number of queries per minute, which can severely hamper its usefulness for query-intensive analyses.

To the best of our knowledge, no list or taxonomy of data utility descriptors exists. Apart from legal constraints that might limit the use of a dataset (e.g. its license or confidentiality regulations), most of the literature on data utility relates to anonymized and/or obfuscated data.[46] Depending on which methods were used for anonymization, the utility of the data may be limited to certain kinds of analyses. For statistical obfuscation methods (so-called *statistical disclosure limitations*), metrics exist to measure the remaining utility of the data to find a reasonable trade-off between anonymity and usefulness.[47] To the best of our knowledge, no such metrics exist for technical methods that have either introduced the utility limitation as an intended outcome[48] or as unintended by-product—for example, when data compression disrupts data properties.[49] In these cases, the used method should be included with the provenance descriptor, so that while data utility cannot be automatically quantified, the user can at least be informed about them.

At this point, the notion of *data utility* is not widespread and the decision of which dataset to use for a particular analysis or which analysis to perform on a given dataset is left to the analyst. Hence, a common notation or model for data utility to store such information alongside the data remains an open research challenge.

In visualization, privacy preservance is mostly reflected by methods that aim to provide a given level of anonymity in the resulting visualization, measured through screen-space privacy metrics.[50] Since there exist no standards for utility descriptors, visual analysis methods do not make use of them or adhere to them. The few visualization approaches, which aim at capturing some notion of utility, apply pragmatic models that link the available datasets with those visual and analytical techniques that are deemed appropriate for them by an expert user.[51] This overall lack of concern with issues of data utility stands in contrast to the early observation in visualization that the *functional role of data*—that is, its use—is a key data characteristic.[52]

DSDs

DSDs give details about the data domain (*data context*) and the data values therein (*data content*). Properties of the data context describe aspects of the space in which the data were gathered or observed, as this is important, for example, to relate the data items to each other. Common instances are the observation space's dimensionality (e.g. two-dimensional (2D)—Lat/Lon, three-dimensional (3D)—Lat/Lon/Alt, four-dimensional (4D)—Lat/Lon/Alt/Time), whether the data are scattered or gridded, and in case of the latter whether the grid type is structured or unstructured. Descriptors of the data content include not only properties, such as data type (e.g. scalar or vector) or min/max values, but also information about missing data or data that are affected by uncertainty.

Since the characteristics of the data space are of utmost importance for correct analysis and visualization of a dataset, they have been extensively investigated from the very beginning of visualization research. There exist a few descriptors that apply to both aspects of the data space—data context and data content—in the same manner. Examples of such descriptors are dimensionality of the data domain (context) and of the data values (content), as well as the scale type of each dimension (e.g. nominal, categorical, ordinal, or interval).[53,54] The following sections highlight data descriptors that are specific to either data context or data content.

Data context. The data context (often also called *data domain*, *independent variables*, or *primary key*) denotes the part of the data that specifies the frame of reference of the data values. Since the frame of reference is spanned via n axes in space, in time, or in some abstract space of identifiers, the data context can be understood as an n-dimensional space. A particular n-tuple specifying a point within that space forms the data context for its associated data content, that is, the gathered data values or data characteristics.[55] Knowledge about the data context is essential to determine, for example, the data's completeness—that is, whether a data entry exists for all identifiers or locations.

One of the first characterizations of the data context was given by Zhou and Feiner[52] under the term *data domain* and in particular *data domain entity*, which can be anything unique from a person to a point in time and space at which a measurement was taken. This generalizes other such notions, like the distinction between *coordinates* and *amounts*,[54] or between the data types *1D, 2D, 3D, temporal*.[56] These data domain entities can have a *point-wise*, *local*, or *global* extent for which they are deemed valid.[57] For example, a given

point in space could not only stand for this particular point, but for its local neighborhood as well. Finally, the characterization by Zhou and Feiner[52] also defines *data relations* that can be used to describe a structure underlying the data domain, such as a grid or a multilevel topology. This is in line with the concept of *relations* by Roth and Mattis[54] and with the data types *network* and *tree* from Shneiderman.[56] Some theories order these characteristics of the data context in layers that are hidden underneath the data content and only visible to the professional user.[58]

The characterization of the data context is probably the most influential in visualization research, as one prominent distinction between Information Visualization and Scientific Visualization is whether the data are spatially referenced.[59] Yet nowadays already 60%–80% of the data are geospatially referenced[60]—including document and image collections, whose depiction is usually considered to be part of information visualization. Hence, this common demarcation line is hard to uphold as most data are somehow spatially referenced. As a result, the distinction between scattered and gridded data becomes of increasing importance as a more meaningful characterization of information visualization and scientific visualization, respectively. Furthermore, the data context may indicate how to partition the data in a meaningful way, which can have repercussions all the way to the storage level (cp. OLAP).

Data content. The data content (often also called *attribute space* or *dependent variables*) denotes the part of the data that specifies the actual (gathered) data values. It is for this part of the data, for which probably the most descriptors exist and for which the border between descriptions from an initial analysis phase and analytical results from later analysis phases is the most blurred. For example, some literature considers clustering results as an inherent characteristic of the data content. Yet, requiring independence as part of the descriptors' objectivity forbids to consider it as such, as clustering depends on a number of subjective assumptions, such as a similarity threshold or even a predefined number of clusters (cf. *k*-means clustering).

Purely descriptive properties of data content are, for example, the types of each data attribute. Abstract distinctions differentiate the data type merely as being *atomic* or *composite*,[52] while more concrete descriptions in common software packages, such as OpenDX, distinguish further between *scalar, vector, matrix*, and *tensor* data. Another common data content descriptor relates to the quality of the data, which subsumes a whole range of dirty data properties, as they are surveyed by Kim et al.,[61] Oliveira et al.,[62] and Gschwandtner et al.[63] This includes the overall *data quality*[64,65] and in particular the *data's uncertainty*[66,67] that plays the most prominent role in visualization besides *missing data, unusable data*, or *undefined data*. On top of these given properties, it is common to derive further descriptors that can be computed without being biased by user parametrization—for example, descriptive statistics.[68]

Notations for data content descriptors exist only partially in some data formats, such as the NetCDF format that will be discussed in further detail in the section on data from climate impact research. Only for the subset of data quality descriptors, specialized notations can be found. Among them are the ISO/IEC standard 25012:2008,[69] as well as a proposal for an extension to the Business Process Model and Notation (BPMN) to encode data quality requirements.[70]

A specific focus in the visualization community lies on representing data quality in general[71,72] and data uncertainty in particular, as communicating the data's trustworthiness is of essence when basing a visual analysis on it. Specifically for the challenge of uncertainty visualization, a number of extensive overview articles provide a good outline of the massive corpus of literature on this topic.[73–80] Furthermore, the problem of visualizing missing data is frequently singled out as a particular challenge, since it is unclear how to show something that is not present. Notable approaches in this direction include *missing value charts* by Theus et al.,[81] *shadow plots* by Swayne and Buja,[82] *missingness profile plots* by Fernstad and Glen,[83] and *missingness maps* by Cheng et al.[84] The opposite of missing data, namely, duplicate data entries, is addressed by visual analytics tools, such as *D-Dupe*[85,86] and *GeoDDupe*.[87]

Notations to describe the data space including data context and data content are, for example, the framework of Galhardas et al.[88] that is based on first-order logic, as well as the *E-notation* by Brodlie[89] and its extension into the *domino notation*.[90] Older variants are the fiber bundle-based notation by Butler and Pendley[91] and the *L-notation* by Bergeron and Grinstein.[92] File formats with metadata capabilities are, for example, CDF, HDF, NetCDF, XDF, or XSIL. DSDs like these are sometimes used to classify visual mappings—for example, as it was done by Rankin[93] or Brodlie.[89]

Gathering data descriptors

The process of gathering data descriptors is not necessarily straightforward, as there exist at least three different sources for descriptors, which may differ in the required effort and their attainable reliability and objectivity. The first source for a descriptor is to *query* it from the data source, if it has been stored alongside

the dataset, for example, as annotation or supplemental material. The second source is to *derive* a descriptor by computing it from the dataset or by inferring it from other descriptors—that is, inferring data utility from data provenance. The third source is the users with their background knowledge about the data, who can be prompted for *input* to specify a descriptor. On top of these basic mechanisms, combinations can be employed. A common combination is that a descriptor, which has been determined once through a computation or a user input, is then stored as an annotation to the dataset, so that it does not need to be recomputed or re-entered, but can be queried directly from the data source in the future.

Tool support for gathering generic data descriptors is rare. The gathering of DSDs is (if at all) only supported as a step in a larger process—for example, for performing automated data quality assessment in *Profiler*[94] or for generating automated previews of datasets in *AutoVis*.[95] These tools understand the descriptors they compute as means toward a particular end and do not allow to uncouple them from the process in which they are embedded, even though they could be beneficial for other purposes as well. The only tools that are geared toward gathering generic data descriptors are designed for capturing data provenance and generally motivated by the goal of traceable and reproducible data analysis. Notable examples for such tools include the well-known frameworks for scientific workflow management *Karma*,[96] *Kepler*,[97] and *Taverna*.[98] In the field of visual analysis, the most advanced provenance management is currently available from dedicated frameworks that capture the visualization process, such as *VisTrails*.[27] Visualization tools can utilize VisTrails' features by integrating it through a common API.[99] Approaches that also aim to capture computational processing steps may be able to extract the provenance information from the analytical scripts being run on the data.[100] The most comprehensive approach would be to use a system-wide capturing that spans different applications, as it is envisioned by *Glass Box*.[101,102]

After having established the fundamental notions of data descriptor classes and the different ways of gathering them, we make use of these concepts to compile data-type-specific descriptors for tabular data in the following section.

Data descriptors for tabular data

This section concretizes our concept of data descriptors by taking a closer look at its concrete instantiation for tabular data. That includes the different descriptors such data entails, as well as methods to gather these descriptors if they are not supplied with the data.

Tabular data encompasses the overwhelming amount of data available in CSV files, spreadsheets, and relational databases. We assume tabular data to be given in the form of a single table (dataset) of rows (records), columns (variables), and cells (individual data items), which can be likened to Codd's third normal form.[103] Non-tabular data are often first transformed into a table, before being visualized. This is embodied in the first step of the visualization pipeline by Card et al.[104] that performs a data transformation from raw data into data tables. More complex settings of multiple tables that are linked via foreign key relations can be broken down into this canonical form. As the different parts of the data correspond to sets of different cardinality—that is, singleton (cell), tuple (row), multiset/bag (column), full dataset (table)—we call these different aspects of tabular data *granularities*.

DFDs for tabular data

As a first observation, we note that all DFDs are applicable to all four granularities. For example, the data can have provenance information attached to individual values, to individual records, to individual variables, or to the entire table. While data provenance descriptors and data utility descriptors are conceptually independent of the kind of dataset they describe, technical particularities of the described dataset still require some adaptation. For example, there can be subtle differences depending on whether the provenance relates to relational databases and SQL queries[105] or spreadsheets and embedded formulas.[106] These finer differences are usually captured by data storage descriptors that detail how to access the data, which is obviously different for relational databases and spreadsheets. For data having been stored from a spreadsheet in a CSV format, this could be whether the file is comma-separated or tab-separated, and how many lines of table header it contains. For data stored in relational database systems, this could be the schema of a table, as it is detailed by the SHOW COLUMN FROM table statement in SQL. The output of this statement also gives a number of DSDs, as they are discussed next.

DSDs for tabular data

DSDs are not only much more specific to tabular data than DFDs, but they also apply mostly to its specific granularities. We list a number of typical DSDs for tabular data in Table 1. Note that this listing does not list uncommon usage of descriptors. For example, principally it is possible to have names for records or even individual cells and there certainly exist scenarios

Table 1. Data space descriptors for tabular data.

Data space descriptors	Data context	Data content
Granularity		
Value (cell)	Contextual uncertainty (e.g. uncertain position or time point)	Uncertainty of measurement, calculation, or simulation Type of value (regular, missing, undefined)
Record (row)	Context outlier Neighborhood (e.g. connected records via grid structure)	Content outlier Topological feature (e.g. critical point)
Variable (column)	Variable name Scale type (e.g. nominal, ordinal, interval) Unit and valid range Extent (point, local, global) Type of dimensions (spatial, temporal, identifier, other)	Variable name Scale type (e.g. nominal, ordinal, interval) Data type (scalar, vector, matrix, tensor) Unit and valid range Univariate statistical measures (e.g. min, max, mean, skewedness) Spatial/temporal pattern (e.g. constancy, monotonicity, periodicity)
Dataset (table)	Kind of space (e.g. Euclidean) Spatial dimensionality Variable relations (e.g. day + month + year, first name + surname) Grid type (structured, unstructured) Variable combinations that form unique keys	Multivariate statistical measures (e.g. correlation, principal components)

in which this is desirable—yet, it is not very common, so we list the descriptor "name" only for variables. Table 1 contains those general DSDs that also apply to tabular data and adds some data-type-specific descriptors. The descriptors in Table 1, which were not mentioned in the previous section, are as follows:

Unit and valid range. Given a variable's unit of measurement, we can imply various other properties, such as the variable's semantics (e.g. degree Celsius indicates a temperature measurement) or its valid value range (e.g. values in degree Celsius cannot be lower than −273, 15°C).
Spatial/temporal continuity. If the data context provides a spatial and/or temporal frame of reference for the data content, certain continuities among the data values of a variable may emerge. For example, we can objectively evaluate if the data values of a variable are monotonically increasing/decreasing with time or spatial distance and express this as a descriptor of that variable.
Variable combinations that form unique keys. Besides the variables, which are explicitly denoted to be keys or IDs, a dataset may contain other combinations of variables that uniquely identify the records. Mechanisms, such as *primary key analysis*,[107] can be used to find such alternative identifiers. Yet often the most interesting case is when a variable combination that is expected to be unique, turns out not to be, indicating inconsistencies in the dataset.

These descriptors are typically associated with a single granularity. Yet, this association is not necessarily exclusive. For example, univariate statistical measures, such as min/max/median, can be either descriptors of the variable (column) for which they have been computed or descriptors of the individual value (cell) that constitutes the identified min/max/median. Furthermore, there exist descriptors that can be applied to all granularities and which we did not place in Table 1 for this reason. Important examples include the following:

Number/size. Everything in a dataset can be counted or measured in terms of its memory footprint. This can be of interest as an information about the data, but also point to errors in the dataset.
Duplications. Certain values and records, but also entire variables or datasets, can be identical or close to identical, which can make them of higher or lower interest to a user. While a numerical value appearing twice somewhere within the dataset does not seem like a notable occurrence, this is certainly different if the value is a person's name or a supposedly unique identifier.
Inconsistencies/mismatches. Data descriptors can be erroneous as well. For example, the dataset may have changed since the descriptors were stored, or the descriptors describe the dataset in a prototypical ideal way, but the actual data are messy and incomplete. In both cases, it is important to check if the descriptors match the data and to annotate those parts of the data that do not.

Out of this collection of descriptors for tabular data, only a few can be stored together with the data in the common file and database formats. The CSV format typically contains the variable name and sometimes also the scale type in the file header. Whereas relational databases keep these descriptors together with additional schema information in separate tables. Other than these basic descriptors are rarely given in the standard storage formats, even though it would be useful to have advanced descriptors available to bootstrap subsequent visual analysis steps. In this situation, we can gather further descriptors given appropriate tool support.

Gathering data descriptors for tabular data

When gathering data descriptors for variables, records, and the entire dataset, dependencies between them have to be taken into account and—if possible—to be resolved. This can hardly be achieved in a purely autonomous preprocess that runs without user intervention. Instead, this gathering process requires a well-defined workflow that combines computation for those descriptors that can be automatically derived with user interaction for those that rely on the background knowledge of the user. Such a semi-automated gathering of data descriptors poses the challenge that automated computations must be confined to acceptable runtimes so that users do not get frustrated waiting for intermediary prompts for their input.

To address these points, we propose a series of guidelines for gathering descriptors:

- The gathering process should *follow a stepwise gathering procedure*. This allows for running the individual gathering steps in a configurable order. That order can be defined so that it minimizes repeated calculations and user inputs. Cyclic dependencies between the data granularities—for example, certain record descriptors requiring column descriptors and vice versa—can be resolved by breaking up the gathering of descriptors for a granularity into multiple steps.
- The gathering process should allow for *different degrees of interaction*. By also providing the possibility for user inputs and manual adjustments, the process can on one hand adapt to "dirtier" data with many inconsistencies that might not lend itself to automatic descriptor acquisition. On the other hand, it can also benefit from the users' additional background knowledge about the data. Note that together with a stepwise procedure, the degree of interaction can be different for the individual gathering steps.
- The gathering process should adhere to given *time frames/speed constraints*. As the computation of some descriptors is computationally expensive, it is important to be able to limit the necessary amount of time as needed. This limit can be observed by first querying all descriptors that are already available from the dataset itself, then computing only the very essential descriptors that are still missing, before finally gathering more "advanced" descriptors—for example, running an automatic key analysis on a large number of variables where many possible variable combinations have to be checked. Due to the stepwise gathering procedure, the gathering can be stopped at each step along the process, when a predefined time limit is reached.

We have developed a software tool for gathering descriptors from tabular data that implements these guidelines. It is driven by the idea of providing explicit information about the dataset that allows for gaining first insights and deciding on visualization possibilities. In this sense, it is conceptually different from existing software tools for assessing the data quality and improving it through data cleaning. This is often termed *data wrangling*[108] and a number of software tools have been developed to help with it—for example, *AJAX*,[109] *Potter's wheel*,[110] *Wrangler*,[111] or *Profiler*.[94] These tools aim to produce data that are consistently formatted and sufficiently complete for subsequent visual analysis steps. Whereas, our tool aims at gathering information about the data that can be used in the subsequent visual analysis to decide what to view (selection of interest) and how to view it (parametrization of the representation), as it is discussed in a later section. The procedure used by our tool is shown in Figure 2 and the corresponding user interface is depicted in the screenshot in Figure 3. It adopts the stepwise gathering approach, which is detailed in the following.

Variable descriptors I. First and foremost, the process gathers information about the data types stored in each column as variable descriptors. This is the most basic information to gather, as it does not require any other information about the data. Our acquisition algorithm runs over all values per column and determines the type of the majority of entries using heuristics that match digits, delimiters, and characters and assigns fitting data types. Note that this step cannot be fully automated, as the algorithm can discern nominal variables from discrete numerical variables, but cannot detect ordinal data types. If there exists an ordering among nominal data values, it needs to be interactively

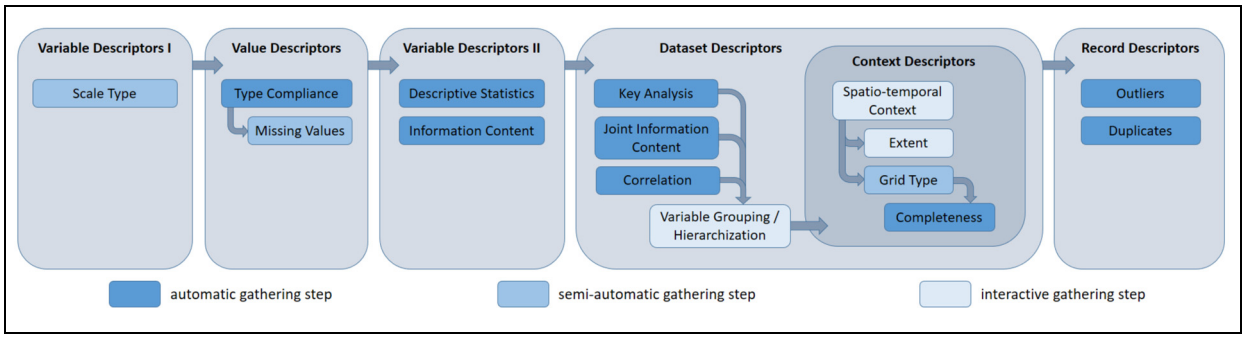

Figure 2. Steps for gathering descriptors from tabular data shown in the order in which our approach determines them. Arrows indicate dependencies between these steps and different shades of blue denote different levels of interactivity.

specified by the user who has the appropriate domain knowledge and can thus redefine the variable into ordinal data type. This makes the gathering of the data type descriptor a semi-automatic step. As this first block of variable descriptors consists only of this single gathering step, our tool combines its interface (Figure 3(a)) with the interface for the following gathering steps of value descriptors (Figure 3(b)).

Value descriptors. Once the type information is known, we can gather value descriptors. In particular, we aim to describe type compliance and missing values. The former can be done automatically by checking against the data type having been gathered for each variable in the previous step. If a value is not compliant, a corresponding descriptor will be added to that value's table cell.

Determining the missing values requires some user input, as "missing" does not necessarily mean that the cell is empty, but it could also hold a placeholder value that is out of range, such as UINT_MAX (4294967295, 0xffffffff). When such a placeholder value exists and it is type compliant, which we check first, only the users with their background knowledge about the data can decide whether this is a realistic value or a placeholder for a missing value. This makes this gathering step a semi-automatic one.

Variable descriptors II. In this next step, we gather more variable descriptors, such as descriptive statistics and each variable's information content. For these descriptors to be meaningful, it was important to mark down the non-compliant and missing values first, so that, for example, a UINT_MAX placeholder does not skew the computation of extreme values and distributions. These descriptive statistics can be computed automatically, since the variable types are known from the very first gathering step and appropriate statistics can thus be computed—for example, the mean for continuous numerical variables, the median for discrete numerical variables, and the most frequent term for nominal variables. As these statistical metrics are required by many of the following gathering steps, we placed their computation as early as possible in the process.

On top of these standard measures, we also determine each variable's information content by computing the Shannon entropy over the set of all values per variable. These can be used to identify and eliminate variables that contain a constant value throughout, which is surprisingly common in practice. Both statistical descriptors and information content are displayed for the users to inspect in Figure 3(c). Note that for convenience reasons, this table also contains the data type from the first gathering step.

Dataset descriptors. In this step, the first three descriptors have the common goal of providing information for discriminating between data context and data content. To this end, we run a primary key analysis, determine the joint information content of n-tuples of variables, and compute the bivariate Pearson correlation between all pairs of variables. These descriptors are then presented to the users who interactively specify data context and data content based on them. They can furthermore define hierarchies or groupings of associated variables, as it is suggested by Robertson[112]—for example, grouping two variables "first name" and "surname" into "name."

Once specified, we gather additional descriptors for the data context. At first, the users are asked to interactively specify those context variables that denote a spatial and/or temporal frame of reference. In addition, for each such reference, the users can define its extent—that is, point, local, or global—which is important for a subsequent visualization, as it tells whether it is possible to interpolate between the data or not. The procedure then tries to establish the data's grid type (structured or unstructured) by checking the spatial and temporal references for equidistance across

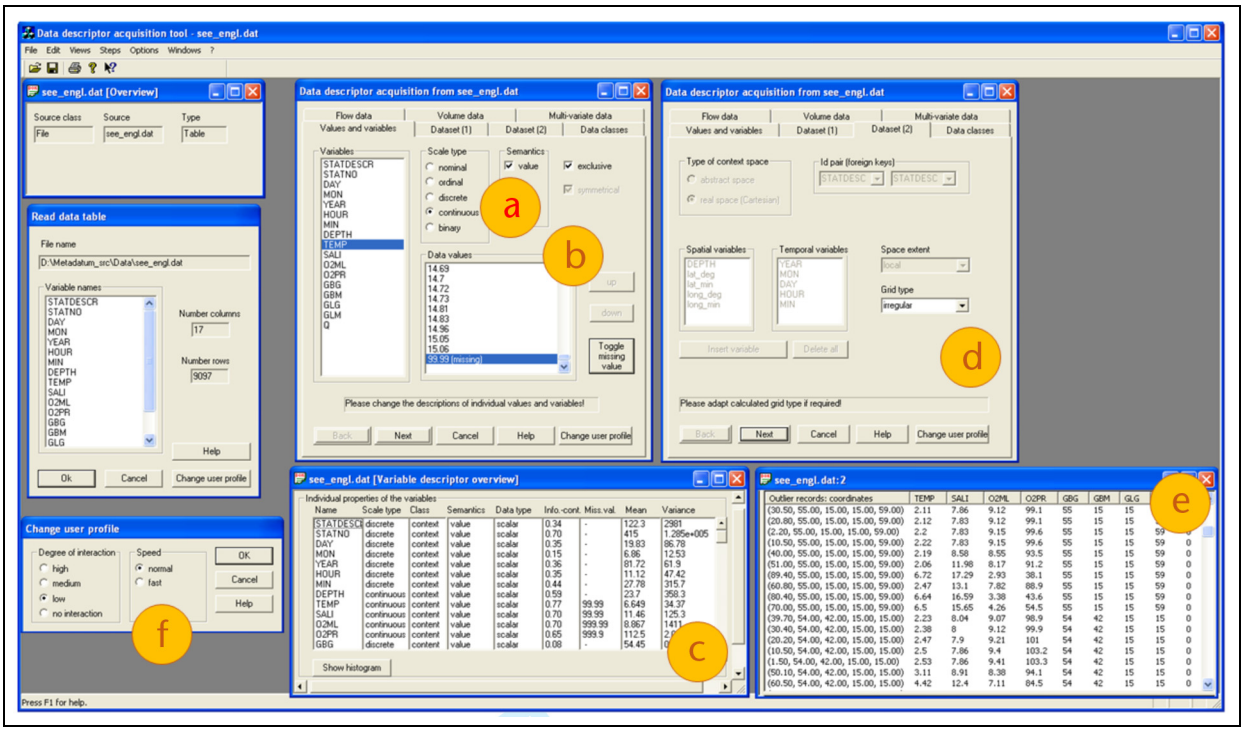

Figure 3. Screenshot from our data descriptor gathering tool for a tabular example dataset containing marine measurements conducted by the Leibniz Institute for Baltic Sea Research. Besides the generic dialogs for loading a dataset and getting a first overview in the top left, the screenshot features the interfaces for each of the individual gathering steps outlined in Figure 2: (a) variable descriptors I (data type), (b) value descriptors (type compliance, missing values), (c) variable descriptors II (descriptive statistics, information content), (d) dataset descriptors (variable groupings, spatial/temporal context, extent, grid type), and (e) record descriptors (outliers, duplicates). In addition, (f) shows the dialog that allows for adjusting the computational profile.

records. Since this heuristic is easily misled by a single outlier or undetected header row, we follow a semi-automatic approach and present its results to the users for validation and correction. Finally, if we have determined a regularly structured grid, we automatically check for completeness—that is, whether there is a corresponding data record for all possible grid positions. While completeness would be a record descriptor by the data granularity it describes, it still is treated as a dataset descriptor because the records it describes are missing and thus cannot be marked as such. This is different from missing individual values, as they were determined in the beginning, for which an empty table cell exists to which to attach the corresponding descriptor. The interface for these descriptors is spread over two tabs (see Figure 3(d))—the first tab accommodates the descriptors that discern between data context and content and the second tab holds the dataset's context descriptors.

Record descriptors. This last step gathers information about outlier records and duplicate records in the dataset. Outlier records are determined automatically by statistical means.[113] As for duplicates, we automatically check for so-called *inconsistent duplicates* that contain different information, but refer to the same entity—that is, the same customer listed twice under different addresses.[62] Figure 3(e) shows found outliers in a table view for the user to inspect.

In the standard configuration of our tool, we go through these steps trying to automatically gather as many descriptors as possible, while asking the user for input only as much as necessary. In accordance with our guidelines, we also provide other degrees of interaction from "no interaction" (automatically compute as many descriptors as possible and leave out the rest) to a "high degree of interaction" (report all automatically derived descriptors to the user for validation and readjustment). Furthermore, we also have two different computational profiles—"normal" and "fast." The "normal" mode follows the default prioritization of automated gathering through computation for as many as descriptors as possible and an interactive gathering for the rest. In contrast, the "fast" mode tries first to query stored descriptors from the data source itself. For those descriptors that are not available from the data source, the system automatically performs

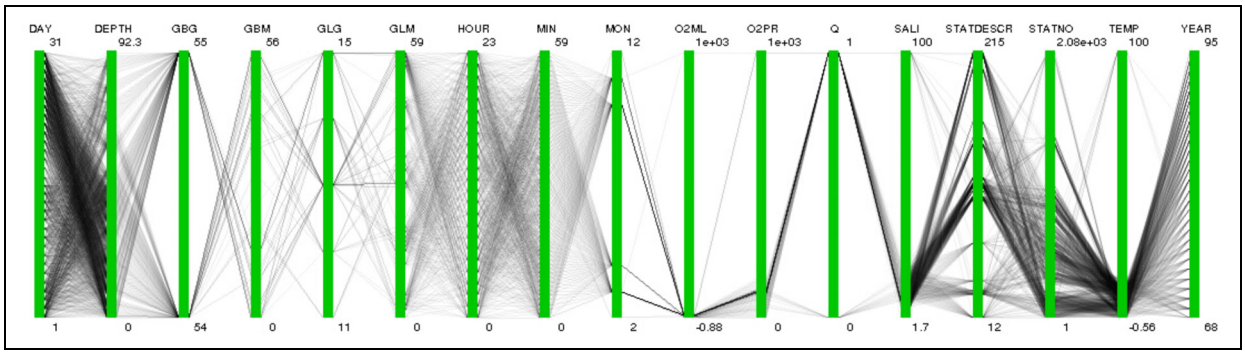

Figure 4. Parallel coordinates view of the raw marine dataset from Figure 3 without taking descriptors into account.

computationally inexpensive gathering steps (e.g. descriptive statistics or correlations) and asks the user for input on computationally expensive ones (e.g. key analysis). Note that the fast mode comes with the price of possibly having incorrect descriptors, as the ones accompanying the dataset may be outdated and the ones entered by an inexperienced user may be incorrect. Hence, the fast mode is best used by an experienced user on trustworthy data sources that are known to provide valid descriptors. Both parametrizations of the gathering process are shown in Figure 3(f).

Our software for gathering data descriptors was designed to be a general-purpose tool for tabular data about which nothing more than its tabular nature is known or assumed. The tool can be extended to discern more specific data types that follow known formats and value ranges, which can be exploited for their detection. For example, country codes could easily be detected and used accordingly in a geographic mapping, as it is done by recent versions of Tableau and MS Excel.

Leveraging data descriptors for visualizing tabular data

On one hand, the gathered data descriptors can be visualized themselves to graphically communicate high-level information about the dataset, which is mainly used for subset selection. For example, Dos Santos and Brodlie[114] introduced visual displays for certain data descriptors, which were designed for providing easier access to the data filtering step in the visualization preprocess. Their *interaction graph* and *n-dimensional window* give an overview over the dimensionality of the attribute space—that is, the data content. They allow for selecting subspaces of interest (reducing the dimensionality—that is, columns) or variable ranges of interest (reducing the number of data records—that is, rows), respectively. Other instances of descriptor visualizations include GeoVISTA's display of the maximum conditional entropy and correlation values between data content dimensions,[115] as well as the arrangement of data dimensions in a way that allows for exploring their interdependencies by Yang et al.[116]

On the other hand, the description can be used to suitably parametrize visualizations of the described dataset. It is noteworthy that to this end, data descriptors are at least implicitly already part of each visualization system, as without knowledge about the distribution of data values no meaningful color-coding and no sensible scaling of axes is possible. The concept of data descriptors gives these already existing means of describing a dataset a formal framework and advocates for dedicated mechanisms for gathering and managing them. So it is not a question of whether to use data descriptors or not, but how to use (and reuse) them.

Figure 4 shows the marine dataset that was already used in Figure 3, as it would be depicted in the absence of any further information about it. By default, the coordinate axes are sorted alphabetically by variable name, which is certainly not optimal, but it at least allows users to quickly seek out a variable of interest. Ideally, one would want to see related variables placed on axes close to each other, so as to ease their combined inspection. One can furthermore observe some unreasonably high values, such as water temperatures (TEMP) of around the boiling point of 100°C or salinity measurements (SALI) around 100%. While these values can be easily spotted and identified as being invalid, probably placeholders for missing values, they nevertheless distort the axes and reduce the axis resolution for valid values. For example, the majority of values on the temperature and salinity axes are now being compressed into the lower part and are hardly discernible.

After gathering data descriptors, as depicted in Figure 3, we can leverage this additional information about the data to alleviate these problems. To first establish a sense of plausibility for a given dataset, we can use a network diagram depicting a bivariate correlation network of the dataset's variables. This diagram, shown in Figure 5, exhibits a group of correlated data content

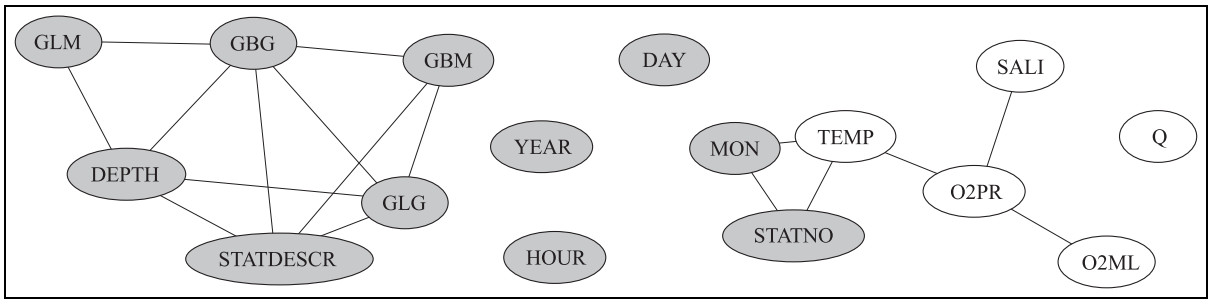

Figure 5. Network diagram of the unsigned bivariate correlation between different variables from the marine dataset described in Figure 3. Each node represents a variable, where gray nodes indicate data context variables and white nodes indicate data content variables. Links between two nodes denote a correlation of at least 0.15—a threshold that is necessary to set, as all variables are minimally correlated, which would result in a fully connected and thus meaningless display.

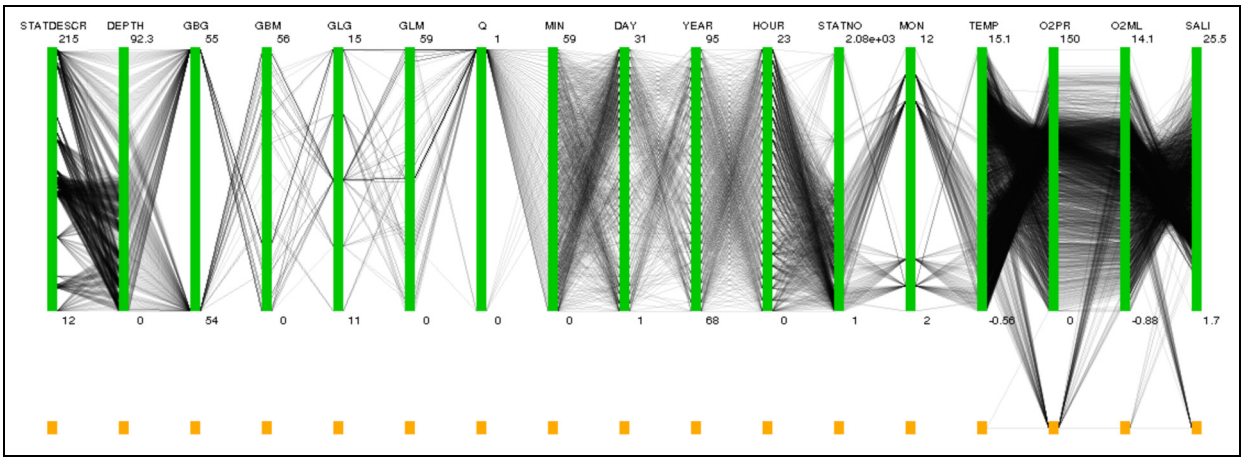

Figure 6. Improved parallel coordinates view from Figure 5. The axes have been re-ordered so that highly correlated variables are positioned close together. The orange marks below each axis single out the placeholder value for missing data.

variables on the right side of the figure. The correlations between salinity (SALI), relative oxygen levels (O2PR), absolute oxygen levels (O2ML), and water temperature (TEMP) are to be expected in a marine dataset. If they would not show up, the source of the dataset should be questioned. The credibility of the dataset is further underlined by the correlation between water temperature and month of the year (MON), which reflects a well-known seasonal pattern in the Baltic Sea. Merely the correlation between month, water temperature, and the numerical ID of the measurement station (STATNO) is an artifact that either happened by chance or is due to a particular numbering scheme for these stations that we are not aware of.

We can further use the bivariate correlations to improve the parallel coordinate plot from Figure 4 by auto-adjusting the order of the axes, so that correlated variables are placed in each other's proximity.[117] We can furthermore use identified placeholders for missing values to map them onto separate positions, so that the axes do not get distorted by them. Figure 6 shows the outcome of these descriptor-driven adaptations. The group of correlated data context variables from the left side of Figure 5 is clearly being placed together at the very left of the parallel coordinates, whereas the other observed group of correlated variables gets placed at the right side. The missing values are singled out below the axes and marked in orange. This does not only improve the visibility of the legitimate values as these are now spread across each axis but also eases the application of interactive means, such as brushing. Figure 7 illustrates the correlations among a subset of variables by brushing low and high temperatures, respectively.

Data descriptors for climate impact research

Many data-intensive scientific domains have established domain-specific data descriptors for their

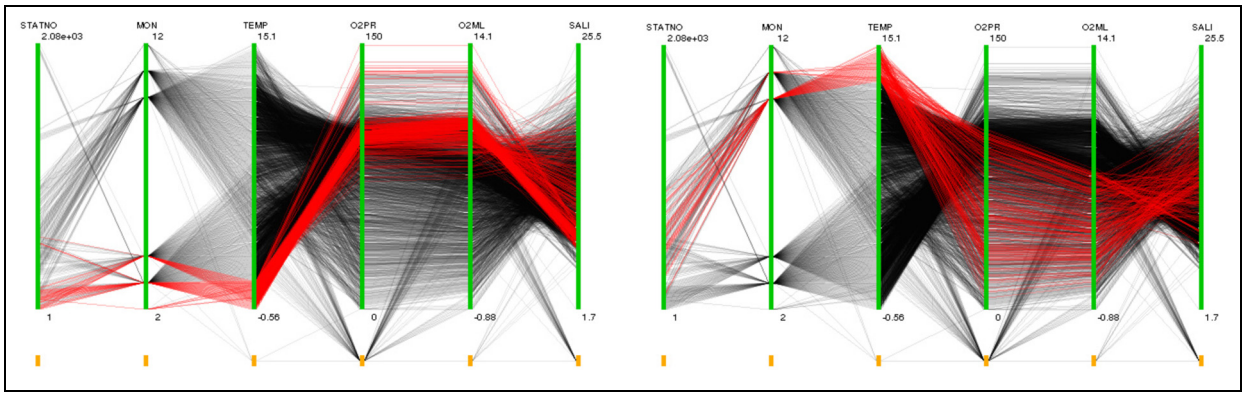

Figure 7. Looking at the variables from the six rightmost axes in Figure 6, we can investigate the correlations via brushing. The left figure highlights the low temperature records and we can observe that at these temperatures, we get mainly high oxygen levels and low salinity, whereas in the right figure, the highlighted high temperature records exhibit the reverse relations.

datasets—for example, ecology,[118] astronomy,[119] or systems biology.[120] Dedicated systems, so-called *metadata repositories*, that serve the purpose of storing, managing, and querying not the datasets themselves, but merely their DFDs and DSDs at dataset granularity have emerged from these fields.[121–123]

For this article, we have chosen the field of climate impact research to instantiate our set of tabular data descriptors. It presents a most challenging scenario for data descriptors, as it encompasses measurement and simulation data from a multitude of different disciplines, including meteorology, climatology, ecology, agriculture, hydrology, economy, and sociology. Hence, the domain-specific requirements for data descriptors in each of these fields are to some level reflected in those established for climate impact research. Furthermore, working with data from such a multitude of disciplines makes it necessary to meticulously keep track of the different datasets and their various revisions for their cross-disciplinary analysis. Hence, it is not surprising that the field of climate impact research has already developed a number of different notations and standards for data descriptors. While having such standards is a promising direction in theory, in practice climate impact researchers have to deal with a number of caveats that more often than not prevent the use of given data descriptors. These include the following:

- *Competing standards.* The different standards for data descriptors each describe some data aspects and leave out others. Yet they are not complementary to each other, as they overlap in some parts and are disjoint in others.
- *Different versions of a standard.* Standards evolve to capture notions that arise over time. This creates incompatibilities among different versions of the same standard.
- *Flexibility in interpreting a standard.* The standards leave some flexibility to their realization, which is why two software tools that officially support the same standard may not be able to use each other's descriptors.
- *Incomplete implementations of the standard.* Standards are rarely implemented in full and most software tools work with some sensible subsets that are useful in their context, but hardly match across tools.
- *Standardized descriptors still require validation.* Even if standards are fully implemented and a full set of data descriptors is available for a dataset, that does not mean that the provided descriptors are correct.

This shows that by having such standards, the data description and the use of data descriptors can hardly be automated in the background. While necessary to keep track of the data, it becomes another aspect of the data to which the user has to attend. To do so, the user must have basic knowledge about them in order to resolve conflicting descriptors or to identify implausible ones. This section gives a list of data flow and DSDs that are specific for the field of climate impact research, including their availability in the most important standards. To aid the climate researchers in maintaining consistent and valid descriptors for their data despite the challenges outlined above, this section furthermore introduces a software module for gathering data descriptors within a climate data visualization support tool.

DFDs for climate data

We have catalogued DFDs that are specifically suitable for climate-related data in Table 2 (top). In general, the rather generic ISO 19115-1 geodata standardization already includes DFDs, such as *provenance information* (e.g. evaluation method for quality assessment),

storage information (e.g. format, recommended decompression algorithm), and *utility information* (e.g. purpose). This standard is a good fit for geospatial data in general, but it does not explicitly support climate-specific descriptors. Explicitly, we mean that while the standard provides places to put climate-specific information in textual form for the user, there are no dedicated fields for this information that make it available to a visual analysis tool in the sense of our definition of data descriptors.

In particular, data provenance descriptors play an important role in climate impact research, as the data can originate from a wealth of sources, including measurements, simulations, or further postprocessing steps, as well as from a variety of disparate fields that all contribute to climate impact research. This is reflected by the NetCDF-CF convention, where "CF" stands for "Climate Forecast" and denotes an extension to the NetCDF standard that is developed by the University Corporation for Atmospheric Research (UCAR). It includes a number of more specific data provenance descriptors, as indicated in the top part of Table 2. For data originating from simulations, such climate-specific descriptors include information about the climate model, such as type and version, and about the simulation run on that model, such as the used driver, which are of major importance to assess and reproduce climate simulations. For measured data, this includes information about the measurement device (accuracy, precision, resolution, and sensitivity) and other acquisition information when conducting weather observations or collecting data from paleoclimatic ice cores or flowstones.

DSDs for climate data

DSDs that are relevant for the visual analysis of climate-related data are given in Table 2 (bottom). Here, it is the data context that is best covered by existing standards. This is not surprising, as climate researchers measure and simulate very different aspects and processes, but always in the same geophysical space. Hence, there is a consensus about what climate researchers want to describe about the data context and this consensus has been formalized through standards. Whereas for the data content, possible descriptors are much more diverse and thus their set is much less standardized.

A few generic data context descriptors are part of the general NetCDF convention, such as the information about masked data—that is, data that is only available for certain regions, such as land or sea area. Others that are more specific are captured by the NetCDF-CF extension. For example, for representing the dynamics of oceans, atmosphere, and ice shields, physical variables are provided as sub-models in different spatial dimensionalities (1D, 2D, 3D), with partly different, linked grid structures and varying temporal granularities (see, for example, Petoukhov et al.[124]) that define the *data context*. For a meaningful visualization of climate data, these structures and dependencies need to be known. In addition, typical *data content* descriptors, such as climate-related regions of interest in the data, range from centers of pressure systems,[125] to weather fronts and storm tracks,[126] and even to the 3D tracking of clouds, dust, and atmospheric pollutants.[127] Descriptors for paleo-climate analysis include periodicity and time-delayed correlations. While uncertainty information can be explicitly stored using the NetCDF-U extension, in practice this is rarely done and it is more common to include an additional variable representing the uncertainty information.

Gathering data descriptors for climate data

In the previous section on tabular data, we concerned ourselves mostly with the computation and interactive specification of data descriptors as these are usually not provided alongside the data. For data from the domain of climate impact research, the situation is quite different, as the sections on climate-specific data descriptors and their standards have illustrated: a range of descriptors are usually already given—yet, they are possibly incomplete, incompatible, or inconsistent. Thus, a gathering of data descriptors in climate impact research focuses on retrieving or querying those existing data descriptors, computing those that are missing or do not match the data, and converting them into the right standard and version for the visual analysis tool to be subsequently used.

Precisely for this purpose, we have developed a data descriptor module within a climate data visualization support tool.[128] This module helps climate scientists to bridge the gap between the data descriptors that are given and those that should be given for a subsequent visual analysis. In a first step, it *queries* the descriptors stored with the NetCDF/NetCDF-CF data to obtain any descriptors already provided by the data. If they are incomplete, it *derives* additional data descriptors from the dataset in a second step. Finally, in a third step, it presents the queried and derived descriptors to the user in order to prompt for *input* for the interactive adaptation of those that have been gathered and for the completion of those that cannot be computed automatically.

The first step queries data descriptors from the NetCDF data description. NetCDF is a particularly good source for dataset descriptors, as many descriptors are mandatory by the NetCDF convention and must be given. This includes information about the

Table 2. Data descriptors for climate data as they are supported by the various standards.

Data provenance	Data storage	Data utility
NetCDF-CF: Institution, date Name of the climate (impact) model Simulation experiment type (e.g. Monte-Carlo) Data generation workflow/ operator sequence Not standardized: Author	NetCDF-CF: Climate/model-specific conventions Not standardized: Storage format (e.g. GRIB, ASCII, binary) Data partitioning scheme (e.g. all data in one file, each time step in a separate file)	Not standardized: Kind of analyses the dataset can be used for (e.g. hydrology simulations, storm track analysis)
Data space descriptors	**Data context**	**Data content**
Granularity		
Value	NetCDF-U: Dating uncertainty (e.g. age dating for ice cores or flowstones)	NetCDF-CF: Domain-specific missing values NetCDF-U: Domain-specific value uncertainties (e.g. of the emission scenario, global/regional climate model, impact model)
Record	NetCDF: Grid values restricted to certain regions (e.g. ocean only, land surface only, with or without Greenland/Arctic) NetCDF-CF: Measurement station Measurement position change (e.g. balloon or ship) NetCDF-U: Uncertainty	NetCDF-CF: Meteorological/climatic features: 1D: centers of pressure systems Not standardized: Meteorological/climatic features: 2D: storm tracks, weather fronts, jet stream 3D: clouds, dust, circulation patterns
Variable	NetCDF: Dimensions describing simulation ensemble factors NetCDF-CF: Domain-specific type of spatial dimensions (longitude, latitude, pressure level) Values defined for centers, edges, or vertices of grid cells? Variable-specific properties of time (e.g. different time steps) Kind of coordinate reference system (e.g. geographical, projected) and associated properties (e.g. rotated pole, conformal, equidistant)	NetCDF-CF: Holds certain climate quantity (e.g. temperature 2 m above sea level, precipitation including snow and/or hail) Not standardized: Fit of simulated variable distributions to reference data (e.g. measurements or reanalysis data) -> bias measures Trends of mean values, variability, and extremes Periodicities and time-delayed correlations
Dataset	NetCDF: Number and kind of homogeneous subsets (e.g. earth surface and 3D atmospheric variable sets) NetCDF-CF: Geospatial extent (global, regional, urban)	NetCDF-CF: Relation between variables (e.g. thickness and temperature of sea ice)

grid's dimensionality and structure, which allows us to automatically distinguish between data context variables (i.e. longitude, latitude, and time) and data content variables (i.e. the measured or simulated values) as well as to relate different grids/ensemble members.

The second step gathers optional data descriptors that were not given with the data. For example, often missing within NetCDF are the data extent (point, local, global) or a variable's unit. Where possible, our tool tries to infer these from domain knowledge and assumes, for example, that "lightning" is point information and that a variable "temperature" has the unit "Kelvin." It also tries to find and fix common spelling errors and replaces, for example, the unknown identifier "units" with the standard-conform identifier "unit." Only if no descriptor is found—neither directly nor by the described inference—our tool aims to gather it using the pipeline from Figure 2.

The third step presents the results of the previous two steps to the user, as it is illustrated in Figure 8. The shown interface provides an overview of the data content variables, each being assigned a colored glyph that denotes the context, which provides the frame of reference for that variable (Figure 8(a)). The number of nodes that comprise these glyphs indicate the dimensionality of the corresponding data context, with differently colored glyphs marking different contexts. In the shown example, one can see two 4D data contexts colored yellow for ice days and summer days and orange for wind speed. To find out concretely which variables comprise these contexts, one can switch to the data context tab (Figure 8(b)). It becomes evident that the two 4D contexts are both composed of latitude, longitude, and time and that they only differ in the included altitude variable. One can further notice that these altitude variables are somewhat peculiar, as the "Nr of Values" descriptor shows that they only contain a single data value each: 2 and 10 m, respectively, as a quick glance at the "Value Range" descriptor reveals. So, the altitude is in both cases not a variable (as there is no variability), but a constant that simply means that wind speeds where simulated at an altitude of 10 m, while ice days and summer days where defined at 2 m. Trying to visualize the data in this "pseudo" 4D form would most certainly lead to an ill-configured 3D visualization, as the data are actually flat and should also be visualized as such. The interface further allows the users to inspect context descriptors for selected subsets (Figure 8(c)) and DFDs (Figure 8(d)) and to interactively readjust them if necessary.

The software module has been designed in collaboration with researchers from the Potsdam Institute of Climate Impact Research and is in active use. User interviews have highlighted two principal usage scenarios: before and after the visualization. Using the module before the visualization reflects the idea of an initial analysis step that gives first insight into the data to decide on later analysis steps. Whereas the usage afterward reflects a debugging process where users found their data misrepresented and are looking for the reason. The above example of the "pseudo" 4D dataset illustrates how easily a visualization system can misinterpret a dataset and thus how important it is to be able to go back from an improper visualization to inspect and adjust the data descriptors.

Leveraging data descriptors for visualizing climate-related data

One of the main problems in climate impact research is neither a lack of data descriptors nor a lack of standards to convey them, but that they are only selectively and inconsistently used by different visualization tools, as can be seen in Figure 9. The figure shows a dataset that contains the initial state for a COSMO/CLM (CCLM) climate simulation run.[129] The depicted variable quantifies the height of the snow cover over Europe, as indicated by the following NetCDF description:

```
float W_SNOW(time, rlat, rlon);
  W_SNOW:standard_name =
    "lwe_thickness_of_surface_snow_amount";
  W_SNOW:long_name =
    "surface snow amount";
  W_SNOW:units = "m";
  W_SNOW:grid_mapping = "rotated_pole";
  W_SNOW:coordinates = "lon lat";
  W_SNOW:_FillValue = -1.e + 20f;
```

This description contains valuable information for generating a proper visualization for the dataset: that the values are given in meters is important for providing a legend, that missing entries are indicated by a value of -1.0×10^{20} is helpful for masking these entries in the resulting view, and that the given coordinates are rotated is relevant for a correct spatial mapping to the map or globe. Such a rotation of the coordinate grid is often used to yield evenly sized grid cells for regions that are not close to the equator, to ensure stable numerical simulation. If some of these descriptors were missing or incorrect—for example, units or missing values—the researcher would have been able to add them using the gathering module. So, we assume that our data descriptors are correct, complete, and conform to the NetCDF standard, so that a visualization tool should theoretically be able to render a correct view of the snow cover dataset. The

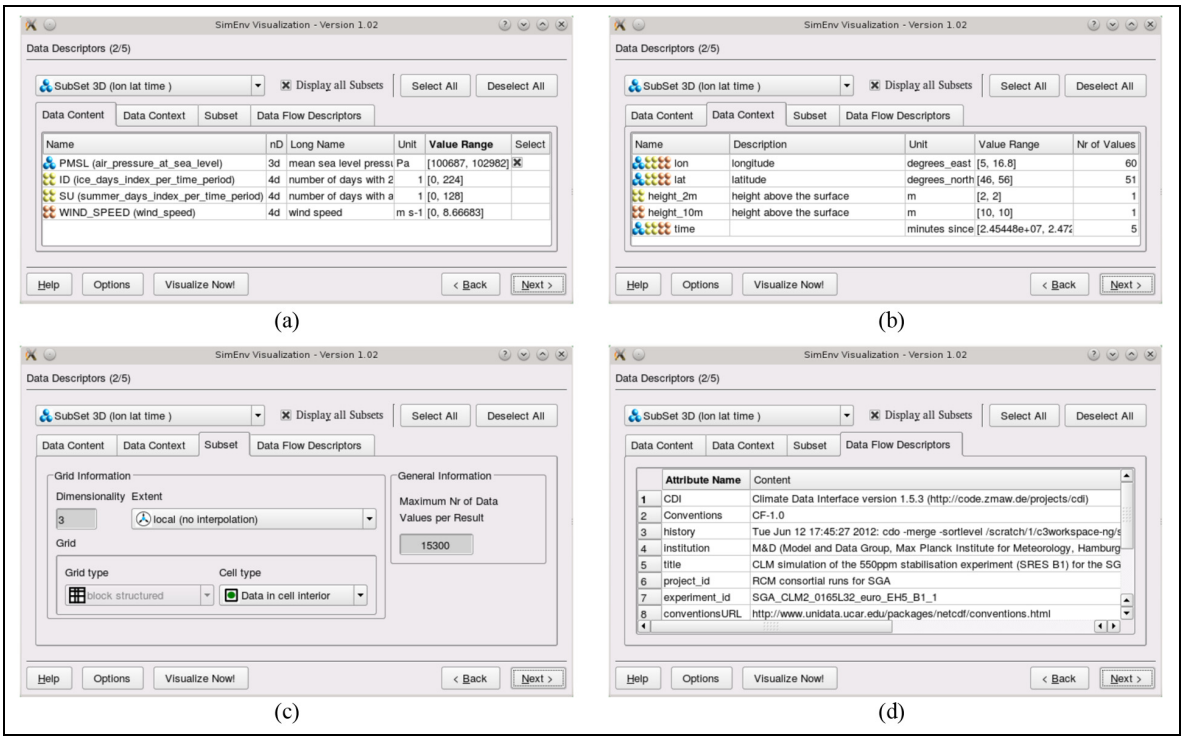

Figure 8. The gathering module allows a detailed examination of the descriptors for (a) the data content, (b) the data context, (c) gridded subsets of the context, and (d) the data flow. The shown example is from a dataset generated by a regional climate model simulation with the CLM model.

grid adjustment for the spatial mapping can be determined from the specifics of the rotation, which are also given in NetCDF:

```
char rotated_pole;
  rotated_pole:grid_mapping_name =
    "rotated_latitude_longitude";
  rotated_pole:grid_north_pole_latitude =
    39.25f;
  rotated_pole:grid_north_pole_longitude =
    -162.f;
  float rlon(rlon);
    rlon:axis = "X";
    rlon:standard_name
 = "grid_longitude";
    rlon:long_name = "rotated longitude";
    rlon:units = "degrees";
  float rlat(rlat);
    rlat:axis = "Y";
    rlat:standard_name = "grid_latitude";
    rlat:long_name = "rotated latitude";
    rlat:units = "degrees";
```

Yet, Figure 9 illustrates that given the same dataset and the same data description in NetCDF, different visualization tools generate very different outcomes. We have tested the ability of four different visualization tools that are commonly used in climate research to visualize the NetCDF-described snow cover dataset using their default settings. *Avizo* (see http://www.fei.com/software/avizo3d) properly recognizes the rotated grid from the NetCDF descriptors and automatically adjusts for the rotation, so that the data are correctly mapped onto Europe. Yet, in its standard configuration, Avizo does not correctly mask the missing values, which basically coincide with sea surfaces, and it also does not display the unit of measurement in the legend. Whereas *OpenDX*[130] properly masks the missing values and displays the units correctly, but does not correctly translate the spatial mapping to adjust for the rotated grid. This leads to the data clearly showing the outline of Europe being overlaid on the map of Africa. Similarly, *Ferret*[131] also handles the missing values correctly and gives an indication of the unit of measurement, yet neglects the rotation of the coordinates, as can be seen from the latitude/longitude labels on the axes. Finally *NCL* (see http://www.ncl.ucar.edu), the NCAR Command Language, can actually leverage all three of the highlighted descriptors—units, missing values, and rotation—using one of its standard example scripts.

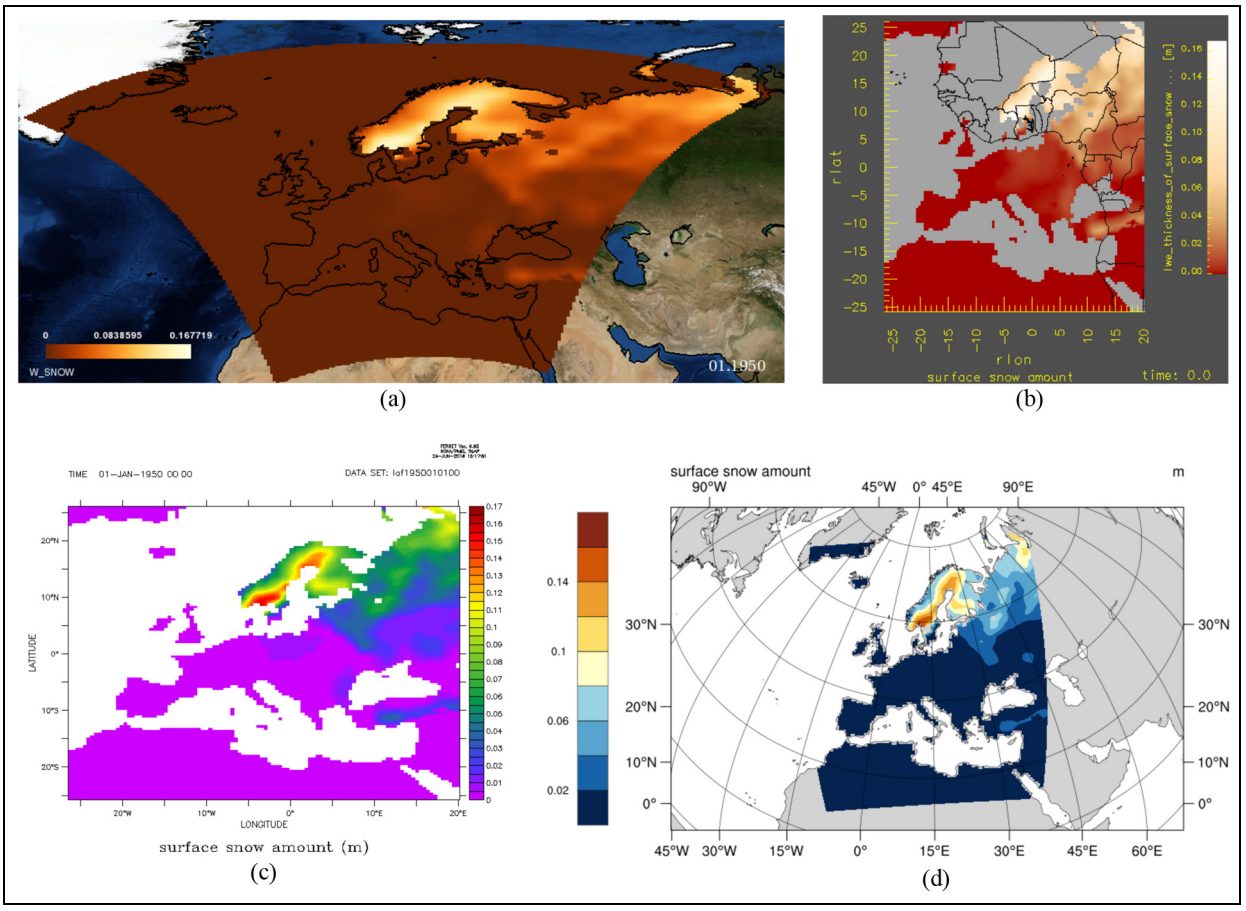

Figure 9. Visualizations of the Snow Cover Dataset with four different standard tools commonly used in climate science. It can be observed that (a) Avizo, (b) OpenDX, and (c) Ferret make only selective use of the given NetCDF descriptors, which results in faulty visualizations. Only (d) NCL was able to actually leverage all of the given descriptors for generating a proper visualization.

To be fair, we have to note that these visualizations were generated with the respective tools using standard parametrizations and default options without any further user intervention. Some of their deficits could be alleviated by further manual fine-tuning. For example, Avizo allows users to adjust the color mapping in the *Colormap Editor*, which can be used to mask the missing values by hand and thus to eventually produce a proper visualization of that dataset. Yet even for that, the users themselves must go over the NetCDF code in a text editor to find out which value is used to indicate missing data in order to adjust the colors accordingly. Domain knowledge about which visualization tool supports which kinds of NetCDF descriptors could potentially help to identify tools that are suitable for a dataset at hand in future work.

Concluding remarks

With the proposed concept of data descriptors, we do not present yet another taxonomy of data types and structures, but instead a unifying view on those that already exist. From its scope, our concept is more in line with approaches, such as *meta-metadata*,[132] that aim to bring together different data descriptor standards under a unified meta standard. Yet, we differ in the path taken toward such a unifying view: while the existing approaches take a top-down perspective and define new standards for researchers to adhere to, our concept takes a bottom-up perspective by filling in the gaps and resolving inconsistencies.

Implications

From our own experience in working with climate researchers, this bottom-up approach is the more practical one as it yields concrete results that we can already start using while waiting for the definite data standard to arrive. Yet this makes it also the more involved approach as compared to defining a suitable data description standard and further assuming that any given analysis input adheres to it. It takes

computational effort and the involvement of the users with their background knowledge to provide information about data that are

- Accurate (matches underlying data);
- Complete (covers all relevant data aspects);
- Consistent (does not contradict itself);
- Current (reflects data changes);
- Conforms to whichever conventions are used by the variety of existing analysis tools.

Our concept of data descriptors lifts this information from a few scattered auxiliary measures to being data entities in themselves. This allows us to centralize the required effort for providing reliable information about data in a dedicated software module—our gathering pipeline. The pipeline serves as the principal access point to this information for users to adjust them and for analysis methods to utilize them.

Limitations

Providing data descriptors is only one side of the equation. On the other side, we have no influence over whether and to which degree subsequent visual analysis steps actually use given data descriptors. This was illustrated by the examples in the previous section. To some degree, this lies in the nature of such a bottom-up approach that rather offers descriptors for use than to enforce their observance. After all, it is hard to determine in general, which descriptors *must* be given and used, and which *can* be given and used—that is, which are ultimately necessary and which would be helpful, but one could do without. This depends largely on the concrete visual analysis task at hand. While we do not know the analysis task in beforehand, data type (e.g. tabular) and application domain (e.g. climatology) can help to limit the set of all possible data descriptors to those that are suitable for the data type and typical in the domain. In the same way, as we have tailored our general descriptor framework to tabular data in a first step, and then to the concrete requirements of the climate research domain, adaptations are necessary for other data types and domains. This highlights once again that the descriptors listed in this article and the pipeline for gathering them are not a one-stop solution for all possible data/domain combinations, but rather a blueprint to be adapted to the specifics of other data types and to be refined for other domains. Providing a set of descriptors that is well adapted to the data and tasks of a particular domain is certainly more likely to be picked up on by the tools and users in that domain than some unwieldy all-encompassing generic metadata solution.

Generalization

It is noteworthy that common visual analysis strategies do not explicitly include such an initial analysis step of gathering information about the data. Following the commonly applied strategies, an analyst can initiate a visual analysis either with an *overview-first step*[56] or with an *analyze-first step*.[133] Yet in both cases, it remains challenging to decide which overview or analysis method shall be invoked, respectively, on which data subset and with which parameter settings. Hence, our initial gathering can be understood as a *describe-first step* that precedes overview or analysis and collects information about the data. This information can either be visualized directly so that users get a meta-view of their data, or it can be used indirectly for an informed descriptor-driven selection and parametrization of appropriate computational or visual methods.

Continuation

For the future, we anticipate that descriptive information about data will play an increasingly important role in visual data analysis for two reasons. On one hand, the push for Big Data increases the amount of data, which in turn has to be described in a meaningful way to select the right subset for an analysis at hand. On the other hand, more often than not these days, the data provider is different from the data analyst and thus information about datasets needs to be passed on.[134] As this trend grows with current movements, such as Open Data and Open Science, the visualization and visual analytics community needs to develop concepts and tools to deal with it. We strongly believe that the concept of data descriptors will serve as an important foundation for these developments.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work originated from a long-standing collaboration between the University of Rostock and the Potsdam Institute of Climate Impact Research. The authors acknowledge financial support by the Federal Ministry for Education and Research via the Potsdam Research Cluster for Georisk Analysis, Environmental Change and Sustainability (PROGRESS).

References

1. Adèr HJ. Phases and initial steps in data analysis. In: Adèr HJ and Mellenbergh GJ (eds) *Advising on research methods: a consultant's companion*. Huizen: Johannes van Kessel Publishing, 2008, pp. 333–355.

2. Gschwandtner T, Aigner W, Miksch S, et al. TimeCleanser: a visual analytics approach for data cleansing of time-oriented data. In: *Proceedings of the conference on knowledge technologies and data-driven business (i-Know'14)* (ed S Lindstaedt, M Granitzer and H Sack), Graz, 16–19 September 2014, pp. 18-1–18-8. New York: ACM.
3. Silva S, Santos BS and Madeira J. Using color in visualization: a survey. *Comput Graph* 2011; 35(2): 320–333.
4. Bergman LD, Rogowitz BE and Treinish LA. A rule-based tool for assisting colormap selection. In: *Proceedings of the IEEE conference on visualization (VIS'95)*, Atlanta, GA, 29 October–3 November 1995, pp. 118–125. New York: IEEE.
5. Lin S, Fortuna J, Kulkarni C, et al. Selecting semantically-resonant colors for data visualization. *Comput Graph Forum* 2013; 32(3pt4): 401–410.
6. Setlur V and Stone MC. A linguistic approach to categorical color assignment for data visualization. *IEEE T Vis Comput Gr* 2016; 22(1): 698–707.
7. Gitelman L (ed.). *"Raw data" is an oxymoron*. Cambridge, MA: MIT Press, 2013.
8. Duval E. Metadata standards: what, who & why. *J Univers Comput Sci* 2001; 7(7): 591–601.
9. Rogowitz BE and Matasci N. Metadata Mapper: a web service for mapping data between independent visual analysis components, guided by perceptual rules. In: *Proceedings of the conference on visualization and data analysis (VDA'11)* (ed PC Wong, J Park, MC Hao, et al.), San Francisco, CA, 23 January 2011, paper no. 78650I, pp. 1–13. Bellingham, WA: SPIE.
10. Flöring S. *KnoVA: a reference architecture for knowledge-based visual analytics*. PhD Thesis, University of Oldenburg, Oldenburg, 2012.
11. Arens Y, Hovy EH and Vossers M. On the knowledge underlying multimedia presentations. In: Maybury MT (ed.) *Intelligent multimedia interfaces*. Menlo Park, CA: AAAI Press, 1993, pp. 280–306.
12. Steinacker A, Ghavam A and Steinmetz R. Metadata standards for web-based resources. *IEEE Multimedia* 2001; 8(1): 70–76.
13. International Organization for Standardization (ISO) 19115-1:2014. Geographic information (metadata).
14. Riede M, Schueppel R, Sylvester-Hvid KO, et al. On the communication of scientific data: the full-metadata format. *Comput Phys Commun* 2010; 181(3): 651–662.
15. Carata L, Akoush S, Balakrishnan N, et al. A primer on provenance. *Commun ACM* 2014; 57(5): 52–60.
16. Davison A. Automated capture of experiment context for easier reproducibility in computational research. *Comput Sci Eng* 2012; 14(4): 48–56.
17. Simmhan YL, Plale B and Gannon D. A survey of data provenance in e-science. *SIGMOD Rec* 2005; 34(3): 31–36.
18. Simmhan YL, Plale B and Gannon D. *A survey of data provenance techniques*. Technical report IUB-CS-TR618, August 2005. Bloomington, IN: Computer Science Department, Indiana University.
19. Glavic B and Dittrich KR. Data provenance: a categorization of existing approaches. In: *Proceedings of the GI-Fachtagung Datenbanksysteme in Business, Technologie und Web (BTW'07)* (ed A Kemper, H Schöning, T Rose, et al.; Number 103 in lecture notes in informatics), Aachen, Germany, 5–9 March 2007, pp. 227–241. Bonn, Germany: Bonner Köllen Verlag.
20. Da Cruz SMS, Campos MLM and Mattoso M. Towards a taxonomy of provenance in scientific workflow management systems. In: *Proceedings of the world conference on services* (ed LJ Zhang), Los Angeles, CA, 6–10 July 2009, pp. 259–266. New York: IEEE.
21. Buneman P, Khanna S, Wang-Chiew T, et al. Why and where: a characterization of data provenance. In: *Proceedings of the international conference on database theory (ICDT'01)* (ed J Van den Bussche and V Vianu; Number 1973 in lecture notes in computer science), London, 4–6 January 2001, pp. 316–330. Berlin: Springer.
22. Cohen S, Cohen-Boulakia S and Davidson S. Towards a model of provenance and user views in scientific workflows. In: *Proceedings of the workshop on data integration in the life sciences (DILS'06)* (ed U Leser, F Naumann and B Eckman; Number 4075 in lecture notes in computer science), Hinxton, 20–22 July 2006, pp. 264–279. Berlin: Springer.
23. Ludäscher B, Podhorszki N, Altintas I, et al. From computation models to models of provenance: the RWS approach. *Concurr Comp: Pract E* 2008; 20(5): 507–518.
24. Moreau L, Clifford B, Freire J, et al. The Open Provenance Model core specification (v1.1). *Future Gener Comp Sy* 2011; 27(6): 743–756.
25. Keim DA, Kohlhammer J, Ellis G, et al. (eds). *Mastering the information age*. Goslar: Eurographics Association, 2010.
26. Freire J, Koop D, Santos E, et al. Provenance for computational tasks: a survey. *Comput Sci Eng* 2008; 10(3): 11–21.
27. Silva CT, Freire J and Callahan SP. Provenance for visualizations: reproducibility and beyond. *Comput Sci Eng* 2007; 9(5): 82–89.
28. Groth DP and Streefkerk K. Provenance and annotation for visual exploration systems. *IEEE T Vis Comput Gr* 2006; 12(6): 1500–1510.
29. Heer J, Mackinlay J, Stolte C, et al. Graphical histories for visualization: supporting analysis, communication, and evaluation. *IEEE T Vis Comput Gr* 2008; 14(6): 1189–1196.
30. Ragan ED, Endert A, Sanyal J, et al. Characterizing provenance in visualization and data analysis: an organizational framework of provenance types and purposes. *IEEE T Vis Comput Gr* 2016; 22(1): 31–40.
31. Stitz H, Luger S, Streit M, et al. AVOCADO: visualization of workflow-derived data provenance for reproducible biomedical research. *Comput Graph Forum* 2016; 35(3): 481–490.
32. Vassiliadis P. Data warehouse metadata. In: Liu L and Özsu MT (eds) *Encyclopedia of database systems*. New York: Springer, 2009, pp. 669–675.

33. Hoxmeier JA. Dimensions of database quality. In: Khosrow-Pour M (ed.) *Encyclopedia of information science and technology*. Hershey, PA: Idea Group, 2005, pp. 886–891.
34. Codd EF. A relational model of data for large shared data banks. *Commun ACM* 1970; 13(6): 377–387.
35. Codd EF. *The relational model for database management* (version 2). Boston, MA: Addison-Wesley, 1990.
36. Angles R and Gutierrez C. Survey of graph database models. *ACM Comput Surv* 2008; 40(1): 1–39.
37. Zhuge H. Resource space model, its design method and applications. *J Syst Software* 2004; 72(1): 71–81.
38. Zhuge H, Yao E, Xing Y, et al. Extended resource space model. *Future Gener Comp Sy* 2005; 21(1): 189–198.
39. Franklin M, Halevy A and Maier D. From databases to dataspaces: a new abstraction for information management. *SIGMOD Rec* 2005; 34(4): 27–33.
40. International Organization for Standardization (ISO)/IEC 10027:1990. Information resource dictionary system (IRDS) framework.
41. World Wide Web Consortium (W3C). Resource description framework (RDF) 1.1 *concepts and abstract syntax*, 2014, http://www.w3.org/TR/rdf-concepts/
42. North C and Shneiderman B. Snap-together visualization: a user interface for coordinating visualizations via relational schemata. In: *Proceedings of the working conference on advanced visual interfaces (AVI'00)* (ed VD Geaú, S Levialdi and L Tarantino), Palermo, 24–26 May 2000, pp. 128–135. New York: ACM.
43. Lieberman MD, Taheri S, Guo H, et al. Visual exploration across biomedical databases. *IEEE/ACM T Comput Bi* 2011; 8(2): 536–550.
44. Cammarano M, Dong X, Chan B, et al. Visualization of heterogeneous data. *IEEE T Vis Comput Gr* 2007; 13(6): 1200–1207.
45. Tshagharyan G and Schulz HJ. A graph-based overview visualization for data landscapes. *Comput Sci Inform Technol* 2013; 1(3): 225–232.
46. Stephen R. *Resource availability awareness and data utility: The foundation for a DSS framework in a pervasive computing environment*. PhD Thesis, University of Maryland at Baltimore County, Catonsville, MD, 2008.
47. Karr AF, Kohnen CN, Oganian A, et al. A framework for evaluating the utility of data altered to protect confidentiality. *Am Stat* 2006; 60(3): 224–232.
48. Grammer G, Joshi S, Kroeschel W, et al. Obfuscating sensitive data while preserving data usability. Patent application US 20120272329, USA, 2012.
49. Bassiouni MA. Data compression in scientific and statistical databases. *IEEE T Software Eng* 1985; 11(10): 1047–1058.
50. Dasgupta A, Chen M and Kosara R. Measuring privacy and utility in privacy-preserving visualization. *Comput Graph Forum* 2013; 32(8): 35–47.
51. Streit M, Schulz HJ, Lex A, et al. Model-driven design for the visual analysis of heterogeneous data. *IEEE T Vis Comput Gr* 2012; 18(6): 998–1010.
52. Zhou MX and Feiner SK. Data characterization for automatically visualizing heterogeneous information. In: *Proceedings of the IEEE symposium on information visualization (InfoVis'96)* (ed ND Gershon, S Card and SG Eick), San Francisco, CA, 28–29 October 1996, pp. 13–20. New York: IEEE.
53. Stevens SS. On the theory of scales of measurement. *Science* 1946; 103(2684): 677–680.
54. Roth SF and Mattis J. Data characterization for intelligent graphics presentation. In: *Proceedings of the SIGCHI conference on human factors in computing systems (CHI'90)* (ed JC Chew and J Whiteside), Seattle, WA, 1–5 April 1990, pp. 193–200. New York: ACM.
55. Andrienko N and Andrienko G. *Exploratory analysis of spatial and temporal data—a systematic approach*. Berlin: Springer, 2006.
56. Shneiderman B. The eyes have it: a task by data type taxonomy for information visualizations. In: *Proceedings of the IEEE symposium on visual languages (VL'96)*, Boulder, CO, 3–6 September 1996, pp. 336–343. New York: IEEE.
57. Robertson PK. A methodology for choosing data representations. *IEEE Comput Graph* 1991; 11(3): 56–67.
58. Lux M. Level of data—a concept for knowledge discovery in information spaces. In: *Proceedings of the conference on information visualization (IV'98)* (ed E Banissi, F Khosrowshahi and M Sarfraz), London, 31 July 1998, pp. 131–136. New York: IEEE.
59. Tory M and Möller T. Human factors in visualization research. *IEEE T Vis Comput Gr* 2004; 10(1): 72–84.
60. Hahmann S and Burghardt D. How much information is geospatially referenced? Networks and cognition. *Int J Geogr Inf Sci* 2013; 27(6): 1171–1189.
61. Kim W, Choi BJ, Hong EK, et al. A taxonomy of dirty data. *Data Min Knowl Disc* 2003; 7(1): 81–99.
62. Oliveira P, Rodrigues F and Henriques P. A formal definition of data quality problems. In: *Proceedings of the international conference on information quality (ICIQ'05)*, Massachusetts Institute of Technology (MIT), Cambridge, MA, 4–6 November 2005.
63. Gschwandtner T, Gärtner J, Aigner W, et al. A taxonomy of dirty time-oriented data. In: *Proceedings of the cross domain conference and workshop on availability, reliability, and security (CD-ARES'12)* (ed G Quirchmayr, J Basl, I You, et al.; Number 7465 in lecture notes in computer science), Prague, 20–24 August 2012, pp. 58–72. Berlin: Springer.
64. Batini C and Scannapieca M. *Data quality: concepts, methodologies and techniques*. New York: Springer, 2006.
65. Josko JMB, Oikawa MK and Ferreira JE. A formal taxonomy to improve data defect description. In: *Proceedings of the workshops on database systems for advanced applications (DASFAA'16)* (ed H Gao, J Kim and Y Sakurai), Dallas, TX, 16–19 April 2016, pp. 307–320. Cham: Springer.
66. Ayyub BM and Klir GJ. *Uncertainty modeling and analysis in engineering and the sciences*. Boca Raton, FL: CRC Press, 2006.

67. Drosg M. *Dealing with uncertainties: a guide to error analysis*. 2nd ed. Berlin: Springer, 2009.
68. Cleary J, Holmes G, Cunningham SJ, et al. Metadata for database mining. In: *Proceedings of the IEEE conference on metadata*, Silver Spring, MD, 16–18 April 1996.
69. International Organization for Standardization (ISO)/IEC 25012:2008. Software product quality requirements and evaluation (SQuaRE) data quality model.
70. Rodríguez A, Caro A, Cappiello C, et al. A BPMN extension for including data quality requirements in business process modeling. In: *Proceedings of the international workshop on business process model and notation (BPMN'12)* (ed J Mendling and M Weidlich; Lecture notes in business information processing), Vienna, 12–13 September 2012, volume 125, pp. 116–125. Berlin: Springer.
71. Sulo R, Eick S and Grossman RL. DaVis: a tool for visualizing data quality. In: *Poster compendium of the IEEE symposium on information visualization (InfoVis'05)*, Minneapolis, MN, 23–25 October 2005, pp. 45–46. New York: IEEE.
72. Josko JMB and Ferreira JE. Visualization properties for data quality visual assessment: an exploratory case study. *Inform Visual*. Epub ahead of print 14 March 2016. DOI: 10.1177/1473871616629516.
73. Pang AT, Wittenbrink CM and Lodha SK. Approaches to uncertainty visualization. *Visual Comput* 1997; 13(8): 370–390.
74. Thomson J, Hetzler E, MacEachren A, et al. A typology for visualizing uncertainty. In: *Proceedings of the conference on visualization and data analysis (VDA'05)* (ed RF Erbacher, JC Roberts, MT Gröhn, et al.), San Jose CA, 16–20 January 2005. Bellingham, WA: SPIE.
75. Correa CD, Chan YH and Ma KL. A framework for uncertainty-aware visual analytics. In: *Proceedings of the IEEE symposium on visual analytics science and technology (VAST'09)* (ed J Stasko and JJ Van Wijk), Atlantic City, NJ, 12–13 October 2009, pp. 51–58. New York: IEEE.
76. Skeels M, Lee B, Smith G, et al. Revealing uncertainty for information visualization. *Inform Visual* 2010; 9(1): 70–81.
77. Ward M, Xie Z, Yang D, et al. Quality-aware visual data analysis. *Computation Stat* 2011; 26(4): 567–584.
78. Potter K, Rosen P and Johnson CR. From quantification to visualization: a taxonomy of uncertainty visualization approaches. In: *Proceedings of the IFIP working conference on uncertainty quantification in scientific computing: IFIP advances in information and communication technology* (ed AM Dienstfrey and RF Boisvert), Boulder, CO, 1–4 August 2011, volume 377, pp. 226–249. Berlin: Springer.
79. Brodlie K, Osorio RA and Lopes A. A review of uncertainty in data visualization. In: Dill J, Earnshaw R, Kasik D, et al. (eds) *Expanding the frontiers of visual analytics and visualization*. London: Springer, 2012, pp. 81–109.
80. Ristovski G, Preusser T, Hahn HK, et al. Uncertainty in medical visualization: towards a taxonomy. *Comput Graph* 2014; 39: 60–73.
81. Theus M, Hofmann H, Siegl B, et al. MANET—extensions to interactive statistical graphics for missing values. In: *New techniques and technologies for statistics II: proceedings of the second Bonn seminar*. Amsterdam: IOS Press, pp. 247–259.
82. Swayne DF and Buja A. Missing data in interactive high-dimensional data visualization. *Computation Stat* 1998; 13(1): 15–26.
83. Fernstad SJ and Glen RC. Visual analysis of missing data—to see what isn't there. In: *Proceedings of the IEEE conference on visual analytics science and technology (VAST'14)* (ed M Chen, D Ebert and C North), Paris, 25–31 October 2014, pp. 249–250. New York: IEEE.
84. Cheng X, Cook D and Hofmann H. Visually exploring missing values in multivariable data using a graphical user interface. *J Stat Softw* 2015; 68(6): 1–23.
85. Bilgic M, Licamele L, Getoor L, et al. D-Dupe: an interactive tool for entity resolution in social networks. In: *proceedings of the IEEE symposium on visual analytics science and technology (VAST'06)* (ed PC Wong and D Keim), Baltimore, MD, 31 October–2 November 2006, pp. 43–50. New York: IEEE.
86. Kang H, Getoor L, Shneiderman B, et al. Interactive entity resolution in relational data: a visual analytic tool and its evaluation. *IEEE T Vis Comput Gr* 2008; 14(5): 999–1014.
87. Kang H, Sehgal V and Getoor L. GeoDDupe: a novel interface for interactive entity resolution in geospatial data. In: *Proceedings of the international conference information visualization (IV'07)* (ed E Banissi, RA Burkhard, G Grinstein, et al.), Zurich, 4–6 July 2007, pp. 489–496. New York: IEEE.
88. Galhardas H, Simon E and Tomasic A. A framework for classifying scientific metadata. In: *Proceedings of the AAAI workshop on artificial intelligence and information integration*, Madison, WI, 26–27 July 1998, pp. 106–113. California, USA: AAAI Press.
89. Brodlie KW. Visualization techniques. In: Brodlie KW, Carpenter LA, Earnshaw RA, et al. (eds) *Scientific visualization: techniques and applications*. Berlin: Springer, 1992, pp. 37–85.
90. Brodlie KW and Noor NM. Visualization notations, models and taxonomies. In: *Proceedings of the theory and practice of computer graphics conference (TPCG'07)* (ed IS Lim and D Duce), Bath, Wales, 3–15 June 2007, pp. 207–212. Goslar: Eurographics Association.
91. Butler DM and Pendley MH. A visualization model based on the mathematics of fiber bundles. *Comput Phys* 1989; 3(5): 45.
92. Bergeron RD and Grinstein GG. A reference model for the visualisation of multi-dimensional data. In: *Proceedings of the European computer graphics conference and exhibition (EG'89)* (ed W Hansmann, FRA Hopgood and W Strasser), Hamburg, 4–8 September, pp. 393–399. Goslar: Eurographics Association.
93. Rankin R. A taxonomy of graph types. *Inform Des J* 1990; 6(2): 147–159.
94. Kandel S, Parikh R, Paepcke A, et al. Profiler: integrated statistical analysis and visualization for data quality assessment. In: *Proceedings of the working conference on*

advanced visual interfaces (AVI'12) (ed G Tortora, S Levialdi and M Tucci), Capri Island, 21–25 May 2012, pp. 547–554. New York: ACM.
95. Wills G and Wilkinson L. AutoVis: automatic visualization. *Inform Visual* 2010; 9(1): 47–69.
96. Simmhan YL, Plale B and Gannon D. Karma2: provenance management for data driven workflows. In: Zhang LJ (ed.) *Web services research for emerging applications: discoveries and trends*. Hershey, PA: IGI Global, 2008, pp. 317–339.
97. Ludäscher B, Altintas I, Berkley C, et al. Scientific workflow management and the Kepler system. *Concurr Comp: Pract E* 2006; 18(10): 1039–1065.
98. Belhajjame K, Wolstencroft K, Corcho O, et al. Metadata management in the Taverna workflow system. In: *Proceedings of the IEEE international symposium on cluster computing and the grid (CCGRID'08)* (ed T Priol, L Lefevre and R Buyya), Lyon, 19–22 May 2008, pp. 651–656. New York: IEEE.
99. Callahan SP, Freire J, Scheidegger CE, et al. Towards provenance-enabling ParaView. In: *Proceedings of the provenance and annotation workshop (IPAW'08)* (ed J Freire, D Koop and L Moreau; Number 5272 in lecture notes in computer science), Salt Lake City, UT, 17–18 June 2008, pp. 120–127. Berlin: Springer.
100. Huq MR, Apers PMG and Wombacher A. ProvenanceCurious: a tool to infer data provenance from scripts. In: *Proceedings of the conference on extending database technology (EDBT'13)* (ed NW Paton, G Guerrini, B Catania, et al.), Genoa, 18–22 March 2013, pp. 765–768. New York: ACM.
101. Cowley P, Nowell L and Scholtz J. Glass Box: an instrumented infrastructure for supporting human interaction with information. In: *Proceedings of the Hawaii conference on system sciences (HICSS'05)*, Big Island, HI, 6 January 2005, p. 296c. New York: IEEE.
102. Cowley P, Haack J, Littlefield R, et al. Glass box: capturing, archiving, and retrieving workstation activities. In: *Proceedings of the ACM workshop on continuous archival and retrieval of personal experiences (CARPE'06)*, Santa Barbara, CA, 28 October 2006, pp. 13–18. New York: ACM.
103. Wickham H. Tidy data. *J Stat Softw* 2014; 59(10): 1–23.
104. Card SK, Mackinlay J and Shneiderman B. *Readings in information visualization: using vision to think*. San Francisco, CA: Morgan Kaufmann, 1999.
105. Glavic B, Miller RJ and Alonso G. Using SQL for efficient generation and querying of provenance information. In: Tannen V, Wong L, Libkin L, et al. (eds) *In search of elegance in the theory and practice of computation* (Lecture notes in computer science), vol. 8000. Berlin: Springer, 2013, pp. 291–320.
106. Asuncion HU. In situ data provenance capture in spreadsheets. In: *Proceedings of the IEEE conference on eScience*, Stockholm, 5–8 December 2011, pp. 240–247. New York: IEEE.
107. Borek A, Woodall P, Oberhofer M, et al. A classification of data quality assessment methods. In: *Proceedings of the international conference on information quality (ICIQ'11)*, Adelaide, SA, Australia, 18–20 November 2011, pp. 189–203.
108. Kandel S, Heer J, Plaisant C, et al. Research directions in data wrangling: visualizations and transformations for usable and credible data. *Inform Visual* 2011; 10(4): 271–288.
109. Galhardas H, Florescu D, Shasha D, et al. AJAX: an extensible data cleaning tool. *SIGMOD Rec* 2000; 29(2): 590–596.
110. Raman V and Hellerstein JM. Potter's wheel: an interactive data cleaning system. In: *Proceedings of the conference on very large data bases (VLDB'01)*, Roma, 11–14 September 2001, pp. 381–390. San Francisco, CA: Morgan Kaufmann Publishers.
111. Kandel S, Paepcke A, Hellerstein J, et al. Wrangler: interactive visual specification of data transformation scripts. In: *Proceedings of the international conference on human factors in computing systems (CHI'11)*, Vancouver, BC, Canada, 7–12 May 2011, pp. 3363–3372. New York: ACM.
112. Robertson PK. A methodology for scientific data visualisation: choosing representations based on a natural scene paradigm. In: *Proceedings of the IEEE conference on visualization (VIS'90)* (ed A Kaufman), San Francisco, CA, 23–26 October 1990, pp. 114–123. New York: IEEE.
113. Blommestijn SQ and Peerbolte EAL. Outliers and extreme observations: what are they and how to handle them? In: Adèr HJ and Mellenbergh GJ (eds) *Advising on research methods—selected topics 2012*. Huizen: Johannes van Kessel Publishing, 2012, pp. 81–105.
114. Dos Santos S and Brodlie K. Gaining understanding of multivariate and multidimensional data through visualization. *Comput Graph* 2004; 28(3): 311–325.
115. MacEachren AM. Exploring high-D spaces with multiform matrices and small multiples. In: *Proceedings of the IEEE symposium on information visualization (InfoVis'03)* (ed T Munzner and S North), Seattle, WA, 19–21 October 2003, pp. 31–38. New York: IEEE.
116. Yang J, Hubball D, Ward MO, et al. Value and relation display: interactive visual exploration of large data sets with hundreds of dimensions. *IEEE T Vis Comput Gr* 2007; 13(3): 494–507.
117. New JR. *Visual analytics for relationships in scientific data*. PhD Thesis, University of Tennessee, Knoxville, TN, 2009.
118. Jones MB, Schildhauer MP, Reichman OJ, et al. The new bioinformatics: integrating ecological data from the gene to the biosphere. *Annu Rev Ecol Evol S* 2006; 37: 519–544.
119. Hurt RL, Gauthier AJ, Christensen LL, et al. Sharing images intelligently: the astronomy visualization metadata standard. In: *Proceedings of the conference on communicating astronomy with the public (CAP'07)* (ed LL Christensen, M Zoulias and I Robson), Athens, 8–11 October 2007, pp. 450–453. Palaio Faliro: Eugenides Foundation.
120. Stanford NJ, Wolstencroft K, Golebiewski M, et al. The evolution of standards and data management

practices in systems biology. *Mol Syst Biol* 2015; 11(12): 851.
121. Jones MB, Berkley C, Bojilova J, et al. Managing scientific metadata. *IEEE Internet Comput* 2001; 5(5): 59–68.
122. Berkley C, Bowers S, Jones MB, et al. Improving data discovery for metadata repositories through semantic search. In: *Proceedings of the international conference on complex, intelligent and software intensive systems (CISIS'09)* (ed L Barolli, F Xhafa and HH Hsu), Fukuoka, Japan, 16–19 March 2009, pp. 1152–1159. New York: IEEE.
123. Xiao B, Zhang C, Mao Y, et al. Review and exploration of metadata management in data warehouse. In: *Proceedings of the IEEE conference on industrial electronics and applications (ICIEA'15)*, Auckland, New Zealand, 15–17 June 2015, pp. 928–933. New York: IEEE.
124. Petoukhov V, Ganopolski A, Brovkin V, et al. CLIMBER-2: a climate system model of intermediate complexity. Part I: model description and performance for present climate. *Clim Dynam* 2000; 16(1): 1–17.
125. Wong PC, Foote H, Leung R, et al. Vector fields simplification—a case study of visualizing climate modeling and simulation data sets. In: *Proceedings of the IEEE conference on visualization (VIS'00)* (ed T Ertl, B Hamann and A Varshney), Salt Lake City, UT, 8–13 October 2000, pp. 485–488. New York: IEEE.
126. Moorhead RJ and Zhu Z. Feature extraction for oceanographic data using a 3D edge operator. In: *Proceedings of the IEEE conference on visualization (VIS'93)* (ed GM Nielson and D Bergeron), San Jose, CA, 25–29 October 1993, pp. 402–405. New York: IEEE.
127. Ma KL and Smith PJ. Cloud tracing in convection-diffusion systems. In: *Proceedings of the IEEE conference on visualization (VIS'93)* (ed GM Nielson and D Bergeron), San Jose, CA, 25–29 October 1993, pp. 253–260. New York: IEEE.
128. Nocke T, Flechsig M and Böhm U. Visual exploration and evaluation of climate-related simulation data. In: *Proceedings of the winter simulation conference (WSC'07)* (ed SG Henderson, B Biller, MH Hsieh, et al.), Washington, DC, 9–12 December 2007, pp. 703–711. New York: IEEE.
129. Rockel B, Will A and Hense A. The regional climate model COSMO-CLM (CCLM). *Meteorol Z* 2008; 17(4): 347–348.
130. Thompson DL, Braun JA and Ford R. *OpenDX—paths to visualization*. 2nd ed. Missoula, MT: Visualization and Imagery Solutions, Inc., 2004.
131. Hankin S, Harrison E, Osborne J, et al. A strategy and a tool, Ferret, for closely integrated visualization and analysis. *J Visual Comp Animat* 1996; 7(3): 149–157.
132. Kerne A, Qu Y, Webb AM, et al. Meta-metadata: a metadata semantics language for collection representation applications. In: *Proceedings of the ACM international conference on information and knowledge management (CIKM'10)* (ed XJ Huang, G Jones, N Koudas, et al.), Toronto, ON, Canada, 26–30 October 2010, pp. 1129–1138. New York: ACM.
133. Keim DA, Mansmann F, Schneidewind J, et al. Challenges in visual data analysis. In: *Proceedings of the international conference on information visualization (IV'06)* (ed E Banissi, RA Burkhard, A Ursyn, et al.), London, UK, 5–7 July 2006, pp. 9–16. New York: IEEE.
134. Patro A, Ward MO and Rundensteiner EA. *Seamless integration of diverse data types into exploratory visualization systems*. Technical report WPI-CS-TR-03-12, April 2003. Worcester, MA: Worcester Polytechnic Institute.