Exploration and Visualization in the Web of Big Linked Data: A Survey of the State of the Art [*]

Nikos Bikakis
NTU Athens &
ATHENA R.C.
Greece

Timos Sellis
Swinburne Univ.
of Technology
Australia

ABSTRACT

Data exploration and visualization systems are of great importance in the *Big Data* era. Exploring and visualizing very large datasets has become a major research challenge, of which scalability is a vital requirement. In this survey, we describe the major prerequisites and challenges that should be addressed by the modern exploration and visualization systems. Considering these challenges, we present how *state-of-the-art* approaches from the *Database* and *Information Visualization* communities attempt to handle them. Finally, we survey the systems developed by *Semantic Web* community in the context of the *Web of Linked Data*, and discuss to which extent these satisfy the contemporary requirements.

Keywords

Visual analytics, big data challenges, data exploration, large databases, visual exploration, semantic web, visualization tools, scalability

1. INTRODUCTION

The purpose of *data exploration* and *visualization* is to offer ways for information perception and manipulation, as well as knowledge extraction and inference [68, 56]. Data visualization[1] provides users with an intuitive means to explore the content of the data, identify interesting patterns, infer correlations and causalities, and supports sense-making activities. Data exploration and visualization systems are of great importance in the *Big Data* era, in which the volume and heterogeneity of available information make it difficult for humans to manually explore and analyse data.

Most traditional systems cannot handle the large size of many contemporary datasets. Exploring and visualizing large datasets has become a major research challenge [24, 119, 55, 103, 140, 49]. Therefore, modern systems have to take into account *scalability*, as a vital requirement. Dealing with scalability, modern systems have to address numerous issues related to storage, access, rendering/presentation, interaction, etc.

In the *Web of Data* (WoD) context, following the abundance of *Linked Data*, several recent efforts have offered tools and techniques for exploration and visualization in many different domains [35]. However, most of these approaches fail to take into account issues related to performance and scalability.

In this work, we describe the major requirement and challenges that should be addressed by the modern exploration and visualization systems. Additionally, we refer to state-of-the-art approaches from the Database and Information Visualization communities, which attempt to handle some of these challenges. Further, we describe the systems that have been developed in the context of WoD, and discuss to which extent they satisfy the contemporary requirements.

2. CHALLENGES

Most traditional exploration and visualization systems operate in an *offline* way, limited to accessing static sets of preprocessed data. Additionally, they restrict themselves to dealing with *small* dataset sizes, which can be easily handled and explored with conventional data management and (visual) explorations techniques.

On the other hand, nowadays, the Big Data era has realized the availability of the great number and variety of *very large* datasets that are *dynamic* in nature. For example, most data sources offer query or API endpoints for online access and updates; in other cases (e.g., scientific databases), new data is constantly arrived (e.g., on a daily/hour basis). Beyond these, modern systems should operate on exploratory context. In an *exploration scenario*, it common that users are interesting in finding something interesting and useful without previously know what exactly are searching for, until the time they identify it. In this case, users perform a sequence of operations (e.g., queries), in which the result of each operation determine the formulation of the next operation. Finally, an increasingly large number of *diverse users* (i.e., different preferences, skills, etc.) explore and analyse data in a plethora of *different scenarios*.

Therefore, some of the major challenges that should be dealt with by modern systems, are posed by the: (1) Large size and the dynamic nature of data in conjunction with the exploration-driven setting; and (2) Variety of tasks and users.

Large & Dynamic Data in Exploration-driven Setting. One of the major challenges in exploration and visualization is related to the *size* that characterizes most contemporary datasets. A second challenge is related to the availability of query and API endpoints for online data access and retrieval, as well as the cases where that data is received in a stream fashion. The later pose the challenge of handling large sets of data in a *dynamic* setting, and as a result, a preprocessing phase (e.g., traditional indexing) is prevented.

In this respect, modern visualization and exploration systems must be able to *efficiently* and *effectively* handle *billion objects dynamic datasets* throughout an *exploratory scenario*. Therefore, scalable and efficient data structures and algorithms have to be developed. Crucial issues related to storage, access, management,

[*]This paper appears in 6th International Workshop on Linked Web Data Management (LWDM 2016).

[1]Throughout this paper we use the term "visualization" referring to visual data exploration.

LWDM '16 March 15, 2016, Bordeaux, France

presentation, interaction (e.g., pan, zoom, search, drill-down), etc. over large dynamic datasets have to be handled. Scalability has become a major challenge for the modern systems. Beyond these, systems have to efficiently operate on machines with limited computational and memory resources (e.g., laptops).

In a "conventional" setting (e.g., explore a small fragment of a preprocessed dataset), most of the aforementioned issues can be handled by the traditional systems that provide database exploration and analysis, such as Tableau[2] (previously know as Polaris [124]), DEVise [98], Spotfire [3], VisDB [81], Lumira[3], QlikView[4], Datawatch[5], etc. However, in a "modern" setting, when a large part (or the whole) of a billion objects dynamic dataset has to be explored, the aforementioned traditional database-oriented systems cannot be adopted.

In conjunction with performance issues, modern systems have to address challenges related to visual presentation and interaction issues. Particularly, systems should be able to present, as well as, offer ways to "easily" explore large datasets. Handling a large number of data objects is a challenging task; modern systems have to "*squeeze a billion records into a million pixels*" [119]. Even, in much smaller datasets, offering a dataset overview is extremely difficult; in both cases *information overloading* is a common issue. As aslo stated in the visual information seeking mantra: "*overview first, zoom and filter, then details on demand*" [118], gaining *overview* is crucial in the visual exploration scenario. Based on the aforementioned, it follows that a basic requirement of the modern systems is to develop methods that provide *summaries* and *abstractions* over the enormous number of data objects.

In order to tackle both performance and presentations issues, a large number of systems adopt *approximation techniques* (a.k.a. *data reduction* techniques) in which partial results are computed. Considering the existing approaches, most of them are based on: (1) sampling and filtering [46, 105, 2, 69, 17]; or/and (2) aggregation (e.g., binning, clustering) [42, 25, 74, 73, 97, 138, 96, 1, 15, 71]. In this respect, some modern database-oriented systems adopt approximation techniques using query-based approaches (e.g., query translation, query rewriting) [17, 74, 73].

In order to improve efficiency several systems adopt *incremental* (a.k.a. *progressive*) techniques. In these techniques the results/visual elements are computed/constructed incrementally based on user interaction or as time progresses (e.g., [123, 25]). Numerous recent systems integrate incremental and approximate techniques, in these approaches, approximate answers are computed incrementally over progressively larger samples of the data [46, 2, 69].

The dynamic setting prevents modern systems from preprocessed the data. Additionally, it is common in exploration scenarios only a small fragment of data to be accessed by the user. In this context, an adaptive indexing approach [67] is used in [144], where the indexes are created incrementally and adaptively throughout exploration. Similarly, in [25] the hierarchy tree is incrementally constructed based on user's interaction. Finally, in some approaches, parallel architectures are adopted; e.g., [41, 78, 77, 69].

To sum up, modern systems should provide scalable techniques that on-the-fly effectively (i.e., in a way that can be easily explored) handle a large number of data objects over an exploration scenario, using a limited number of resources

Variety of Tasks & Users. The requirement of scalable, on-the-fly exploration and analysis must be coupled with the diversity of preferences and requirements posed by different users and tasks.

Therefore, the modern systems should provide the user with the ability to customize the exploration experience based on her preferences and the requirements posed by the examined task. For example, systems should allow the user to: (1) organize data into different ways, according to the type of information or the level of detail she wishes to explore (e.g., [25]); (2) modify approximation criteria, thresholds, sampling rates, etc. (e.g., [78]); (3) define her own operations for data manipulation and analysis (e.g., aggregation, statistical, filtering functions), etc. Furthermore, systems should automatically adjust their parameters, by taking into account the environment setting (e.g., screen resolution, memory size) [74, 25, 73].

Beyond the personalization, modern systems should provide mechanisms that assist the user and reduce the effort needed on their part. In this direction, several approaches have been recently developed. In what follows, we mention some of the most common ones. Several systems assist users by recommending visualization that seems to be more useful or capture surprising and/or interesting data; e.g., [139, 134, 82]. Other approaches help users to discover interest areas in the dataset; by capturing user interests, they guide her to interesting data parts; e.g., [37]. Finally, in other cases systems provide explanations regarding data trends and anomalies; e.g., [141].

3. EXPLORATION & VISUALIZATION SYSTEMS

This section reviews works related to exploration and visualization in the WoD. A large number of works studying issues related to WoD visual exploration and analysis have been proposed in the literature [35, 101, 4]. In what follows, we classify these works into the following categories: (1) Browsers and exploratory systems (Section 3.1), (2) Generic visualization systems (Section 3.2), (3) Domain, vocabulary & device-specific visualization systems (Section 3.3), (4) Graph-based visualization systems (Section 3.4), (5) Ontology visualization systems (Section 3.5), and (6) Visualization libraries (Section 3.6).

3.1 Browsers & Exploratory Systems

WoD browsers have been the first systems developed for WoD utilization and analysis [35, 4]. Similarly to the traditional ones, WoD browsers provide the functionality for link navigation and representation of WoD resources and their properties; thus enabling browsing and exploration of WoD in a most intuitive way. WoD browsers mainly use tabular views and links to provide navigation over the WoD resources.

Haystack [111] is one of the first WoD browsers, it exploits stylesheets in order to customize the data presentation. Similarly, *Disco*[6] renders all information related to a particular RDF resource as HTML table with property-value pairs. *Noadster* [113] performs property-based data clustering in order to structure the results. *Piggy Bank* [66] is a Web browser plug-in, that allows users to convert HTML content into RDF. *LESS* [13] allows users to create their own Web-based templates in order to aggregate and display WoD. *Tabulator* [21] another WoD browser, additionally provides maps and timeline visualizations. *LENA* [87] provides different views of data, following user's criteria that are expressed as SPARQL queries. *Visor* [110] provides a multi-pivot approach for exploring graphs, allowing users to explore multiple nodes at a time, as well as to connect points of interest. Finally, in the context

[2] tableau.com
[3] sap-lumira.com
[4] clickview.com
[5] datawatch.com

[6] www4.wiwiss.fu-berlin.de/bizer/ng4j/disco

Table 1: Generic Visualization Systems

System	Year	Data Types⋆	Vis. Types⋆⋆	Recomm.	Preferences	Statistics	Sampling	Aggregation	Incr.	Disk	Domain	App. Type
Rhizomer [30]	2006	N, T, S, H, G	C, M, T, TL	✓							generic	Web
VizBoard [135, 136, 109]	2009	N, H	C,S, T	✓	✓		✓				generic	Web
LODWheel [126]	2011	N, S, G	C, G, M, P								generic	Web
SemLens [59]	2011	N	S		✓						generic	Web
LDVM [29]	2013	S, H, G	B, M, T, TR	✓							generic	Web
Payola [84]	2013	N, T, S, H, G	C, CI, G, M, T, TL, TR								generic	Web
LDVizWiz [11]	2014	S, H, G	M, P, TR	✓							generic	Web
SynopsViz [26, 25]	2014	N, T, H	C, P, T, TL	✓	✓	✓		✓	✓	✓	generic	Web
Vis Wizard [131]	2014	N, T, S	B, C, M, P, PC, SG	✓	✓						generic	Web
LinkDaViz [129]	2015	N, T, S	B, C, S, M, P	✓	✓						generic	Web
ViCoMap [112]	2015	N, T, S	M			✓					generic	Web

⋆ N: Numeric, T: Temporal, S: Spatial, H: Hierarchical (tree), G: Graph (network)
⋆⋆ B: *bubble chart*, C: *chart*, CI: *circles*, G: *graph*, M: *map*, P: *pie*, PC: *parallel coordinates*, S: *scatter*, SG: *streamgraph*, T: *treemap*, TL: *timeline*, TR: *tree*

of faceted browsing, */facet* [62], *Humboldt* [86] and *gFacet* [57] provide faceted navigation over WoD resources.

Explorator [7] is a WoD exploratory tool that allows users to browse a dataset by combining search and facets. *VisiNav* [53] is a system that allows users to pose expressive exploratory-based queries. The system is built on top of following concepts: keyword search, object focus, path traversal, and facet selection. *Information Workbench* (IWB) [52] is a generic platform for semantic data management offering several back-end (e.g., triple store) and front-end tools. Regarding the front-end, IWB offers a flexible user interface for data exploration and visualization. *Marbles*[7] formats RDF triples using the Fresnel vocabulary (a vocabulary for rendering RDF resources as HTML). Also, it retrieves information about a resource by accessing Semantic Web indexes and search engines. Finally, *URI Burner*[8] is a service which retrieves data about resources. For the requested resources, it generates an RDF graph by exploiting existing ontologies and other knowledge from the Web.

3.2 Generic Visualization Systems

In the context of WoD visual exploration, there is a large number of generic visualization frameworks, that offer a wide range of visualization types and operations. Next, we outline the best known systems in this category.

In Table 1 we provide an overview and compare several generic visualization systems. The *Year* column presents the released date. The *Data Types* column specifies the supported data types. The *Vis. Types* column presents the types of visualizations that are provided. The *Recomm*. column indicates systems that offer recommendation mechanisms for visualization settings (e.g., appropriate visualization type, visualization parameters). The *Preferences* column captures the ability of the users to apply data (e.g., filter, aggregate) or visual (e.g., increase abstraction) operations. The *Statistics* column captures the provision of statistics about the visualized data. The *Sampling* column indicates systems that exploit techniques based on sampling and/or filtering. The *Aggregation* column indicates systems that exploit techniques based on aggregation (e.g., binning, clustering). The *Incr*. column indicates systems that adopt incremental techniques; i.e., the results/visualization are computed/generated based on user interaction or as time progresses. Finally, the *Disk* column indicates systems that use external memory (e.g., file, database) to perform operations during runtime (i.e., not just initially load data from disk).

Rhizomer [30] provides WoD exploration based on a overview, zoom and filter workflow. Rhizomer offers various types of visualizations such as maps, timelines, treemaps and charts. *VizBoard* [135, 136, 109] is an information visualization workbench for WoD build on top of a mashup platform. VizBoard presents datasets in a dashboard-like, composite, and interactive visualization. Additionally, the system provides visualization recommendations. *Payola* [84] is a generic framework for WoD visualization and analysis. The framework offers a variety of domain-specific (e.g., public procurement) analysis plugins (i.e., analyzers), as well as several visualization techniques (e.g., graphs, tables). In addition, Payola offers collaborative features for users to create and share analyzers. In Payola the visualizations can be customized according to ontologies used in the resulting data.

The *Linked Data Visualization Model* (LDVM) [29] provides an abstract visualization process for WoD datasets. LDVM enables the connection of different datasets with various kinds of visualizations in a dynamic way. The visualization process follows a four stage workflow: Source data, Analytical abstraction, Visualization abstraction, and View. LDVM considers several visualization techniques, e.g., circle, sunburst, treemap, etc. Finally, the LDVM has been adopted in several use cases [85]. *Vis Wizard* [131] is a Web-based visualization system, which exploits data semantics to simplify the process of setting up visualizations. Vis Wizard is able to analyse multiple datasets using brushing and linking methods. Similarly, *Linked Data Visualization Wizard* (LDVizWiz) [11] provides a semi-automatic way for the production of possible visualization for WoD datasets. In a same context, *LinkDaViz* [129] finds the suitable visualizations for a give part of a dataset. The framework uses heuristic data analysis and a visualization model in order to facilitate automatic binding between data and visualization options.

Balloon Synopsis [117] provides a WoD visualizer based on HTML and JavaScript. It adopts a node-centric visualization approach in a tile design. Additionally, it supports automatic information enhancement of the local RDF data by accessing either remote SPARQL endpoints or performing federated queries over endpoints using the Balloon Fusion service [116]. Balloon Synopsis offers customizable filters, namely ontology templates, for the users to handle and transform (e.g., filter, merge) input data. *LODWheel* [126] is a Web-based visualizing tool which combines JavaScript libraries (e.g., MooWheel, JQPlot) in order to visualize RDF data in charts and graphs. *SemLens* [59] is a visual tool that combines scatter plots and semantic lenses, offering visual discovery of correlations and patterns in data. Objects are arranged in a scatter plot and are analysed using user-defined semantic lenses. *ViCoMap*

[7] mes.github.io/marbles
[8] linkeddata.uriburner.com

[112] combines WoD statistical analysis and visualization, in a Web-based tool, which offers correlation analysis and data visualization on maps.

Finally, *SynopsViz* [26, 25] is a Web-based visualization tool built on top of a generic tree-based model. The adopted model performs a hierarchical aggregation, allowing efficient personalized multilevel exploration over large datasets. In order to provide scalability under different exploration scenarios, the model offers a method that incrementally constructs the hierarchy based on user's interaction, as well as a method that enables dynamic and efficient adaptation of the hierarchy to the user's preferences.

3.3 Domain, Vocabulary & Device-specific Visualization Systems

In this section, we present systems that target visualization needs for specific types of data and domains, RDF vocabularies or devices.

Several systems focus on visualizing and exploring geo-spatial data. *Map4rdf* [92] is a faceted browsing tool that enables RDF datasets to be visualized on an OSM or Google Map. *Facete* [122] is an exploration and visualization tool for SPARQL accessible data, offering faceted filtering functionalities. *SexTant* [20] and *Spacetime* [133] focus on visualizing and exploring time-evolving geo-spatial data. The *LinkedGeoData Browser* [121] is a faceted browser and editor which is developed in the context of LinkedGeoData project. Finally, in the same context *DBpedia Atlas* [132] offers exploration over the DBpedia dataset by exploiting the dataset's spatial data. Furthermore, in the context of linked university data, *VISUalization Playground* (VISU) [6] is an interactive tool for specifying and creating visualizations using the contents of linked university data cloud. Particularly, VISU offers a novel SPARQL interface for creating data visualizations. Query results from selected SPARQL endpoints are visualized with Google Charts.

A variety of systems target multidimensional WoD modelled with the Data Cube vocabulary. *CubeViz* [43, 114] is a faceted browser for exploring statistical data. The tool provides data visualizations using different types of charts (i.e., line, bar, column, area and pie). The *Payola Data Cube Vocabulary* [60] adopts the LDVM stages [29] in order to visualize RDF data described by the Data Cube vocabulary. The same types of charts as in CubeViz are provided in this tool. The *OpenCube Toolkit* [75] offers several tools related to statistical WoD. For example, *OpenCube Browser* explores RDF data cubes by presenting a two-dimensional table. Additionally, the *OpenCube Map View* offers interactive map-based visualizations of RDF data cubes based on their geo-spatial dimension. The *Linked Data Cubes Explorer* (LDCE) [79] allows users to explore and analyse statistical datasets. Finally, [106] offers several map and chart visualizations of demographic, social and statistical linked cube data.

Regarding device-specific systems, *DBpedia Mobile* [18] is a location-aware mobile application for exploring and visualizing DBpedia resources. *Who's Who* [32] is an application for exploring and visualizing information focusing on several issues that appear in the mobile environment. For example, the application considers the usability and data processing challenges related to the small display size and limited resources of the mobile devices.

3.4 Graph-based Visualization Systems

A large number of systems visualize WoD datasets adopting a *graph-based* (a.k.a., node-link) approach [102]. In Table 2 we provide an overview and compare several graph-based visualization systems. Table 2 is structured in a similar way to Table 1. Additionally, in this table the *Keyword* column indicates systems that provide keyword search functionality. The *Filter* column indicates systems that provide mechanisms for data filtering. Note that, Table 2 also includes the ontology visualization systems (Section 3.5) that follow a node-link approach (indicated by using the term "ontology" in the Domain column).

RelFinder [58] is a Web-based tool that offers interactive discovery and visualization of relationships (i.e., connections) between selected WoD resources. *Fenfire* [54] and *Lodlive* [31] are exploratory tools that allow users to browse WoD using interactive graphs. Starting from a given URI, the user can explore WoD by following the links. *LODeX* [19] is a tool that generates a representative summary of a WoD source. The tool takes as input a SPARQL endpoint and generates a visual (graph-based) summary of the WoD source, accompanied by statistical and structural information of the source. *IsaViz* [108] allows users to zoom and navigate over the RDF graph, and also it offers several "edit" operations (e.g., delete/add/rename nodes and edges). In the same context, *graphVizdb* [23, 22] is built on top of spatial and database techniques offering interactive visualization over very large (RDF) graphs. *ZoomRDF* [142] employs a space-optimized visualization algorithm in order to increase the number of resources which are displayed. *Trisolda* [38] proposes a hierarchical RDF graph visualization. It adopts clustering techniques in order to merge graph nodes. *Paged Graph Visualization* (PGV) [36] utilizes a Ferris-Wheel approach to display nodes with high degree. *RDF graph visualizer* [115] adopts a node-centric approach to visualize RDF graphs. Rather than trying to visualize the whole graph, nodes of interest (i.e., staring nodes) are discovered by searching over nodes labels; then the user can interactively navigate over the graph. *RDF-Gravity*[9] visualizes RDF and OWL data. It offers filtering, keyword search and editing the graph layout. Also, the nodes can be displayed in different colors and shapes based on their RDF types. A different approach has been adopted in [127], where sampling techniques have been exploited. Finally, *Gephi* [15] is a generic tool that offers several visualization and analysis features over graph data.

3.5 Ontology Visualization Systems

The problems of *ontology visualization and exploration* have been extensively studied in several research areas (e.g., biology, chemistry). In what follows we focus on graph-based ontology visualization systems that have been developed in the WoD context [47, 40, 51, 91, 80]. In most systems, ontologies are visualized following the node-link paradigm [100, 99, 64, 104, 27, 45, 65, 94, 5, 89, 125] [10,11]. On the other hand, *CropCircles* [137] uses a geometric containment approach, representing the class hierarchy as a set of concentric circles. Furthermore, hybrids approaches are adopted in other works. *Knoocks* [88] combines containment-based and node-link approaches. In this work, ontologies are visualized as nested blocks where each block is depicted as a rectangle containing a sub-branch shown as tree map. Finally, *OntoTrix* [14] and *NodeTrix* [61] use node-link and adjacency matrix representations.

3.6 Visualization Libraries

Finally, there is a variety of Javascript libraries which allow WoD visualizations to be embedded in Web pages. *Sgvizler* [120] is a JavaScript wrapper for visualizing SPARQL results. Sgvizler allows users to specify SPARQL Select queries directly into HTML elements. Sgvizler uses Google Charts to generate the output, offering numerous visualizations types such as charts, treemaps, graphs,

[9] semweb.salzburgresearch.at/apps/rdf-gravity

[10] protegewiki.stanford.edu/wiki/OntoGraf

[11] protegewiki.stanford.edu/wiki/OWLViz

Table 2: Graph-based Visualization Systems

System	Year	Keyword	Filter	Sampling	Aggregation	Incr.	Disk	Domain	App. Type
RDF-Gravity [9]	2003	✓	✓					generic	Desktop
IsaViz [108]	2003	✓	✓					generic	Desktop
RDF graph visualizer [115]	2004	✓						generic	Desktop
GrOWL [89]	2007	✓	✓	✓				ontology	Desktop
NodeTrix [61]	2007				✓			ontology	Desktop
PGV [36]	2007					✓	✓	generic	Desktop
Fenfire [54]	2008							generic	Desktop
Gephi [15]	2009		✓	✓	✓			generic	Desktop
Trisolda [38]	2010			✓	✓	✓		generic	Desktop
Cytospace [127]	2010	✓	✓	✓	✓		✓	generic	Desktop
FlexViz [45]	2010	✓	✓					ontology	Web
RelFinder [58]	2010							generic	Web
ZoomRDF [142]	2010			✓	✓	✓		generic	Desktop
KC-Viz [104]	2011			✓				ontology	Desktop
LODWheel [126]	2011		✓		✓			generic	Web
GLOW [64]	2012			✓	✓			ontology	Desktop
Lodlive [31]	2012	✓						generic	Web
OntoTrix [14]	2013			✓	✓			ontology	Desktop
LODeX [19]	2014			✓	✓			generic	Web
VOWL 2 [100, 99]	2014							ontology	Web
graphVizdb [23, 22]	2015	✓	✓	✓			✓	generic	Web

timelines, etc. *Visualbox* [50] provides an environment where users can build and debug SPARQL queries in order to retrieve WoD; then, a set of visualization templates is provided to visualize results. Visualbox uses several visualization libraries like Google Charts and D3 [28], offering 14 visualization types.

4. DISCUSSION

In this section we discuss to which extent the systems developed in the WoD context fulfilled the nowadays requirements, focussing on performance and scalability issues, availability of personalized services facilities for assisting users through exploration.

As previously mentioned, most of WoD exploration and visualization systems do not handle issues related to performance and scalability. They basically adopt traditional techniques in order to handle small sets of data.

As we can observe from Table 1, generic systems support several types of data (e.g., numeric, temporal, graph, spatial) and provide a plethora of visualization types. Additionally, an increasing number of recent systems (e.g., LinkDaViz, Vis Wizard, LDVizWiz, LDVM) focus on providing recommendation mechanisms. Particularity, these systems mainly recommend the most suitable visualization technique by considering the type of input data.

Regarding visual scalability, as we can see in Table 1, none of the systems, with the exceptions of SynopsViz and VizBoard cases, adopt approximation techniques (i.e., sampling/filtering, aggregation). Hence, the existing approaches assume that all the examined data objects can be presented on the screen and handled by traditional visualization techniques. Due to this assumption, the current systems restrict their applicability to small sets of data.

In conjunction with the limited visual scalability, most of the existing systems (except for SynopsViz) do not exploit external memory during runtime. Particularly, they initially load all the examined objects in main memory, assuming that the main memory is large enough. An alternative approach is adopted by the SynopsViz system, which incrementally retrieves data and generates visualizations based on user interaction. As a result, each time, only a part of the examined dataset needs to be loaded in main memory.

The graph-based exploration and visualization systems are presented in Table 2. These systems are of great importance in WoD, due to the graph structure of the RDF data model. Although several systems offer sampling or aggregation mechanisms, most of these systems load the whole graph in main memory. Given the large memory requirements of graph layout algorithms in order to draw a large graph, the current WoD systems are restricted to handle small sized graphs.

In order to be able to handle large graphs, modern WoD systems should adopt more sophisticated techniques similar to those proposed by the information visualization community. Particularly, state-of-the-art systems for exploring large graphs utilize hierarchical aggregation approaches where the graph is recursively decomposed into smaller sub-graphs (in most cases using clustering and partitioning) that form a hierarchy of abstraction layers [93, 10, 95, 9, 8, 1, 143, 12, 15, 71, 130]. Other approaches adopt edge bundling techniques which aggregate graph edges to bundles [48, 44, 107, 90, 34, 63]. Beyond hierarchical approaches, WoD systems should also consider disk-based implementations, such as [22, 1, 72, 127, 130].

To sum up, WoD community should consider scalability and performance as vital requirements for the development of the future exploration and visualization systems. Handing large datasets is crucial in the Big Data era. Therefore, in what follows we summarize some possible directions for the future WoD exploration and visualization systems. Approximation techniques such as sampling and aggregation that have been widely used in systems from database and information visualization communities, have to be adopted and adjusted to WoD data and requirements. Systems should be integrated with disk structures, retrieving data dynamically during runtime. Also caching and prefetching techniques may be exploited; e.g., [128, 76, 70, 16, 33, 83, 39]. Data structures and indexes should be developed focusing on WoD tasks and data, such as Nanocubes [96] in the context of spatio-temporal data exploration, and HETree [25] in numeric and temporal datasets. Finally, considering users' perspective, beyond visualization recommendations, modern WoD systems should provide more sophisticated mechanisms that capture users' preferences and assist them throughout large data exploration and analysis tasks.

5. REFERENCES

[1] J. Abello, F. van Ham, and N. Krishnan. ASK-GraphView: A Large Scale Graph Visualization System. *TVCG*, 12(5), 2006.

[2] S. Agarwal, B. Mozafari, A. Panda, H. Milner, S. Madden, and I. Stoica. BlinkDB: Queries with Bounded Errors and Bounded Response Times on Very Large Data. In *EuroSys*, 2013.

[3] C. Ahlberg. Spotfire: An Information Exploration Environment. *SIGMOD Record*, 25(4), 1996.

[4] F. Alahmari, J. A. Thom, L. Magee, and W. Wong. Evaluating Semantic Browsers for Consuming Linked Data. In *ADC*, 2012.

[5] H. Alani. TGVizTab: An Ontology Visualisation Extension for Protege. In *Workshop on Visualizing Information in Knowledge Engineering*, 2003.

[6] M. Alonen, T. Kauppinen, O. Suominen, and E. Hyvönen. Exploring the Linked University Data with Visualization Tools. In *ESWC*, 2013.

[7] S. F. C. Araújo, D. Schwabe, and S. D. J. Barbosa. Experimenting with Explorator: a Direct Manipulation Generic RDF Browser and Querying Tool. In *Visual Interfaces to the Social and the Semantic Web*, 2009.

[8] D. Archambault, T. Munzner, and D. Auber. Grouse: Feature-Based, Steerable Graph Hierarchy Exploration. In *EuroVis*, 2007.

[9] D. Archambault, T. Munzner, and D. Auber. GrouseFlocks: Steerable Exploration of Graph Hierarchy Space. *TVCG*, 14(4), 2008.

[10] D. Archambault, T. Munzner, and D. Auber. Tugging Graphs Faster: Efficiently Modifying Path-Preserving Hierarchies for Browsing Paths. *TVCG*, 17(3), 2011.

[11] G. A. Atemezing and R. Troncy. Towards a linked-data based visualization wizard. In *COLD*, 2014.

[12] D. Auber. Tulip - A Huge Graph Visualization Framework. In *Graph Drawing Software*. 2004.

[13] S. Auer, R. Doehring, and S. Dietzold. LESS – Template-Based Syndication and Presentation of Linked Data. In *ESWC*, 2010.

[14] B. Bach, E. Pietriga, and I. Liccardi. Visualizing Populated Ontologies with OntoTrix. *IJSWIS*, 9(4), 2013.

[15] M. Bastian, S. Heymann, and M. Jacomy. Gephi: An Open Source Software for Exploring and Manipulating Networks. In *ICWSM*, 2009.

[16] L. Battle, R. Chang, and M. Stonebraker. Dynamic Prefetching of Data Tiles for Interactive Visualization, 2015. Technical Report.

[17] L. Battle, M. Stonebraker, and R. Chang. Dynamic reduction of query result sets for interactive visualizaton. In *BigData*, 2013.

[18] C. Becker and C. Bizer. Exploring the Geospatial Semantic Web with DBpedia Mobile. *J. Web Sem.*, 7(4), 2009.

[19] F. Benedetti, L. Po, and S. Bergamaschi. A Visual Summary for Linked Open Data sources. In *ISWC*, 2014.

[20] K. Bereta, C. Nikolaou, M. Karpathiotakis, K. Kyzirakos, and M. Koubarakis. SexTant: Visualizing Time-Evolving Linked Geospatial Data. In *ISWC*, 2013.

[21] T. Berners-Lee, Y. Chen, L. Chilton, D. Connolly, R. Dhanaraj, J. Hollenbach, A. Lerer, and D. Sheets. Tabulator: Exploring and Analyzing linked data on the Semantic Web. In *SWUI*, 2006.

[22] N. Bikakis, J. Liagouris, M. Krommyda, G. Papastefanatos, and T. Sellis. Towards Scalable Visual Exploration of Very Large RDF Graphs. In *ESWC*, 2015.

[23] N. Bikakis, J. Liagouris, M. Krommyda, G. Papastefanatos, and T. Sellis. graphVizdb: A Scalable Platform for Interactive Large Graph Visualization. In *ICDE*, 2016.

[24] N. Bikakis and G. Papastefanatos. Visual Exploration and Analytics of Big Data: Challenges and Approaches, 2016.

[25] N. Bikakis, G. Papastefanatos, M. Skourla, and T. Sellis. A Hierarchical Aggregation Framework for Efficient Multilevel Visual Exploration and Analysis, 2015. Techn. Rep., `http://arxiv.org/abs/1511.04750`.

[26] N. Bikakis, M. Skourla, and G. Papastefanatos. rdf:SynopsViz - A Framework for Hierarchical Linked Data Visual Exploration and Analysis. In *ESWC*, 2014.

[27] T. Boinski, A. Jaworska, R. Kleczkowski, and P. Kunowski. Ontology visualization. In *ITNGSWUI*, 2010.

[28] M. Bostock, V. Ogievetsky, and J. Heer. D^3 Data-Driven Documents. *TVCG*, 17(12), 2011.

[29] J. M. Brunetti, S. Auer, R. García, J. Klímek, and M. Necaský. Formal Linked Data Visualization Model. In *iiWAS*, 2013.

[30] J. M. Brunetti, R. Gil, and R. García. Facets and Pivoting for Flexible and Usable Linked Data Exploration. In *Interacting with Linked Data Workshop*, 2012.

[31] D. V. Camarda, S. Mazzini, and A. Antonuccio. LodLive, exploring the web of data. In *I-SEMANTICS*, 2012.

[32] A. E. Cano, A. Dadzie, and M. Hartmann. *Who's Who* - A Linked Data Visualisation Tool for Mobile Environments. In *ESWC*, 2011.

[33] S. Chan, L. Xiao, J. Gerth, and P. Hanrahan. Maintaining interactivity while exploring massive time series. In *IEEE VAST*, 2008.

[34] W. Cui, H. Zhou, H. Qu, P. C. Wong, and X. Li. Geometry-Based Edge Clustering for Graph Visualization. *TVCG*, 14(6), 2008.

[35] A. Dadzie and M. Rowe. Approaches to visualising Linked Data: A survey. *Semantic Web*, 2(2), 2011.

[36] L. Deligiannidis, K. Kochut, and A. P. Sheth. RDF data exploration and visualization. In *Workshop on CyberInfrastructure: Information Management in eScience*, 2007.

[37] K. Dimitriadou, O. Papaemmanouil, and Y. Diao. Explore-by-Example: An Automatic Query Steering Framework for Interactive Data Exploration. In *SIGMOD*, 2014.

[38] J. Dokulil and J. Katreniaková. Using Clusters in RDF Visualization. In *Advances in Semantic Processing*, 2009.

[39] P. R. Doshi, E. A. Rundensteiner, and M. O. Ward. Prefetching for Visual Data Exploration. In *DASFAA*, 2003.

[40] M. Dudás, O. Zamazal, and V. Svátek. Roadmapping and Navigating in the Ontology Visualization Landscape. In *EKAW*, 2014.

[41] A. Eldawy, M. Mokbel, and C. Jonathan. HadoopViz: A MapReduce Framework for Extensible Visualization of Big Spatial Data. In *ICDE*, 2016.

[42] N. Elmqvist and J. Fekete. Hierarchical Aggregation for Information Visualization: Overview, Techniques, and Design Guidelines. *TVCG*, 16(3), 2010.

[43] I. Ermilov, M. Martin, J. Lehmann, and S. Auer. Linked Open Data Statistics: Collection and Exploitation. In *Knowledge Engineering and the Semantic Web*, 2013.

[44] O. Ersoy, C. Hurter, F. V. Paulovich, G. Cantareiro, and A. Telea. Skeleton-Based Edge Bundling for Graph Visualization. *TVCG*, 17(12), 2011.

[45] S. Falconer, C. Callendar, and M.-A. Storey. A Visualization Service for the Semantic Web. In *Knowledge Engineering and Management by the Masses*. 2010.

[46] D. Fisher, I. O. Popov, S. M. Drucker, and M. C. Schraefel. Trust Me, I'm Partially Right: Incremental Visualization Lets Analysts Explore Large Datasets Faster. In *CHI*, 2012.

[47] B. Fu, N. F. Noy, and M.-A. Storey. Eye Tracking the User Experience - An Evaluation of Ontology Visualization Techniques. *Semantic Web Journal*, 2015.

[48] E. R. Gansner, Y. Hu, S. C. North, and C. E. Scheidegger. Multilevel Agglomerative Edge Bundling for Visualizing Large Graphs. In *PacificVis*, 2011.

[49] P. Godfrey, J. Gryz, and P. Lasek. Interactive Visualization of Large Data Sets, 2015. Technical Report.

[50] A. Graves. Creation of Visualizations Based on Linked Data. In *WIMS*, 2013.

[51] F. Haag, S. Lohmann, S. Negru, and T. Ertl. OntoViBe: An Ontology Visualization Benchmark. In *VISUAL*, 2014.

[52] P. Haase, M. Schmidt, and A. Schwarte. The Information Workbench as a Self-Service Platform for Linked Data Applications. In *COLD*, 2011.

[53] A. Harth. VisiNav: A system for visual search and navigation on web data. *J. Web Sem.*, 8(4), 2010.

[54] T. Hastrup, R. Cyganiak, and U. Bojars. Browsing Linked Data with Fenfire. In *WWW*, 2008.

[55] J. Heer and S. Kandel. Interactive Analysis of Big Data. *ACM Crossroads*, 19(1), 2012.

[56] J. Heer and B. Shneiderman. Interactive Dynamics for Visual Analysis. *Commun. ACM*, 55(4), 2012.

[57] P. Heim, T. Ertl, and J. Ziegler. Facet Graphs: Complex Semantic Querying Made Easy. In *ESWC*, 2010.

[58] P. Heim, S. Lohmann, and T. Stegemann. Interactive Relationship Discovery via the Semantic Web. In *ESWC*, 2010.

[59] P. Heim, S. Lohmann, D. Tsendragchaa, and T. Ertl. SemLens: visual analysis of semantic data with scatter plots and semantic lenses. In *I-SEMANTICS*, 2011.

[60] J. Helmich, J. Klímek, and M. Necaský. Visualizing RDF Data Cubes Using the Linked Data Visualization Model. In *ESWC*, 2014.

[61] N. Henry, J. Fekete, and M. J. McGuffin. NodeTrix: a Hybrid Visualization of Social Networks. *TVCG*, 13(6), 2007.

[62] M. Hildebrand, J. van Ossenbruggen, and L. Hardman. /facet: A Browser for Heterogeneous Semantic Web Repositories. In *ISWC*, 2006.

[63] D. Holten. Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data. *TVCG*, 12(5), 2006.

[64] W. Hop, S. de Ridder, F. Frasincar, and F. Hogenboom. Using Hierarchical Edge Bundles to visualize complex ontologies in GLOW. In *ACM SAC*, 2012.

[65] A. Hussain, K. Latif, A. Rextin, A. Hayat, and M. Alam. Scalable Visualization of Semantic Nets using Power-Law Graphs. *AMIS*, 8(1), 2014.

[66] D. Huynh, S. Mazzocchi, and D. R. Karger. Piggy Bank: Experience the Semantic Web Inside Your Web Browser. In *ISWC*, 2005.

[67] S. Idreos, M. L. Kersten, and S. Manegold. Database Cracking. In *CIDR*, 2007.

[68] S. Idreos, O. Papaemmanouil, and S. Chaudhuri. Overview of Data Exploration Techniques. In *SIGMOD*, 2015.

[69] J. Im, F. G. Villegas, and M. J. McGuffin. VisReduce: Fast and Responsive Incremental Information Visualization of Large Datasets. In *BigData*, 2013.

[70] P. Jayachandran, K. Tunga, N. Kamat, and A. Nandi. Combining User Interaction, Speculative Query Execution and Sampling in the DICE System. *PVLDB*, 7(13), 2014.

[71] J. F. R. Jr., H. Tong, J. Pan, A. J. M. Traina, C. T. Jr., and C. Faloutsos. Large Graph Analysis in the GMine System. *TKDE*, 25(1), 2013.

[72] J. F. R. Jr., H. Tong, A. J. M. Traina, C. Faloutsos, and J. Leskovec. GMine: A System for Scalable, Interactive Graph Visualization and Mining. In *VLDB*, 2006.

[73] U. Jugel, Z. Jerzak, G. Hackenbroich, and V. Markl. Faster Visual Analytics through Pixel-Perfect Aggregation. *PVLDB*, 7(13), 2014.

[74] U. Jugel, Z. Jerzak, G. Hackenbroich, and V. Markl. VDDA: automatic visualization-driven data aggregation in relational databases. *VLDBJ*, 2015.

[75] E. Kalampokis, A. Nikolov, P. Haase, R. Cyganiak, A. Stasiewicz, A. Karamanou, M. Zotou, D. Zeginis, E. Tambouris, and K. A. Tarabanis. Exploiting Linked Data Cubes with OpenCube Toolkit. In *ISWC*, 2014.

[76] A. Kalinin, U. Çetintemel, and S. B. Zdonik. Interactive Data Exploration Using Semantic Windows. In *SIGMOD*, 2014.

[77] A. Kalinin, U. Çetintemel, and S. B. Zdonik. Searchlight: Enabling Integrated Search and Exploration over Large Multidimensional Data. *PVLDB*, 8(10), 2015.

[78] N. Kamat, P. Jayachandran, K. Tunga, and A. Nandi. Distributed and Interactive Cube Exploration. In *ICDE*, 2014.

[79] B. Kämpgen and A. Harth. OLAP4LD - A Framework for Building Analysis Applications Over Governmental Statistics. In *ESWC*, 2014.

[80] A. Katifori, C. Halatsis, G. Lepouras, C. Vassilakis, and E. G. Giannopoulou. Ontology visualization methods - a survey. *ACM Comput. Surv.*, 39(4), 2007.

[81] D. Keim and H.-P. Kriegel. VisDB: Database Exploration using Multidimensional Visualization. *IEEE Computer Graphics and Applications*, 14(5), 1994.

[82] A. Key, B. Howe, D. Perry, and C. R. Aragon. VizDeck: Self-organizing Dashboards for Visual Analytics. In *SIGMOD*, 2012.

[83] H. A. Khan, M. A. Sharaf, and A. Albarrak. DivIDE: efficient diversification for interactive data exploration. In *SSDBM*, 2014.

[84] J. Klímek, J. Helmich, and M. Necaský. Payola: Collaborative Linked Data Analysis and Visualization Framework. In *ESWC*, 2013.

[85] J. Klímek, J. Helmich, and M. Necaský. Use Cases for Linked Data Visualization Model. In *LDOW*, 2015.

[86] G. Kobilarov and I. Dickinson. Humboldt: Exploring Linked Data. In *LDOW*, 2008.

[87] J. Koch and T. Franz. LENA - Browsing RDF Data More Complex Than Foaf. In *ISWC*, 2008.

[88] S. Kriglstein and R. Motschnig-Pitrik. Knoocks: New Visualization Approach for Ontologies. In *Conference on Information Visualisation*, 2008.

[89] S. Krivov, R. Williams, and F. Villa. GrOWL: A tool for visualization and editing of OWL ontologies. *J. Web Sem.*, 5(2), 2007.

[90] A. Lambert, R. Bourqui, and D. Auber. Winding Roads: Routing Edges into Bundles. *CGF*, 29(3), 2010.

[91] M. Lanzenberger, J. Sampson, and M. Rester. Visualization in Ontology Tools. In *CISIS*, 2009.

[92] A. d. Leon, F. Wisniewski, B. Villazón-Terrazas, and O. Corcho. Map4rdf- Faceted Browser for Geospatial Datasets. In *Using Open Data: policy modeling, citizen empowerment, data journalism*, 2012.

[93] C. Li, G. Baciu, and Y. Wang. ModulGraph: Modularity-based Visualization of Massive Graphs. In *Visualization in High Performance Computing*, 2015.

[94] T. Liebig and O. Noppens. OntoTrack: A semantic approach for ontology authoring. *J. Web Sem.*, 3(2-3), 2005.

[95] Z. Lin, N. Cao, H. Tong, F. Wang, U. Kang, and D. H. P. Chau. Demonstrating Interactive Multi-resolution Large Graph Exploration. In *ICDM Workshops*, 2013.

[96] L. D. Lins, J. T. Klosowski, and C. E. Scheidegger. Nanocubes for Real-Time Exploration of Spatiotemporal Datasets. *TVCG*, 19(12), 2013.

[97] Z. Liu, B. Jiang, and J. Heer. *imMens*: Real-time Visual Querying of Big Data. *CGF*, 32(3):421–430, 2013.

[98] M. Livny, R. Ramakrishnan, K. S. Beyer, G. Chen, D. Donjerkovic, S. Lawande, J. Myllymaki, and R. K. Wenger. DEVise: Integrated Querying and Visual Exploration of Large Datasets. In *SIGMOD*, 1997.

[99] S. Lohmann, S. Negru, F. Haag, and T. Ertl. VOWL 2: User-Oriented Visualization of Ontologies. In *EKAW*, 2014.

[100] S. Lohmann, S. Negru, F. Haag, and T. Ertl. Visualizing Ontologies with VOWL. *Semantic Web Journal*, 2015.

[101] N. Marie and F. L. Gandon. Survey of Linked Data Based Exploration Systems. In *IESD*, 2014.

[102] S. Mazumdar, D. Petrelli, K. Elbedweihy, V. Lanfranchi, and F. Ciravegna. Affective graphs: The visual appeal of Linked Data. *Semantic Web*, 6(3), 2015.

[103] K. Morton, M. Balazinska, D. Grossman, and J. D. Mackinlay. Support the Data Enthusiast: Challenges for Next-Generation Data-Analysis Systems. *PVLDB*, 7(6), 2014.

[104] E. Motta, P. Mulholland, S. Peroni, M. d'Aquin, J. M. Gómez-Pérez, V. Mendez, and F. Zablith. A Novel Approach to Visualizing and Navigating Ontologies. In *ISWC*, 2011.

[105] Y. Park, M. J. Cafarella, and B. Mozafari. Visualization-Aware Sampling for Very Large Databases. In *ICDE*, 2016.

[106] I. Petrou, M. Meimaris, and G. Papastefanatos. Towards a methodology for publishing Linked Open Statistical Data. *eJournal of eDemocracy & Open Government*, 6(1), 2014.

[107] D. Phan, L. Xiao, R. B. Yeh, P. Hanrahan, and T. Winograd. Flow Map Layout. In *InfoVis*, 2005.

[108] E. Pietriga. IsaViz: a Visual Environment for Browsing and Authoring RDF Models. In *WWW*, 2002.

[109] J. Polowinski and M. Voigt. VISO: A Shared, Formal Knowledge Base As a Foundation for Semi-automatic Infovis Systems. In *CHI*, 2013.

[110] I. O. Popov, M. M. C. Schraefel, W. Hall, and N. Shadbolt. Connecting the Dots: A Multi-pivot Approach to Data Exploration. In *ISWC*, 2011.

[111] D. A. Quan and R. Karger. How to make a semantic web browser. In *WWW*, 2004.

[112] P. Ristoski and H. Paulheim. Visual Analysis of Statistical Data on Maps using Linked Open Data. In *ESWC*, 2015.

[113] L. Rutledge, J. van Ossenbruggen, and L. Hardman. Making RDF presentable: integrated global and local semantic Web browsing. In *WWW*, 2005.

[114] P. E. R. Salas, F. M. D. Mota, K. K. Breitman, M. A. Casanova, M. Martin, and S. Auer. Publishing Statistical Data on the Web. *Int. J. Semantic Computing*, 6(4), 2012.

[115] C. Sayers. Node-centric RDF Graph Visualization, 2004. Technical Report HP Laboratories.

[116] K. Schlegel, F. Stegmaier, S. Bayerl, M. Granitzer, and H. Kosch. Balloon Fusion: SPARQL Rewriting based on Unified Co-Reference Information. In *DESWeb*, 2014.

[117] K. Schlegel, T. Weißgerber, F. Stegmaier, C. Seifert, M. Granitzer, and H. Kosch. Balloon Synopsis: A Modern Node-Centric RDF Viewer and Browser for the Web. In *ESWC*, 2014.

[118] B. Shneiderman. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *IEEE Symposium on Visual Languages*, 1996.

[119] B. Shneiderman. Extreme Visualization: Squeezing a Billion Records into a Million Pixels. In *SIGMOD*, 2008.

[120] M. G. Skjæveland. Sgvizler: A JavaScript Wrapper for Easy Visualization of SPARQL Result Sets. In *ESWC*, 2012.

[121] C. Stadler, J. Lehmann, K. Höffner, and S. Auer. LinkedGeoData: A core for a web of spatial open data. *Semantic Web*, 3(4), 2012.

[122] C. Stadler, M. Martin, and S. Auer. Exploring the web of spatial data with facete. In *WWW*, 2014.

[123] C. D. Stolper, A. Perer, and D. Gotz. Progressive Visual Analytics: User-Driven Visual Exploration of In-Progress Analytics. *TVCG*, 20(12), 2014.

[124] C. Stolte and P. Hanrahan. Polaris: A System for Query, Analysis and Visualization of Multi-Dimensional Relational Databases. In *InfoVis*, 2000.

[125] M. D. Storey, N. F. Noy, M. A. Musen, C. Best, R. W. Fergerson, and N. A. Ernst. Jambalaya: an interactive environment for exploring ontologies. In *IUI*, 2002.

[126] M. Stuhr, D. Roman, and D. Norheim. LODWheel - JavaScript-based Visualization of RDF Data. In *COLD*, 2011.

[127] S. Sundara, M. Atre, V. Kolovski, S. Das, Z. Wu, E. I. Chong, and J. Srinivasan. Visualizing large-scale RDF data using Subsets, Summaries, and Sampling in Oracle. In *ICDE*, 2010.

[128] F. Tauheed, T. Heinis, F. Schürmann, H. Markram, and A. Ailamaki. SCOUT: Prefetching for Latent Feature Following Queries. *PVLDB*, 5(11), 2012.

[129] K. Thellmann, M. Galkin, F. Orlandi, and S. Auer. LinkDaViz - Automatic Binding of Linked Data to Visualizations. In *ISWC*, 2015.

[130] C. Tominski, J. Abello, and H. Schumann. CGV - An Interactive Graph Visualization System. *Computers & Graphics*, 33(6), 2009.

[131] G. Tschinkel, E. E. Veas, B. Mutlu, and V. Sabol. Using Semantics for Interactive Visual Analysis of Linked Open Data. In *ISWC*, 2014.

[132] F. Valsecchi, M. Abrate, C. Bacciu, M. Tesconi, and A. Marchetti. DBpedia Atlas: Mapping the Uncharted Lands of Linked Data. In *LDOW*, 2015.

[133] F. Valsecchi and M. Ronchetti. Spacetime: a Two Dimensions Search and Visualisation Engine Based on LinkedData. In *SEMAPRO*, 2014.

[134] M. Vartak, S. Madden, A. G. Parameswaran, and N. Polyzotis. SEEDB: Automatically Generating Query Visualizations. *PVLDB*, 7(13), 2014.

[135] M. Voigt, S. Pietschmann, L. Grammel, and K. Meißner. Context-aware Recommendation of Visualization Components. In *eKNOW*, 2012.

[136] M. Voigt, S. Pietschmann, and K. Meißner. A Semantics-Based, End-User-Centered Information Visualization Process for Semantic Web Data. In *Semantic Models for Adaptive Interactive Systems*. 2013.

[137] T. D. Wang and B. Parsia. CropCircles: Topology Sensitive Visualization of OWL Class Hierarchies. In *ISWC*, 2006.

[138] H. Wickham. Bin-Summarise-Smooth: A Framework for Visualising Large Data. Technical report, 2013.

[139] K. Wongsuphasawat, D. Moritz, A. Anand, J. D. Mackinlay, B. Howe, and J. Heer. Voyager: Exploratory Analysis via Faceted Browsing of Visualization Recommendations. *TVCG*, 22(1), 2016.

[140] E. Wu, L. Battle, and S. R. Madden. The Case for Data Visualization Management Systems. *PVLDB*, 7(10), 2014.

[141] E. Wu and S. Madden. Scorpion: Explaining Away Outliers in Aggregate Queries. *PVLDB*, 6(8), 2013.

[142] K. Zhang, H. Wang, D. T. Tran, and Y. Yu. ZoomRDF: semantic fisheye zooming on RDF data. In *WWW*, 2010.

[143] M. Zinsmaier, U. Brandes, O. Deussen, and H. Strobelt. Interactive Level-of-Detail Rendering of Large Graphs. *TVCG*, 18(12), 2012.

[144] K. Zoumpatianos, S. Idreos, and T. Palpanas. Indexing for interactive exploration of big data series. In *SIGMOD*, 2014.