

Visualizing the Blogosphere with BlogConnect

Justus Broß*, Patrick Schilf†, Maximilian Jenders‡ and Christoph Meinel§

Hasso Plattner Institute at the University of Potsdam
Prof.-Dr.-Helmert-Strasse 2-3, 14482 Potsdam, Germany

Fax: +49 (0)331 5509-325

*Telephone: +49 (0)331 5509-524, Email: justus.bross@hpi.uni-potsdam.de

†Telephone: +49 (0)331 5509-535, Email: patrick.schilf@student.hpi.uni-potsdam.de

‡Telephone: +49 (0)331 5509-222, Email: maximilian.jenders@student.hpi.uni-potsdam.de

§Telephone: +49 (0)331 5509-222, Email: office-meinel@hpi.uni-potsdam.de

Abstract—It can be highly meaningful for individuals, institutions and even governments to extract reliable and insightful trends, opinions or particular pieces of information out of the network single weblogs are embedded in - the blogosphere. However, there has been virtually little in the way of user interfaces designed for the exploration and information gathering from the blogosphere as a whole to allow for an integrated knowledge collection and information extraction tool. Users have to rely on their own capability to find, select or filter entries and navigate through this complex network, a task that can easily become tedious. An approach to solve this problem could be BLOGCONNECT presented in the following, a tool to visualize the inherent network characteristics of and the aggregated knowledge pool of the blogosphere.

I. INTRODUCTION

Since the end of the 90's weblogs - commonly known as "blogs" - have evolved to become an essential component of today's cyber culture [1]. While Sullivan (see [2]) refers to weblogs as "a publishing revolution more profound than anything since the printing press", other communication scientists circumscribe weblogs more carefully as "a new structural change of the public" [3] [4]. Widely accepted blog portals such as BlogPulse, for instance, currently clock more 150 million weblogs worldwide¹. Compared to around 60 million blogs in the year 2006, this constitutes their increasing importance in today's Internet society on a global scale [5] [6].

Single weblogs are embedded into an exceptionally interlinked network, known as the "blogosphere". The overall complexity of the blogosphere is, among others, due to the diverse topics covered and the different fields of interest blogs are employed in. Additionally, their global interconnectedness and the corresponding aggregation of individual knowledge create a gigantic and constantly growing archive of open source intelligence [7]. Leveraging this knowledge by extracting reliable and insightful trends, opinions or particular pieces of information from the blogosphere can be highly meaningful for a multitude of individuals, institutions and even governments: through their direct, informal and unadorned mode of operation, weblogs can serve as a fast provider of insight information about technical product innovations, politics or coverage on a multitude of other topics (see e.g.

[8] [9] [18]). However, it proves to be increasingly difficult for the average Internet user and sympathizer of blogs to grasp the blogosphere's complexity as a whole, yet due to the fact that thousands of new blogs and an almost uncountable number of new posts add up to the before-mentioned collective on a daily basis.

Thus, visualizing insightful information out of this knowledge pool in order to leverage content- and context-related structures residing in the blogosphere forms the ultimate objective of the tool presented in this paper. To provide such a contemporary tool for the improved exploration of the blogosphere, we present BLOGCONNECT: An interactive and well-arranged visualization tool to powerfully explore and browse through the network of blogs. Its user interface was specifically designed to facilitate the operations needed to explore the blogosphere in the most easy, efficient, and enjoyable way possible.

The subsequent section II integrates our work into existent research efforts. Section III is solely dedicated to the tool's visualization by clarifying the functionality of the graphical user interface. The implementation details of our visualization tool are elaborated on in section IV. Based on the insights of the before-mentioned sections, we provide recommendations for further research in section V followed by a general conclusion (section VI) and a list of references.

II. RELATED WORK

There is a variety of research efforts currently in progress that intent to visualize dependencies in between single weblogs within the networks of (partial) blogospheres.

An interesting topic in one blog spreads throughout the blogosphere by being picked up by other blog authors, similar to the dispersion of epidemic diseases, thus the relating visualization technique became known as "infection trees" [10] [11] [12] (also see [13]).

Herring and his colleagues empirically investigated the extent to which (and in what patterns) blogs are interconnected [14]. Their visualization of link patterns in conjunction with quantitative social network analysis as well as qualitative analysis of references and comments found out that the most influential blogs, commonly known as "A-list" blogs, are

¹"BlogPulse Stats" on 11 of April 2011: <http://www.blogpulse.com/>

overrepresented in the network although other groupings of blogs are more densely interconnected.

Chin and Chignel [15] examine how communities can be discovered through interconnected blogs as a form of social hypertext. Their proposed model detects communities in blogs by aligning centrality measures from social network analysis with parameters of the "Sense of Community" model (SOC) of Mcmillan and Chavis [16]. A similar approach of Efimov et al. was noted in the work "Finding 'the life' between the buildings [...]" [17].

"Vizblog" is another visualization technique to reveal similarities between blog entries in different blogs [19]. Through association and content analysis, blog entries are linked to each other to form clusters of related content. By manipulating the graph and filtering content, their visualization let users navigate and explore online discussions. It can furthermore promote participation by highlighting the "buzz" of popular topics and laying out the structure of conversations (see also [20] [21]).

Tirapat et al. have a similar approach in revealing "buzz" in the blogosphere [22]. Their method uses information-retrieval techniques to associate blog entries with topics, collected by crawling an authoritative resource on this subject area. They visualize and systematically analyze several blogs in the same domain of interest (movies) in order to assess to what extent the "buzz" of blogs correlates with public opinion. The collected information is represented in terms of a topic map, which is subsequently visualized in three different types of views, each one designed to communicate a different aspect of the data.

III. VISUALIZATION

The amount of blogs that build up the blogosphere each define their own categories. Blogs share many common categories. They are, however, lacking uniform naming conventions. Nevertheless, it is possible to cluster blogs according to topics solely based on their categories and tags. Mimicking this, the visualization itself is based on a cluster scheme. The following two subsections explain the different meanings of all parts of the visualization and the user interface of BLOGCONNECT.

A. Defining the cluster

The BLOGCONNECT visualization comes as a cluster consisting of solid circles, referred to as blog-"bubbles" in the following, and textual labels. Visualizing a blog, the small and opaque blog-bubble's color depends on its topic (i.e. the one that most blog's posts are attributed with) whilst its size is influenced by its calculated rank, which is described in [24]. The exact location in the cluster - amongst other parameters - is influenced by the connectivity to other blogs, that is incoming and outgoing links. In order to uniquely distinguish blog-Bubbles of the same topic, they are each given a black stroke. Also note that BLOGCONNECT only uses a set of uniquely distinguishable colors which is naturally limited in its size.

The textual labels visualize Trends, which are buzzwords currently being discussed in (parts of) the blogosphere. The determination of trends, their relating blogs and position in the cluster is part of [24].

The cluster contains additional, initially hidden information that can be conveyed by user interaction. Amongst those is the visualization of a blog's connectivity to other ones which is displayed by edges between the Bubbles. Other aspects like interdependencies among posts of a single blog are visualized by external tools, e.g. POSTCONNECT [25], and are accessible through the menu.

B. Exploring the cluster

Figure 2 displays the initial view of the graphical user interface of BLOGCONNECT containing the visualization cluster as previously described in this section. The centralized cluster (figure 2.A.I) is surrounded by other navigational aides which are the Sidebar (figure 2.A.II), the zoom slider (figure 2.A.III) as well as the footer (figure 2.A.IV), which contains all topic names.

The visualization cluster bundles the majority of interaction opportunities as it allows the blogosphere with all relationships among blogs, topics and trends, their interdependencies as well as structural characteristics to be easily and meaningfully explored. Note that in the initial view, only a limited dataset of blogs is rendered. Due to the enormous amount of recorded blogs by our crawler, only the high ranked ones are instantly visualized. The others, in contrast, become active when zooming into the cluster (see subsection IV-D for further details). Inactive blogs are not entirely hidden but stay slightly visible to keep the all contextual information as well as the possibility of switching to another visualization element without the need of resetting the view. When hovering over an active element, the user is provided with brief information related to that element, such as a blog or trend title, via a small tooltip window as displayed in figure 2.C. Clicking on an element leads to a view similar to the one in figure 2.B, exemplary displaying the specific case of a focused blog. Every blog that is not a member of the set of blogs that are referenced by the focused one becomes inactive whilst all remaining stay highlighted.

At the right border of the screen appears the sidebar, consisting of vertical tabs, each of for a different purpose. The "Blogs" and "Hot Topics" tabs list all corresponding visualized elements in textual form to provide alternative means of navigation when looking for a specific blog or trend. The other tabs deal with setting up user-preferences. It grants access to the filter sub-menu, where the user can hide content that might (temporarily) not be of greater interest by maintaining filter rules for any data layer. Subsection IV-D explains the filtering mechanism in detail. Also, a search function as a quick access method - to look up a very specific single blog for example - may be selected from one tab.

A vertical slider at the left border of the screen allows the user to zoom in deeper into the cluster. This enables the user to also explore smaller blogs of lower ranks that might be

hidden at the initial view for reasons already mentioned in the preface of this section. The slider zooms in and out to and from the center of the screen. Even when zoomed in, any sub-cluster can still be reached using drag functionality as known from interactive geographical maps. An *InteractionDomain* ensures that no other action can be executed while dragging or zooming. Subsection IV-E describes this handling mechanism in detail. Zooming can also be applied using a mouse wheel or touch-pad. In the same corner as the zoomer little icons appear to jump to a blog url or access "Sub-visualization" tools when available. POSTCONNECT is one one of these independently implemented tools in the context of the BLOGINTELLIGENCE project to visualize interdependencies of posts within a single blog [25]. More tools to visualize the main information services of BLOGINTELLIGENCE are currently in the planning phase.

IV. IMPLEMENTATION

BLOGCONNECT was implemented as an Adobe Flex² application built upon AXIIS³, which is an open source Flex library for simplifying basic visualization tasks such as mapping and scaling data to graphical representations and grouping them within layouts. The visualization is rendered with the support of DEGRAFA⁴, another open source graphic project, which comes with Axiis. Therefore, all elements that are part of the cluster inherit from Degrafa objects.

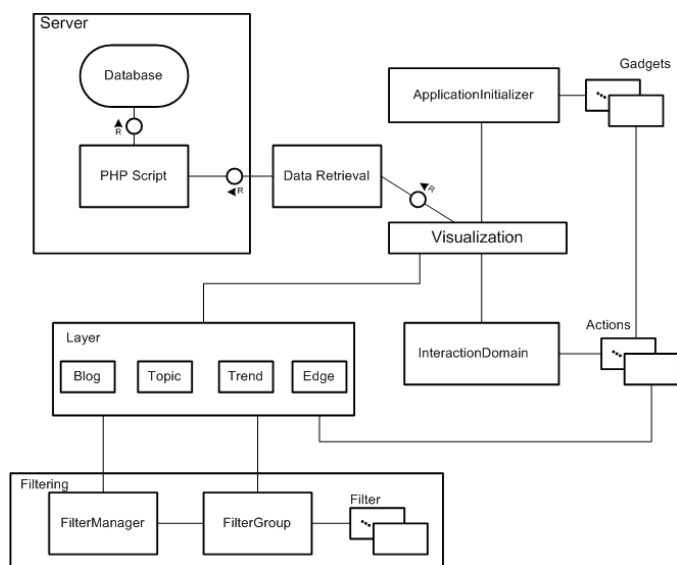


Fig. 1. Architecture

Figure 1 gives a basic overview of the major architecture components. *ApplicationInitializer* is the executable Flex MXML component that launches the application and first initializes the *Visualization* component, which is the central part of BLOGCONNECT's implementation. The *Visualization*

then in turn builds up the cluster scheme in a way as described in subsection III-A. The *Data Retrieval*, *Filter* and *Interaction-Domain* components are supporting packages dealing with the tasks of accessing a database, data filtering and user interaction respectively. Additionally, *Visualization* acts as a controller, directing communication between the other components. Also, descriptor objects for all visualization elements, acting as an abstraction layer, facilitate achieving a separation of visual representation and logic - in both directions, i.e. either looking up graphic to corresponding data (e.g. a blog filter that looks up a single blogs' sprite based on its title when performing a search operation) or the other way around (e.g. a topic descriptor finds all blogs contained in a clicked on a topic's sprite).

A. ApplicationInitializer

The *ApplicationInitializer* component is the application launcher of BLOGCONNECT. It initializes the *Visualization* (see subsection IV-B) and is responsible for displaying any screen overlays such as the menu bar, a zoom slider, the footer and a sidebar containing Gadgets that help the user navigating and exploring BLOGCONNECT. Section III-B further elaborates on the various ways of how the user can interact with each individual overlay.

B. Visualization

Visualization is the central component in BLOGCONNECT. It exposes the visualization cluster whilst hiding the entire back-end functionality underneath it, such as the actual visualization processing as well as connecting the other components of BLOGCONNECT. These are *InteractionDomain*, *Filtering* and *Data Retrieval*. As mentioned in the preface of this section, the actual visualization processing is done by Axiis. *Visualization* therefore inherits from the Axiis DataCanvas class. A DataCanvas holds layouts that each contain graphical elements, called Sprites in Flex, that are instantiated from input data being mapped upon them. In BLOGCONNECT, there are three different layouts, referenced to as *Layer* - one for each of the visualized metrics that are described in subsection III-A. These, hierarchically ordered, layers are implemented by the LayeredLayout class, which is an extension of Axiis' BasicLayout. Thus, there is one layer each for blogs and trends in addition to another one for edges, the latter one illustrating the linking structure between blogs.

When created, every layer retrieves data from a database through the PHP web service described in subsection IV-C. Although Axiis allows to perform some aggregation operation after the data have been retrieved from the database, this step is already processed server-sided by SQL in order to keep the initial loading time low. Thus, to finalize the layer creation, a *FilterGroup* is set up with its referenced dataset acting as a data provider for the layer. In consequence, the execution of a filter in that *FilterGroup* will affect the layer's dataset.

C. PHP Webservice

The input data needed for the visualization are mined by a RSS-feed crawler framework set up exclusively for the

²<http://www.adobe.com/products/flex/>

³<http://www.axiis.org/>

⁴<http://www.degrafa.org/>

BLOGINTELLIGENCE portal (see also [26]) and stored in a MySQL⁵ database. The same crawler also gathers data for the POSTCONNECT application [25]. As ActionScript code is executed client-sided and there is no Adobe flash player interface to directly access a remote database via SQL, BLOGCONNECT utilizes a web service as a server-sided solution. PHP is our language of choice, as libraries for database connection and web service creation were readily available. For database connection, the standard MySQL functions provided by basic PHP are sufficient. The web service is based on NuSOAP⁶, a rewrite of SOAPx4 allowing developers to create and consume web services based on SOAP 1.1, WSDL 1.1 and HTTP 1.0/1.1.

The PHP component provides several canned SQL queries that can be invoked via the web service. These queries do not only retrieve the data from the database, but additionally pre-aggregate it on the server using standard SQL functions. Flex-sided functions invoke those web service calls and then asynchronously wait for the result to be returned so it can be visualized.

D. Filtering

The *Filtering* component filters the amount of data to be visualized. As such it cares about two different use cases: Firstly automated filtering to shrink the the load of data and thus enable a fluently working application, even on older systems. Secondly, the user can set up filters to hide partial content that might (temporarily) not be of interest to him. The interface to the *Visualization* component is the *FilterManager* which maintains a list of *FilterGroups*. A *FilterGroup* deals with filtering one certain data layout and maintains a chain of several rules (*Filter*) to be applied to that layout. A filter rule is very similar to those known from a spam filter in an e-mail client and might look as follows: *Filter blogs where #LinkedBlogs > 5*. Applying this rule will leave only those blog Bubbles with more than five connected blogs to render. On creation, some filters might affect *FilterGroups* in charge of other layouts. When filtering topics, for instance, corresponding blogs and trends should also be filtered.

Filters are attributed to an owner which is either the user or the application itself. Application filters are added automatically, mainly to ensure performant rendering when zooming in and out of the visualization cluster. The maintenance of interdependencies among groups as well as their execution is done by the *FilterManager*, which is the interface to the entire *Filter* component. Once the filtering of a *FilterGroup* is initiated, it will sequentially run all its filters, whereas each resulting data will be used as input for the next rule in the chain. Filters cache their results until they get marked as 'dirty', which occurs whenever the chain order changes. That is, for instance, due to addition or removal of other filters. On recurring execution, only booked filters need to be executed again. This way the number of filter operations, and thus the total group execution time, stays low.

⁵<http://www.mysql.com/>

⁶<http://sourceforge.net/projects/nussoap/>

As application filters are supposed to keep a certain number (dependent on the system capability) of elements on the screen, they are placed at the end of each group's chain. The other way around user filters might filter out even more objects than necessary which would in turn not utilize the system as much as it could. Last but not least we distinguish two different kinds of filters: *TemporaryFilter* and *PermanentFilter*. *PermanentFilters* are the regular ones just described whereas *TemporaryFilters* are not maintained in the filter chain but immediately applied once they are created. In cases such as when the user performs a single search for a blog title, the result set apparently is not of any use as input for further filter operations.

E. InteractionDomain

The *InteractionDomain* controls the message flow between graphical objects and underlying back-end components. Whenever a user interface receives user input, an interaction request is created and sent over to the *InteractionDomain*, which is in charge of deciding whether to process the request or not. It maintains different interaction modes which affect this decision. For instance, the domain switches over to "drag" mode when objects, mainly the visualization cluster itself, are being dragged. Technically, this mode then prohibits all other action requests at this time until the end of the drag operation. Consult subsection III-B to find out more about further use cases for interaction with the user interface.

All UI interaction is wrapped in little container, in the following referred to as "actions". Each single action performs a well-defined task such as highlighting all blog bubbles that are member of the focused one's linking environment. Also, actions may be annotated with animation effects inherited from Flex' Tweens to be played upon execution of the action. To accommodate for users working on slower computers, each effect can be toggled on or off. In case an effect is disabled, every action defines alternative behavior that is executed instead.

Actions may be recursively composed from other actions. Each can - completely or partially, depending on the specific implementation - be undone. The *InteractionDomain* keeps track of the last performed action to possibly revert it when requested to either execute an action or reset the user interface. As a consequence, there is no need for tracking the whole interaction history since UI manipulations do not cause permanent changes to the application but instead can be undone anytime.

Decoupling actions allows for different graphical representations to call the same action to be called on the object they are representing and therefore affect the application in the same manner.

V. FUTURE RESEARCH

BLOGCONNECT is part of the overall BLOGINTELLIGENCE portal and even though we perceive the actual visualization solution BLOGCONNECT as already well-engineered, it still needs some performance optimizations to be highly scalable.

It will be the long-term research goal of BLOGINTELLIGENCE to accomplish real-time provision of its services and to expand its indexing coverage from the German (partial) blogosphere to the European one in the mid-term over to the American blogosphere in the long-term – dependent on possible external fundings or investments.

VI. CONCLUSION

We discussed the rationale, related research efforts and implementation details of our interactive visualization tool for decisively facilitating the exploration of the blogosphere for weblog enthusiasts. It is an interactive and well-arranged visualization tool to powerfully explore and browse through the superstructure of interconnected weblogs. We believe that BLOGCONNECT offers a superior and integrated approach of facilitating the exploration of the blogosphere compared to other research efforts or commercial platform that are currently in existence (see II).

REFERENCES

- [1] S. C. Herring, L. A. Scheidt, S. Bonus, and E. Wright, "Bridging the gap: A genre analysis of weblogs," in *Proceedings of the 37th Hawaii International Conference on System Sciences (HICSS'04)*. Big Island, Hawaii: IEEE, 2008. [Online]. Available: <http://doi.ieeecomputersociety.org/10.1109/hicss.2004.1265271>
- [2] B. A. Sullivan, "The Blogging Revolution - Weblogs Are To Words What Napster Was To Music," 2002.
- [3] C. Von Neuberger, C. Nuernbergk, and M. Rischke, "Weblogs und Journalismus: Konkurrenz, Ergänzung oder Integration? Eine Forschungssynopse zum Wandel der Öffentlichkeit im Internet," pp. 96–112, 2007. [Online]. Available: http://www.media-perspektiven.de/uploads/tx_mmpublications/02-2007_Neuberger.pdf
- [4] H.-J. Bucher and S. Büffel, *Weblogs - Journalismus in der Weltgesellschaft. Grundstrukturen einer netzwerkorientierten Form der Medienkommunikation*, Heidelberg, 2006, pp. 131–156.
- [5] T. Smith, "Power to the People: Social Media Tracker Wave 3," p. 80, 2008. [Online]. Available: http://www.goviral.com/articles/wave_3_20080403093750.pdf
- [6] Technorati.com, "State of the Blogosphere 2009," 2009. [Online]. Available: <http://technorati.com/blogging/feature/state-of-the-blogosphere-2009/>
- [7] J. Schmidt, *Weblogs: eine kommunikationssoziologische Studie*, 1st ed. Uvk, 2006. [Online]. Available: <http://www.citeulike.org/group/560/article/482986>
- [8] J. Broß, "Weblogs, a Promising New Form for E-democracy?" in *2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*. Sydney, Australia: IEEE, Dec. 2008, pp. 667–671. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4740867>
- [9] J. Broß, M. Quasthoff, S. M. Niven, and C. Meinel, "Implementing a corporate weblog for SAP," in *BlogTalk 2009 6th International Conference on Social Software*, J. Breslin, Ed., no. June 2009. Jetsu, South Korea: Springer LNCS 6045, Heidelberg, 2010, pp. 15–28.
- [10] E. Adar and L. A. Adamic, "Tracking Information Epidemics in Blogspace," *Web Intelligence*, 2005. [Online]. Available: <http://portal.acm.org/citation.cfm?id=1092473>
- [11] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins, "Information diffusion through blogspace," in *Proceedings of the 13th international conference on World Wide Web*, no. May 2004. New York, USA: ACM New York, NY, USA, 2004, pp. 491–501. [Online]. Available: <http://portal.acm.org/citation.cfm?id=988739>
- [12] A. Aschenbrenner and S. Miksch, "Blog Mining in a Corporate Environment," Wien, pp. 1–19, 2005. [Online]. Available: http://www.citeulike.org/user/domakesaythink/article/2382823?citation_format=IEEEtran/#
- [13] A. Anjewierden, R. de Hoog, R. Brussee, and L. Efimova, "Detecting knowledge flows in weblogs, 13th," in *Common Semantics for Sharing Knowledge: Contributions to ICCS 2005 13th International Conference on Conceptual Structures*, F. Dau, M.-L. Mugnier, and G. Stumme, Eds. Kassel: Kassel University Press, 2005, pp. 1–12.
- [14] S. Herring, I. Kouper, J. Paolillo, L. Scheidt, M. Tyworth, P. Welsch, E. Wright, and N. Yu, "Conversations in the blogosphere: An analysis" from the bottom up," in *Proceedings of the 38th Hawaii International Conference on System Sciences (HICSS'05)*, vol. 00, no. C. Ieee, 2005, pp. 107b–107b. [Online]. Available: <http://doi.ieeecomputersociety.org/10.1109/HICSS.2005.167>
- [15] A. Chin and M. Chignell, "A social hypertext model for finding community in blogs," in *Conference on Hypertext and Hypermedia (HT'06)*, Odense, Denmark, 2006, pp. 11–22. [Online]. Available: <http://portal.acm.org/citation.cfm?id=1149945>
- [16] D. Mcmillan and D. Chavis, "Sense of community: A definition and theory," *Journal of Community Psychology*, vol. 14, no. 1, pp. 6 – 23, 1986. [Online]. Available: <http://www.citeulike.org/user/GadgetMan/article/354237>
- [17] L. Efimova, S. Hendrick, and A. Anjewierden, "Finding 'the life between buildings': An approach for defining a weblog community," *Internet Research*, vol. 6, no. 1997, pp. 1–15, 2005. [Online]. Available: <http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Finding+the+life+between+buildings:+An+approach+for+defining+a+weblog+community\#0>
- [18] M. Chau and J. Xu, "Studying Customer Groups from Blogs," in *6th WeB 2007. WEB2007*, 2007, pp. 850–856. [Online]. Available: http://www.fbe.hku.hk/~{m}chau/papers/CustomerGroupsFromBlogs_WEB.pdf
- [19] C. Tauro, M. A. Pérez-Quiñones, P. Isenhour, S. Ahuja, and A. Kavanaugh, "VizBlog: Discovering Conversations in the Blogosphere," in *Technology demonstration at Directions and Implications of Advanced Computing-Conference on Online Deliberation*, Berkeley, CA, USA, 2008, p. 6. [Online]. Available: <http://diggov.cs.vt.edu/files/VizBlogDemonstrationPaper.pdf>
- [20] A. Qamra, B. Tseng, and E. Chang, "Mining blog stories using community-based and temporal clustering," in *Proceedings of the 15th ACM international conference on Information and knowledge management*. Arlington, Virginia, USA: ACM Press, 2006, p. 67. [Online]. Available: <http://portal.acm.org/citation.cfm?id=1183614.1183627>
- [21] F. Viégas and J. Donath, "Social network visualization: Can we go beyond the graph," in *Workshop on Social Networks, CSCW*, vol. 4. Citeseer, 2004, pp. 6–10. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.135.4963&rep=rep1&type=pdf>
- [22] T. Tirapat, C. Espiritu, and E. Stroulia, "Taking the Communities Pulse, one Blog at a Time," in *ICWE'06*. Palo Alto, California, USA: ACM Press, 2006, pp. 169–176. [Online]. Available: <http://www.cs.ualberta.ca/~{e}stroulia/EduNuggets/icwe2006-submitted.pdf>
- [23] M. L. Gregory, D. Payne, D. McColgin, N. Cramer, and D. Love, "Visual Analysis of Weblog Content," in *ICWSM'2007*. Boulder, Colorado, USA: PNNL-SA-54063, International Conference on Weblogs and Social Media'07, Boulder, CO, United States (US), 2007. [Online]. Available: <http://www.icws-m.org/papers/5--McColgin-Cramer-Gregory-Payne-Love.pdf>
- [24] J. Broß, K. Richly, P. Schilf, and C. Meinel, *Social Physics of the Blogosphere: Capturing, Analyzing and Presenting Interdependencies of Partial Blogospheres*. New York / Wien: Springer Verlag, 2010, pp. 179–198.
- [25] J. Broß, P. Schilf, and C. Meinel, "Visualizing blog archives to explore content-and context-related interdependencies," in *The IEEE/ACM WI-IAT 2010 Conference, IEEE/ACM WI-IAT 2010*, Ed. Toronto: IEEE Computer Society, 2010, pp. 647–652. [Online]. Available: <http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Visualizing+blog+archives+to+explore+content-+and+context-related+interdependencies\#0>
- [26] J. Broß, M. Quasthoff, P. Berger, P. Hennig, and C. Meinel, "Mapping the Blogosphere with RSS-Feeds," in *2010 24th IEEE International Conference on Advanced Information Networking and Applications*. Perth, Australia: IEEE, 2010, pp. 453–460.

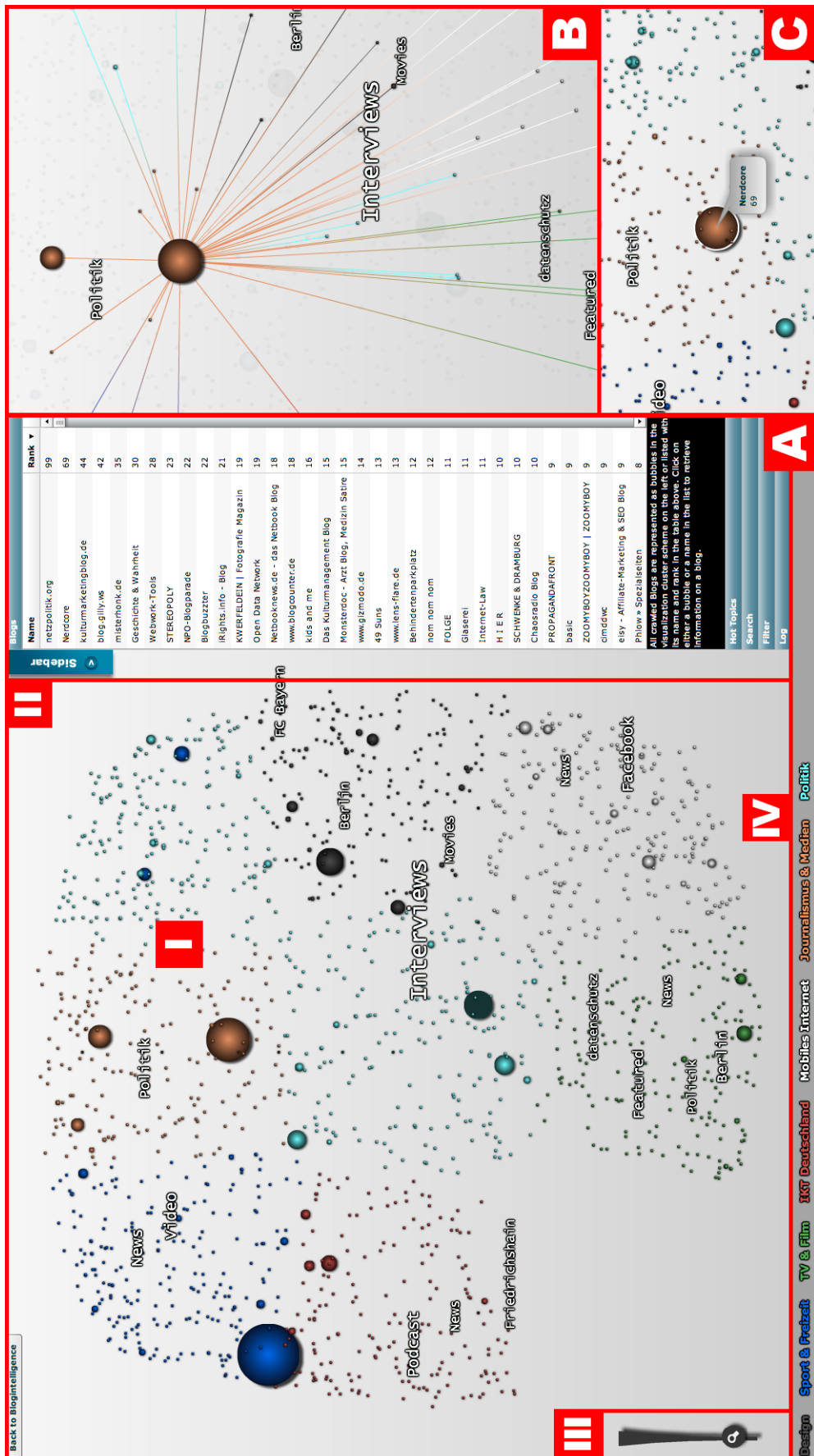


Fig. 2. Main User Interface of BlogConnect