Have Green – A Visual Analytics Framework for Large Semantic Graphs

Pak Chung Wong, George Chin Jr., Harlan Foote, Patrick Mackey, Jim Thomas

Pacific Northwest National Laboratory

ABSTRACT

A semantic graph is a network of heterogeneous nodes and links annotated with a domain ontology. In intelligence analysis, investigators use semantic graphs to organize concepts and relationships as graph nodes and links in hopes of discovering key trends, patterns, and insights. However, as new information continues to arrive from a multitude of sources, the size and complexity of the semantic graphs will soon overwhelm an investigator's cognitive capacity to carry out significant analyses. We introduce a powerful visual analytics framework designed to enhance investigators' natural analytical capabilities to comprehend and analyze large semantic graphs. The paper describes the overall framework design, presents major development accomplishments to date, and discusses future directions of a new visual analytics system known as Have Green.

Keywords: Visual Analytics, Graph and Network Visualization, Information Analytics, Information Visualization

Index Terms: I.6.9 [Visualization] – Information Visualization, Visualization Systems and Software, Visualization Techniques and Methodologies

1 INTRODUCTION

A semantic graph is a network of heterogeneous nodes and links annotated with a domain ontology. The ontology of a semantic graph is a description or specification of the concepts and relationships that exist within the semantic graph [18]. In intelligence analysis, semantic graphs are generated and applied in a visual analysis approach known as *link analysis* [27]. Through link analysis, investigators draw, lay out, and link people, facts, locations, events, objects, and data in hopes of discovering key trends, patterns, and insights.

Link analysis has been applied in a number of high-profile cases recently, including the search for the District of Columbia snipers [26] and the search for Saddam Hussein during the U.S. invasion of Iraq [21]. Many of the link analysis graphs we encounter have properties of small world graphs [22], [33], [34], [35], which generally have high degrees of clustering and small average path lengths relative to their number of nodes. Small world graphs are commonly associated with social networks, neural networks, power grids, and internet traffic.

In today's intelligence environment, however, investigators are bombarded by massive amounts of information from a multitude of sources. The vast amounts of information being fed into semantic graphs may easily overwhelm an investigator's cognitive capacity. The diversity of this information—which usually contains formatted and unformatted text, image, video and audio recordings, and various other databases—also demands new technology to fuse the information together for meaningful analyses. Perhaps the biggest challenge is to deal with the information quality issues of the underlying graphs. One quality of these semantic graphs is that they all contain uncertainties—particular objects and relationships may be missing from the graph or their existence may be suspect or hypothetical. All these require a new generation of analytical tools to effectively understand the semantic graphs.

This paper introduces a new visual analytics framework—known as Have Green—that interactively analyzes semantic graphs with up to one million nodes. Under the new design framework, we are developing new technologies and tools to produce a visual analytics environment that is scalable, ingests both repository graph and graph streams, guarantees interactive responses for query and visualization, runs on multiple computation and display platforms, and most importantly, provides a human computer discourse with walk-in usability for information analytics.

Figure 1 shows the role of Have Green in an interactive graph exploration environment. Have Green fuses the information coming from both the semantic graph repository and the knowledge base before new knowledge is reported. The ultimate goal of Have Green is to produce a working system that enhances investigators' natural analytical capabilities to create, comprehend, and analyze large semantic graphs—allowing investigators to effectively and efficiently perform in an information world that grows more complex daily.

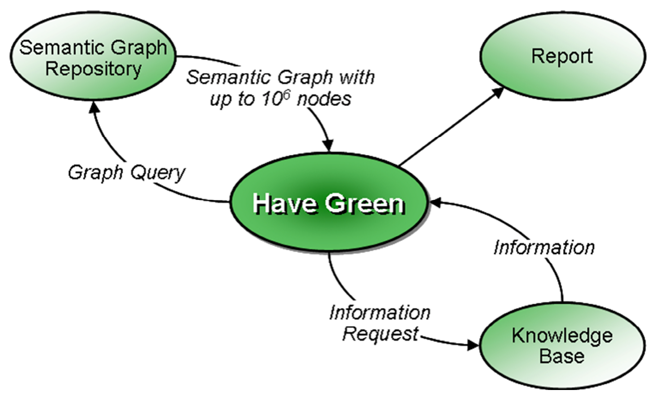

Figure 1: The role of Have Green in an interactive graph exploration environment.

2 RELATED WORK

Different aspects of graph analyses have been studied extensively by diverse communities from multiple disciplines. We highlight some of their work that shares similarities with our approaches.

2.1 Graph Drawing

The graph drawing community has led the studies in most of the graph drawing and layout issues for decades. The two textbooks by Di Battista et al. [8] and Sugiyama [29] summarize most of the major graph drawing algorithms and their applications. The proceedings of the annual Graph Drawing Symposia [9], now in its fourteenth year, and the *Journal of Graph Algorithms and Applications* [13] provide a wealth of information on the cutting-edge technology. The community is also responsible for a series of powerful public domain tools and libraries, including Graphviz [10], JUNG [14], Pajek [23], and Tulip [30].

Email: {pak.wong, george.chin, harlan.foote, patrick.mackey, jim.thomas}@pnl.gov

IEEE Symposium on Visual Analytics Science and Technology 2006
October 31 - November 2, Baltimore, MD, USA
1-4244-0592-0/06/$20.00 © 2006 IEEE

2.2 Graph Visualization

Visualizing graphs and hierarchies have been a major study topic within the data visualization community since its conception in early 90s. The two textbooks by Card et al. [3] and Chen [4] cover much of the major research and applications surrounding graph and hierarchical visualization. The survey paper by Herman et al. [11] represents the most complete literature review up to 2000. The annual IEEE Symposium on Information Visualization [12] continues to produce new results on various topics of graph visualization.

A major difference between the graph visualization and graph drawing communities is that the former almost always involves some sort of interaction, whereas the latter focuses heavily on algorithmic developments. The latest challenge, however, is to integrate the best of the two communities and form a new environment of graph analytics.

2.3 Social Network and Small World Analysis

As defined in Wikipedia [31], a social network is a "social structure made of nodes which are generally individuals or organizations" that "are connected through various social familiarities ranging from casual acquaintance to close familial bonds." Within a social network, people exhibit particular social qualities based on their associations and relationships with other people. For instance, a person who acts as a connection point among multiple social subnetworks is in a position of influence because he or she has close access to many other people. In another example, new ideas and opportunities are more likely to emerge in loosely coupled groups of people with weak associations than tight-knit groups with strong associations because loosely-coupled groups tend to have wider diversities of knowledge and experiences than tightly-knit groups.

Social networks capture and convey social and organizational behaviors and phenomena in a graphical form [32]. Social network graphs or diagrams typically follow a small-world paradigm [22], [33], [34], [35] in that they have high degrees of clustering and small average path lengths relative to their number of nodes. The small-world nature of social networks reflects the concept that people generally organize and link to one another through short chains of associations or acquaintances. Beyond social networks, small-world networks also occur in many other real-world models such as gene regulatory networks and internet network traffic. They are considered a class of random graphs that have been extensively studied in network theory.

2.4 Bio-Molecular Analysis

In biology, different kinds of data and systems may naturally be represented as semantic graphs including metabolic pathways, signaling pathways, gene regulatory networks, protein interaction networks, chemical structure graphs, taxonomies, ontologies, and partonomies. Much of this graph data is stored and managed in public graph databases such as Stanford Research Institute EcoCyc [17], Samuel Lunenfeld Research Institute BIND (Biomolecular Interaction Network Database) [2], University of California at Los Angeles DIP (Database of Interacting Proteins) [42], and Kanehisa Laboratory KEGG (Kyoto Encyclopedia of Genes and Genomes) [16].

Using graph databases, bioinformaticists are generally able to identify and display graphical representations of biological pathways based on a selection of genes, proteins, species, orthologs, and other biological entities. Visualization tools such as the Institute for Systems Biology Cytoscape [25] and Tom Sawyer Software [24] are also available to display biological pathways given a graph specification. Bioinformaticists are generally unable to query against graph databases and visualizations using substructures or patterns within a graph. Furthermore, graph-based results from databases and visualization tools are generally static in the sense that the bioinformaticist may not interact with or manipulate the graphs to understand and explore them. The graphs are simply returned to the bioinformaticist for his or her own personal interpretation.

2.5 The Have Green Tools

We have recently presented a series of working prototypes designed under the framework of Have Green. They include Greenland [37], GreenSketch [39], and GreenArrow [40] to generate, navigate, and visualize large semantic graphs. Case studies based on these technologies have also been used to query graph topology [39] and analyze social networks [41]. More details are given in Section 6 of this paper.

3 Semantic Graphs

We start the paper with the definition of a semantic graph, which is a network of heterogeneous nodes and links annotated with a domain ontology. In our discussion, an ontology of a semantic graph can be considered as a database schema of a relational databases. Figure 2 shows an example of a very simple semantic graph about the relationships among a dozen names with an annotation that lists out some of the potential metadata that may tie to any nodes or links of the semantic graph.

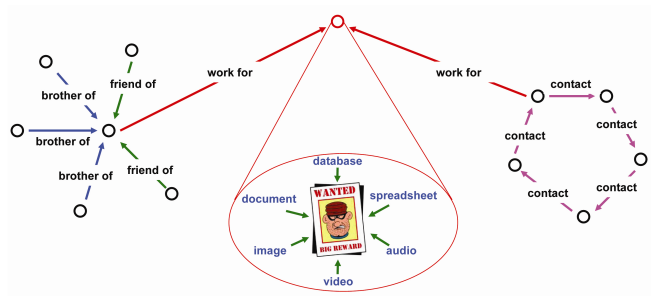

Figure 2: An example of a very simple semantic graph with an annotation that shows potential metadata tied to a graph entity.

In reality, a semantic graph can contain billions of nodes and links in the graph repository for querying. This kind of graph information is usually noisy and loaded with unknown and/or incomplete information. The degree of trustworthiness of any piece of knowledge varies as time goes by as more information arrives to prove or disprove the knowledge.

We can learn a lot from a semantic graph like the one shown in Figure 2. The hierarchy on the left might represent a leader and his followers in a crime. The connections among adjacent nodes on the right might indicate internal communications occurring among a second group of suspects. These two groups might be tied together by a so-called liaison node in the middle. In intelligence analysis, an analyst might want to identify and suppress a liaison node to disrupt collaboration between two groups or to stop a particular scenario from happening. Likewise, a chemistry researcher might wish to remove a liaison node to stop a chemical reaction from occurring.

4 Have Green

Intelligence analysts develop and interact with many kinds of graph and network-based structures and representations in their work and research. Yet, even with this natural emphasis on graphs, analysts have very limited capacity to conduct network analysis on the tremendous amount of graphical data available to them.

Current semantic graph and network analysis tools for intelligence analysis generally aid in the construction and viewing of static graph representations but provide minimal support in the interpretation and analysis of such graphs. On the other hand, a variety of graph-based analytical tools and algorithms are available for defining basic graph representations and conducting general graph operations and queries, but these capabilities exist at a level of abstraction that is inaccessible and incomprehensible to analysts. The aim of the Have Green visual analytics framework is to fill this theoretical and developmental void by creating an analytical environment in which analysts may conduct network analysis in terms, concepts, and a language that is intuitive and meaningful to them.

Have Green is the codename that collectively represents a suite of visual analytics technologies developed recently at PNNL to support the analytical goal of large semantic graphs. Have Green is not merely a set of disparate graph analysis tools but rather a comprehensive, interactive graph exploration environment that provides advanced visual capabilities for querying, navigating, and visualizing large semantic graphs. Figure 3 depicts a system overview of Have Green and its major components.

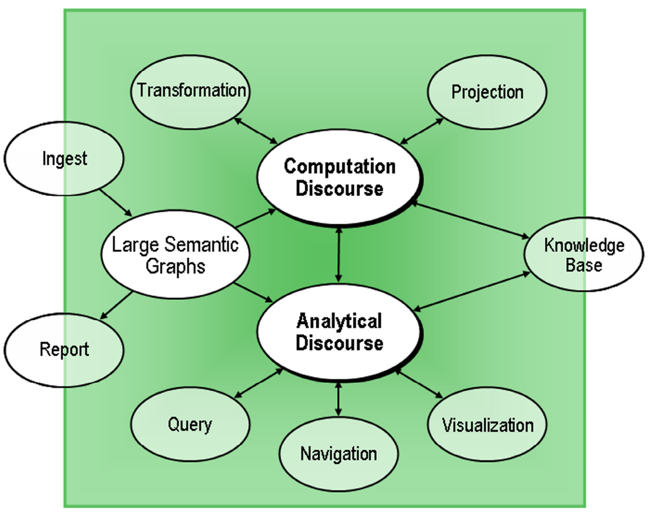

Figure 3: A framework overview of Have Green.

4.1 Framework Overview

As more graph data from different sources is fed into a semantic graph, the attributes and relationships in the graph grow increasingly complex, and the ability of the analyst to comprehend the graph data degrades. Consequently, analysts need the ability to extract specific features or views from a complex, multi-dimensional semantic graph. For instance, an analyst may want to extract slices from a complex semantic graph to examine specific classes of objects and relationships such as a timeline of events, organizational hierarchy of people, communication lines among a group of suspects, or the physical exchange of some material or chemical. In this way, the analyst makes sense of a complex situation by digesting and examining specific dimensions of the situation and then integrating across perspectives to capture and realize a fuller picture.

To facilitate the above kind of multi-perspective analysis, Have Green must be able to generate internal models from semantic graphs that will afford different kinds of analyses. These internal models must then be presented to the analysts in human usable forms. To accomplish this, Have Green must engage in a *computational discourse* between the semantic graphs and the visual analytics framework and an *analytical discourse* between the visual analytics framework and the analyst (see Figure 3). In both discourses, graphs should be integrated with other forms of domain knowledge to facilitate more comprehensive analyses.

4.2 Graph Ingest

Have Green is capable of ingesting both static semantic graphs that appear as single files and dynamic transient graph streams that arrive continuously and unpredictably without regular patterns. Both of them post direct challenges to our promise of an interactive visual analytic tool.

4.2.1 Very Large Static Semantic Graphs

We have so far encountered no problems to ingest a plain graph with up to a million nodes in interactive time. We have, however, seen major issues when we attempt to ingest both the graphs and their metadata together, and do so in interactive time.

We are in the process of developing new approaches to rapidly scan these metadata, bring down their resolutions, and only maintain coarse versions of these metadata in memory. In many cases, only the signatures [38] of the metadata will be kept with the graph after the ingest step. These very small but information-rich data signatures become our key to meeting the interactive response time challenge.

4.2.2 Time Varying Transient Graph Streams

Graph streams analytics not only inherit most of the problems and issues of traditional data streams [28] defined by the databases community, but their visual-requirement also creates a few new issues when we design Have Green. One major challenge is to maintain the shape of the graph visualization when new streams arrive and are integrated.

Di Battista et al. [8] suggest multiple algorithms that support different types of *constraints*, which can be used to address some of our problems. Additionally, we have previously investigated some of the similar issues on text- and sensor-streams [36] and developed a multidimensional scaling [6] (MDS) -based solution. Because there is equivalence between the "stress" function used in the non-metric MDS algorithm developed by Kruskal [19] and the "force-directed" function used in the graph layout algorithm developed independently by Kamada and Kawai [15], we expect to come up with a new solution similar to our work in [36].

4.3 Computation Discourse

In computational discourse, Have Green retrieves semantic graphs from repositories, and transforms and projects them into internal abstract models. These internal models are applied by analysts to perform different kinds of analyses.

4.3.1 Transformation

Data transformations allow analysts to convert data and its associated data model to equivalent representations so as to highlight specific features of the data with minimal loss of information. For instance, semantic graphs often import from and export to table or spreadsheet views. Graphs and spreadsheets may generally consist of the same information, but different aspects of the information are highlighted. Graphs tend to emphasize relationships while tables emphasize the entities.

In another transformation example, the edges of a semantic graph may be translated onto an adjacency matrix where the row and column numbers of a matrix element map to the two nodes in the corresponding semantic graph that are connected by the associated edge. The adjacency matrix is a traditional, equivalent representation of a semantic graph that captures the same information, but emphasizes different attributes or features. For example, the sparsity of a graph is better illustrated through an adjacency matrix than a graph representation. Furthermore, graphs and adjacency matrices are amenable to different kinds of analyses. For example, graphs are amenable to social and other types of network analyses while adjacency matrices are amenable to linear algebraic and eigenstructure analysis.

4.3.2 Projection

Projections map data into alternative data models where the underlying meaning and context of the data shifts. For example, the nodes of a semantic graph may be projected onto a scatterplot, where the distance between any two nodes corresponds to their similarity based on some attribute or feature (e.g., topology, label semantics, time of occurrence). For the original graph, the spatial distance between two nodes carry no inherent meaning, but with the scatterplot, new meaning is introduced and associated with the spatial measure. The general effect of the projection is to make an abstract concept such as topology more interpretable by projecting it upon features that may better sensed and experienced.

With projections, analysts need to comprehend what underlying measures mean, but not necessarily the algorithm or mechanism used to generate the projection. Regarding the scatterplot above, for example, an analyst will accept that the distance between two nodes accurately reflects the nodes' similarity if the correlation conforms to the analyst's general observations and experiences. Analysts need not be aware of how the scatterplot projection is generated to have confidence in its fidelity and accuracy.

For example, bioinformaticists have accepted and are extensively applying BLAST (Basic Local Alignment Search Tool) [1] to search for similar nucleotide or protein sequences. Though very few bioinformaticists are familiar with the complex, statistical code that computes the similarity, the bioinformatics community has accepted and embraced BLAST as an essential and valid analysis tool.

4.4 Analytical Discourse

In facilitating analytical discourse, we wish to allow analysts to interact with semantic graphs in ways that are natural and intuitive. In previous studies [5], we have examined how analysts deploy and apply different kinds of semantic graphs (hand-drawn or computer generated) in intelligence analysis. Analysts use graphs to capture concepts, search relationships and connections, survey the full context of a situation, and identify critical patterns and trends. To best facilitate semantic graph exploration, the interaction and dialogue between analysts and graphs should be supportive and consistent with the above kinds of tasks.

In analytical discourse, three general visual capabilities are essential for exploring and working with graphs. These are:

- Querying – searching a semantic graph for particular nodes, links, or subgraphs based on labels, properties or metadata, and/or topology
- Navigation – moving across a semantic graph at the same resolution, or up and down through different resolutions
- Visualization – presenting a semantic graph through different views and perspectives to highlight critical concepts and insights

4.4.1 Query

In a directed query, an analyst searches for specific entities, subjects, people, locations, and/or objects in the search. Additionally, the analyst may search for specific relationships such as the exchange of money or contraband, or organizational and familial relationships. The query is conducted along a specific topic, theme, or association that is central to the investigation.

In other cases, the analyst may not necessarily have a specific topic, theme, or association in mind. Rather, the focus of the investigation is to identify patterns or trends in the graph data. For example, with computer network data, an analyst may wish to locate computer nodes with high or anomalous activity to identify potential sources of an intrusion or denial of service attack. In such a case, the analyst does not begin the investigation with an initial identifying node, but rather looks towards the graph for patterns or features that stand out.

Queries do not necessarily need to always be initiated by the analyst. Intelligent systems may be developed to semi-automatically detect relevant graph patterns and present them to the analyst. A desirable interface would support both user query and system guided modes that effectively support a "give and take" exchange or discourse between the analyst and the visual analytics framework. This kind of interactive system is often referred to as a "mixed initiative" system, where either the user or the system may initiate interaction.

4.4.2 Navigation

The user and system-initiated visual queries described above combine to promote a general navigation strategy that analysts often employ. In our study of analysts conducting link analysis, we found that analysts will often look over the full structure of a semantic graph and mentally partition the graph into natural clusters of high activity or dense subgraphs. Analysts then drill down into specific clusters in hopes of characterizing the general topics and organizations of those clusters. For example, an analyst might find a particular cluster to represent the hierarchy of an organization or group such as Al Qaeda, or the presence of a biological agent such as Anthrax in a number of terrorist incidences or at different geographic locations. Once a set of clusters have been characterized, the analyst may then pull his focus back out to the larger view to examine how the different topics and organizations interact and relate to one another. The analyst might consider a link between two terrorist groups to identify a potential collaboration between groups or a link between a terrorist group and a biological agent to identify the terrorist group's biological weapon of choice, which then becomes a defining characteristic of that terrorist group.

In general, the analyst follows an iterative investigation path that continually switches from looking at the general structure of the graph to examining local graph content. For large, complex graphs, the overall number of clusters in the graph may become prohibitively large as analysts lose their ability to track and manage the full set of clusters. A more useful navigation approach might be to present the graph at multiple levels of resolution. In this approach, the analyst may need to drill down several levels of resolution before reaching a singular working concept that may be analyzed in the context of other local concepts. In such cases, the analyst will often recursively drill down into more specific and detailed concepts and then successively assemble concepts into contexts on the way up.

4.4.3 Visualization

Analysts are accustomed to traditional views of semantic graphs as nodes and edges. The key property or characteristic of semantic graphs are the relationships among objects they convey. Analysts review semantic graphs to inspect object interactions and organizations such as transactions, processes, groups, compositions, and infrastructures.

Apart from relationships, analysts may wish to investigate data in other forms and models. In some cases, analysts may wish to examine just entities or just relationships. For example, an analyst may wish to organize people into different organizations or groups based on membership or other criteria, or examine the full set of transactions from a particular bank in chronological order. Given these needs, a table may be more appropriate as an end-user representation than a graph since it highlights different aspects of the data such as classification and order.

As previously described, an objective of Have Green is to provide multiple perspectives of the same data such that it may be explored and analyzed in comprehensive and integrative ways. Analysts want to identify critical patterns and insights in the data, which may be best accomplished by allowing analysts to visually view and manipulate the data along different dimensions and perspectives.

4.5 External Knowledge Base

Semantic graphs represent one particular type of data that needs to be integrated with other domain knowledge, which may appear in various forms such as hypotheses, documents, ontologies, dictionaries, and relational databases. As shown in Figure 3, this additional domain knowledge needs to be integrated with graphs through both computational and analytical discourse.

4.6 Report

The *Report* component in Figure 3 covers the general requirements of organizing the analytical results, presenting them to the investigators, and later sharing them with a wider audience. In addition to the traditional concept of a report that neatly lays out the information on screens or printouts, we develop the concept of a dynamic report that allows the audience to participate in the analyses with the evidence included in the report.

A major challenge for Have Green is to automatically generate reports with different degrees of details customized for different audiences in real time. In the finest scale, a report will include all the pieces of evidence stored in their original formats that contribute to the conclusions. For example, a Have Green report may contain, among other things, a portion or partition of a semantic graph, segments of certain surveillance videos, a video facial recognition software, a database of driver license photos, and a visualization that ties everything together with a conclusion.

Software is included in some of the reports because their audiences may need to, for example, adjust the parameters of the recognition program and review the evidence from a different perspective. Sensitive information may require passwords to gain access. The dynamic report itself is indeed a storytelling mechanism that allows its audience to follow through the evidence and review the results. It can also be treated as a collaboration medium that is equipped with required tools and local databases for further analyses.

5 Implementation Details and Issues

The design requirement of Have Green is enormous but manageable. We champion software reusability and practice modular design throughout the development stage.

After the Have Green architecture is formally established, individual components are implemented separately so that we can pinpoint our design weaknesses in the earliest stage. Each component system undergoes multiple usability studies with subjects recruited at the lab. Evaluation results collected from the studies and post-study interviews are used to further revise our designs. These individual components eventually become the building blocks of Have Green.

With the exception of the LAPACK [20] library that is used to compute Eigenvectors of the graph matrices, all the system code is developed locally in compiled Java and C++ codes.

6 Major Accomplishments to date

As previously described, Have Green assembles and integrates capabilities from a series of working prototypes. Each of these prototypes delivers unique and critical capabilities that support key aspects of both computational and analytical discourse. To date, we have developed four major system prototypes (Greenland [37], GreenSketch [39], GreenArrow [40], and GreenMonster) to support Have Green components in Figure 3. While they are designed with a single functionality in mind, all of them come with input/output functions and a high degree of interactive features so that we can execute and evaluate them independently. With the exception of GreenMonster, which is an ongoing development, the usability study results of individual prototypes are included in the corresponding papers.

6.1 Greenland

Greenland [37] allows analysts to *navigat*e and explore large semantic graphs. It provides a traditional directed graph view that may be panned, zoomed, modified, and linked to metadata. Furthermore, the directed graph may be *projected* onto a scatterplot that will permit analysts to examine similarities and distinctions among selected nodes and subgraphs – allowing analysts to identify similar structures or patterns in the graph that may not be visible to the naked eye.

Greenland is our first prototype intended to navigate large semantic graphs using the concept of a data signature [38]. A data signature, in this case, is a multidimensional vector that captures the local topology information surrounding each graph node. The goal is to describe and represent different topological structures as numerical vectors and then use these vectors for different analytical purposes.

For example, we suggested in [37] that the signature of a d-degree undirected graph node can be defined as a vector $(n_1, n_2 \ldots n_d)$ where n_i is the number of the nodes at distance i from the node. Based on this definition, Greenland first extracts signature vectors from a sparse graph and then projects the vectors onto a low-dimensional scatterplot through the use of multidimensional scaling (MDS) [6]. The resultant scatterplot, which reflects the similarities of the vectors, allows users to examine the graph structures and their corresponding real-life interpretations through repeated use of brushing and linking [6] between the two visualizations. Figure 4a shows a snapshot of Greenland with a small world network. Figures 4b-4d demonstrate the linking and brushing process between a graph and a MDS scatterplot generated using the signatures extracted from the graph.

6.2 GreenSketch

While Greenland provides a way to browse a large graph and look for clues, GreenSketch [39] provides a graphical interface needed to support the *query* component of Have Green. By sketching lines, curves, and patterns on an interactive adjacency matrix, analysts may easily create different kinds of rich and expressive graphs that convey real-life patterns and scenarios. Rather than building a graph node-by-node and edge-by-edge, the graph is generated and *transformed* through the adjacency matrix. The constructed graph may then be applied as a prototypical pattern for which to be queried in larger graphs of known or emerging facts and situations.

GreenSketch is indeed an interactive graph generator originally designed to facilitate the creation of descriptive graphs required for multiple analytics tasks. The human-centric design approach of GreenSketch enables analysts to master the creation process without specific training or prior knowledge of graph model theory. The customized user interface encourages analysts to gain insight into the connection between the compact matrix representation and the topology of a graph layout when they sketch their graphs. Both the human-enforced and machine-generated randomness supported by GreenSketch provide the flexibility needed to address the uncertainty factor in many

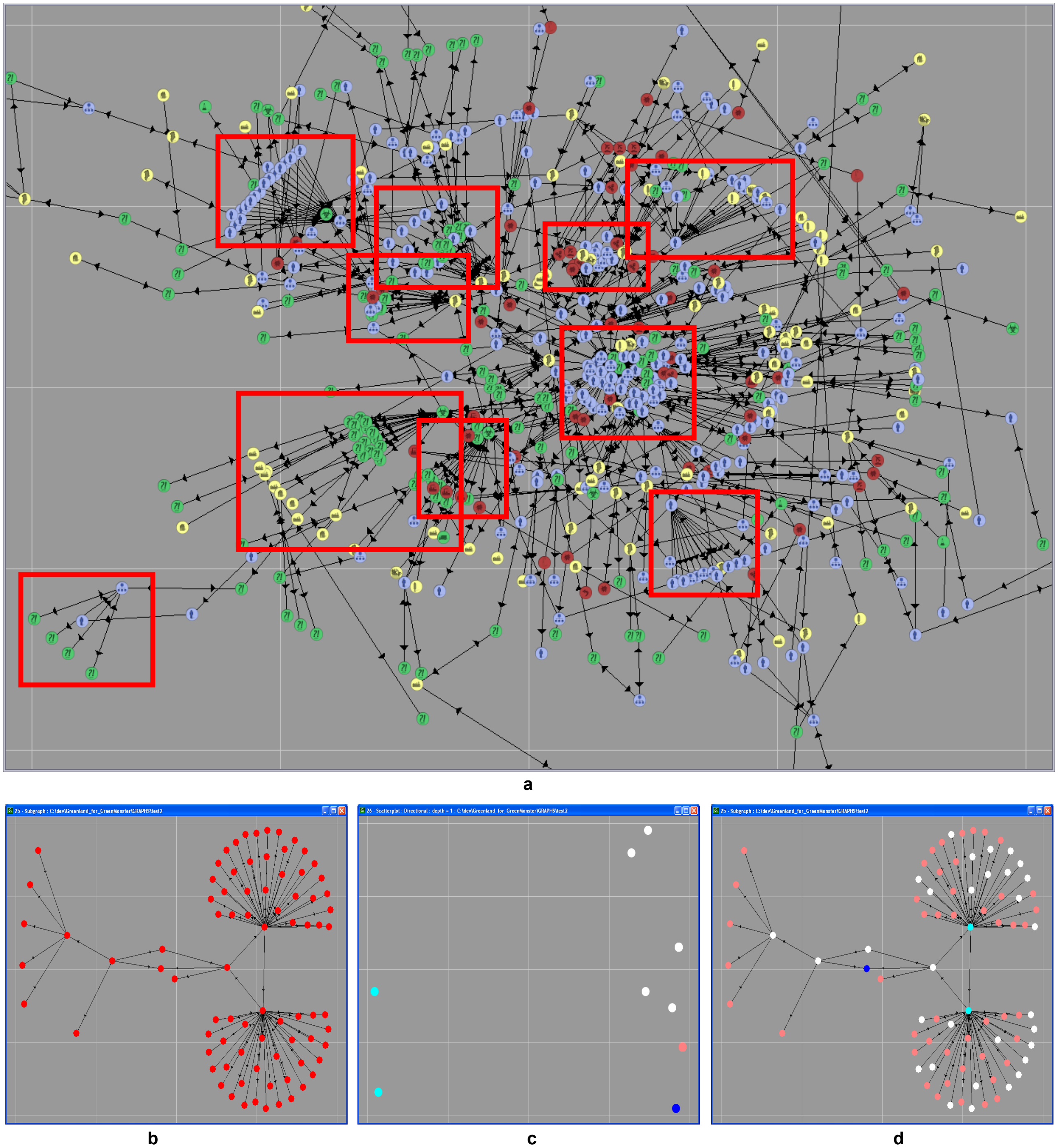

Figure 4:a) Greenland visualizes a small world network with major hierarchies highlighted by the red rectangles. b) A portion of the graph in a). c) A scatterplot generated by scaling the signatures in b). d) Brushing and linking between the scatterplot and the graph.

analytical tasks. Figure 5 depicts two GreenSketch examples of creating graph queries by sketching.

In [39], we demonstrate GreenSketch as a query language tool to study structural features hidden behind a semantic graph. Graph entities that share similarities with the query are correctly identified and extracted from a large semantic graph. More elaborate implementation is under development to support more complicated queries.

6.3 GreenArrow

A hallmark signature of a semantic graph is the rich semantics of its individual nodes and links. Node and edge labels may convey a tremendous amount of information and context, where they may include graph metadata that ranges from a short phrase to a full sentence to an entire paragraph and beyond. Yet supporting such richness and detail in graph labels require new visualization approaches that would allow analysts to better view and comprehend the fuller and more saturated information. To this end, we have developed a practical *visualization* prototype, known as GreenArrow [40], to visualize semantic graphs with extended nodes and link labels.

Our solution is different from all the existing approaches that almost always rely on intensive computational effort to optimize the label placement problem. Instead, labels are programmatically

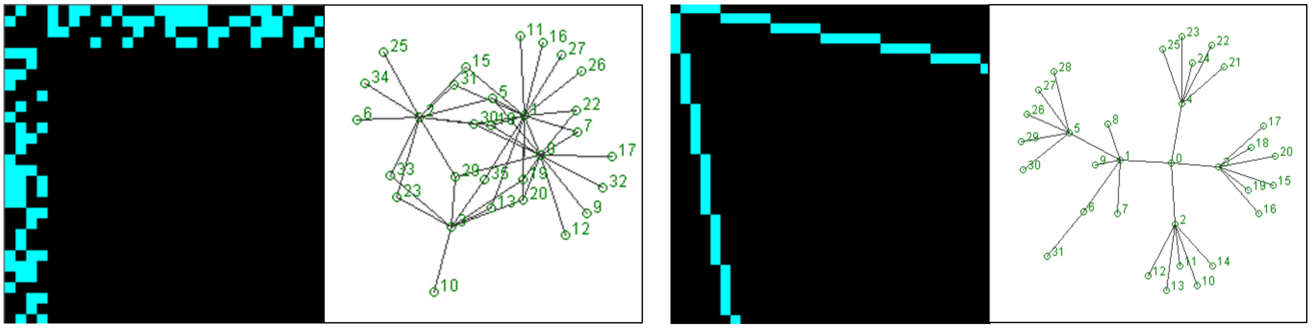

Figure 5: GreenSketch generates two small world graphs in the white board by sketching on the corresponding (black) matrix windows.

and visually integrated into the edges and nodes of the graph where they are presented in static, interactive, and dynamic modes without the requirement for tackling the intractability issues. This allows us to reallocate the computational resources for dynamic presentation of real-time information. Figure 6 shows an example of a social network among a group of people.

Our results indicate that our lightweight solution executes faster and requires less drawing space than most of the traditional techniques. It also performs better in our user-evaluation studies in both static and dynamic modes as reported in [40].

6.4 GreenMonster

GreenMonster is our latest Have Green addition that addresses the scalability issue of our large semantic graphs. The requirement is to provide a capability to visualize semantic graphs with up to one million nodes adaptively and interactively on both desktop computers and PDAs. While GreenMonster belongs to the *projection* component in Figure 3, it also supports the *visualization* component that is under our design's analytical discourse hierarchy. GreenMonster is currently undergoing evaluation.

7 The Next Steps

The essence of science and intelligence analysis is the discovery of new facts, concepts, and insights. Through richness of information, semantic graphs provide a fertile media from which to engage in knowledge discovery. Yet, as we have described in this paper, large semantic graphs have confounding attributes such as complexity, size, and uncertainty that blurs the analyst's vision and prohibits him or her from finding those proverbial needles in the semantic haystack.

Have Green was designed to facilitate knowledge discovery by providing analysts enabling methods and tools to query, navigate, and visualize large semantic graphs. It is a graph analytics platform or environment rather than a finished product. New technology and working prototypes will continue to be included in the framework. More than a suite of tools, however, Have Green provides analysts different models and views of graphs (e.g.,

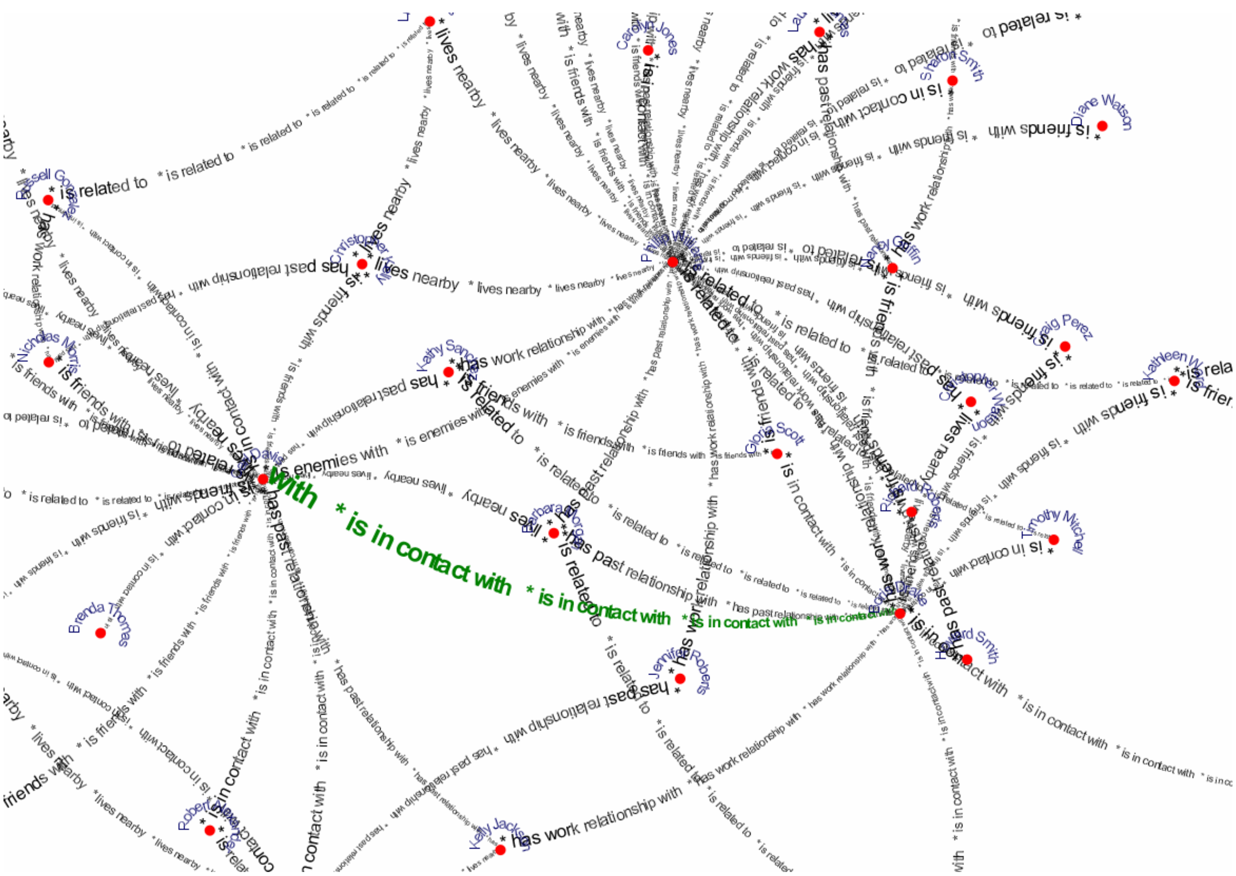

Figure 6: A screen snapshot of the GreenArrow visualization.

scatterplots, adjacency matrixes, data signatures). Through such alternative models, Have Green allows analysts to examine graphs from different angles and perspectives. An intriguing quality of many of the Have Green tools is that they allow analysts to view, comprehend, search, and manipulate semantic graphs without requiring analysts to see and work with traditional graph structures. In continually adding to the Have Green platform, our goal is to allow analysts to forever extract richer information from large semantic graphs through both richer analysis tools and richer interactions with those semantic graphs.

8 CONCLUSIONS

We discuss major challenges of developing a semantic graph analytics system and present a working visual analytics framework—known as Have Green—that addresses many of these challenges. The paper explains the rationale behind our design, showcases four major working prototypes, and suggests upcoming efforts to develop the rest of the Have Green components.

ACKNOWLEDGEMENTS

This work has been sponsored in part by the National Visualization and Analytics Center™ (NVAC™) located at the Pacific Northwest National Laboratory in Richland, WA. The Pacific Northwest National Laboratory is managed for the U.S. Department of Energy by Battelle Memorial Institute under Contract DE-AC05-76RL01830.

REFERENCES

[1] Stephen F. Altschul, Warren Gish, Webb Miller, Eugene W. Myers, and David J. Lipman, "Basic Local Alignment Search Tool," *Journal of Molecular Biology*, volume 215, pages 403-410, 1990.

[2] Gary D. Bader, Doron Betel, and Christopher W. W. Hogue, "BIND: The Biomolecular Interaction Network Database," *Nucleic Acids Research,* volume 31, number 1, pages 248-250, Oxford University Press, 2003.

[3] Stuart K. Card, Jock Mackinlay, and Ben Shneiderman, *Readings in Information Visualization, Using Vision to Think*, Morgan Kaufmann, 1999.

[4] Chaomei Chen, *Information Visualization beyond the Horizon*, second edition, Springer, 2004.

[5] George Chin Jr., Olga Kuchar, Paul Whitney, Mary Powers, and Katherine Johnson, "Graph-Based Comparisons of Scenarios in Intelligence Analysis," *Proceedings of the 2004 IEEE International Conference on Systems, Man and Cybernetics*, pages 3175-3180, Oct 2005.

[6] William S. Cleveland, *Visualizing Data*, Hobart Press, 1993.

[7] Trevor F. Cox and Michael A.A. Cox, *Multidimensional Scaling*, second edition, Chapman & Hall/CRC, 2001.

[8] Giuseppe Di Battista, Peter Eades, Roberto Tamassia, and Ioannis G. Tollis, *Graph Drawing: Algorithms for the Visualization of Graphs*, Prentice Hall, 1999.

[9] Graph Drawing (GD) 2006, http://gd2006.org.

[10] Graphviz, http://www.research.att.com/sw/tools/graphviz/.

[11] I. Herman, G. Melancon, and M.S. Marshall, "Graph Visualization and Navigation in Information Visualization: A Survey," *IEEE Transactions on Visualization and Computer Graphics*, volume 6, number 1, pages, 24-43, IEEE CS Press, 2000.

[12] IEEE Symposium on Information Visualization (InfoVis) 2006, http://conferences.computer.org/infovis/infovis2006/.

[13] Journal of Graph Algorithms and Applications, http://jgaa.info.

[14] JUNG—Java Universal Network/Graph Framework, http://jung.sourceforge.net/faq.html.

[15] Tomihisa Kamada and Satoru Kawai, "An Algorithm for Drawing General Undirected Graphs," *Information Processing Letters*, volume 31, issue 1, pages 7-15, Elsevier North-Holland, Apr 1989.

[16] Minoru Kanehisa and Susumu Goto, "KEGG: Kyoto Encyclopedia of Genes and Genomes," *Nucleic Acids Research*, volume 28, pages. 27-30, 2000.

[17] Peter D. Karp, "Pathway Databases: A Case Study in Computational Symbolic Theories," *Science,* volume 293, issue 5537, pages 2040-2044, 2001.

[18] Tamara Kolda, David Brown, James Corones, Terence Critchlow, Tina Eliassi-Rad, Lise Getoor, Bruce Hendrickson, Vipin Kumar, Diane Lambert, Celeste Matarazzo, Kevin McCurley, Michael Merrill, Nagiza Samatova, Douglas Speck, Ramakrishnan Srikant, Jim Thomas, Michael Wertheimer, and Pak Chung Wong, *Data Sciences Technology for Homeland Security Information Management and Knowledge Discovery, Report of the DHS Workshop on Data Sciences*, Jointly released by Sandia National Laboratories and Lawrence Livermore National Laboratory, Alexandria, VA, 2004.

[19] Joseph B. Kruskal, Nonmetric Multidimensional Scaling: A Numerical Method, *Psychometrika*, volume 29, number 2, pages 115-129, Mar 1964.

[20] LAPACK, http://www.netlib.org/lapack.

[21] Vernon Loeb, "Clan, Family Ties Called Key to Army's Capture of Hussein: 'Link Diagrams' Showed Everyone Related by Blood or Tribe," *Washington Post*, pages A27, Dec 16, 2003.

[22] Stanley Milgram, "The Small World Problem," *Psychology Today*, volume 2, pages 60-67, 1967.

[23] Pajek, http://vlado.fmf.uni-lj.si/pub/networks/pajek/.

[24] Tom Sawyer Software, http://www.tomsawyer.com/.

[25] Paul Shannon, Andrew Markiel, Owen Ozier, Nitin S. Baliga., Jonathan T. Wang, Daniel Ramage, Nada Amin, Benno Schwikowski, and Trey Ideker, "Cytoscape: A Software Environment for Integrated Models of Biomolecular Interaction Networks," *Genome Research,* volume 13, number 11, pages 2498-2504, 2003.

[26] Mindy Sink, "An Electronic Cop that Plays Hunches," *New York Times*, pages B9, Nov 2, 2002.

[27] Malcolm K. Sparrow, "The Application of Network Analysis to Criminal Intelligence: An Assessment of the Prospects," *Social Networks*, volume 13, pages 251-274, 1991.

[28] Stanford Data Stream Manager, http://www-db.stanford.edu/stream/.

[29] Kozo Sugiyama, *Graph Drawing and Applications*, World Scientific Publishing, 2002.

[30] Tulip, http://tulip-software.org/.

[31] Wikipedia, http://www.wikipedia.org.

[32] Stanley Wasserman and Katherine Faust, *Social Network Analysis-Methods and Applications*, Cambridge University Press, 1999.

[33] D.J. Watts, *Small Worlds*, Princeton University Press, 1999.

[34] D.J. Watts and S.H. Strogatz, "Collective Dynamics of 'Small-World' Networks," *Nature*, pages 440-442, Macmillan, 1998.

[35] D.J. Watts, Six Degrees: The Science of a Connected Age, W.W. Norton & Company, 2003.

[36] Pak Chung Wong, Harlan Foote, Dan Adams, Wendy Cowley, and Jim Thomas, "Dynamic Visualization of Transient Data Streams," *Proceedings IEEE Symposium on Information Visualization 2003*, pages 97-104, Oct 2003.

[37] Pak Chung Wong, Harlan Foote, George Chin Jr., Patrick Mackey, and Ken Perrine, "Graph Signatures for Visual Analytics," *IEEE Transactions on Visualization and Computer Graphics*, volume 12, number 6, Nov-Dec 2006.

[38] Pak Chung Wong, Harlan Foote, Ruby Leung, Dan Adams, and Jim Thomas, "Data Signatures and Visualization of Very Large Datasets," *IEEE Computer Graphics and Applications*, volume 20, number 2, IEEE CS Press, 2000.

[39] Pak Chung Wong, Harlan Foote, Patrick Mackey, Ken Perrine, and George Chin Jr, "Generating Graphs for Visual Analytics through Interactive Sketching," *IEEE Transactions on Visualization and Computer Graphics*, volume 12, number 6, Nov/Dec 2006.

[40] Pak Chung Wong, Patrick Mackey, Ken Perrine, James Eagan, Harlan Foote, and Jim Thomas, "Dynamic Visualization of Graphs with Extended Labels," *Proceedings IEEE Symposium on Information Visualization 2005*, pages 73-80, Oct 2005.

[41] Pak Chung Wong, Ken Perrine, Patrick Mackey, Harlan Foote, and Jim Thomas, "Visual Analytics and Storytelling through Video," *Proceedings Compendium IEEE Symposium on Information Visualization 2005*, pages 79-80, Oct 2005.

[42] Ioannis Xenarios, Lukasz Salwinski, Xiaoqun Joyce Duan, Patrick Higney, Sul-Min Kim, and David Eisenberg, "DIP: The Database of Interacting Proteins. A Research Tool for Studying Cellular Networks of Protein Interactions," *Nucleic Acid Research,* volume 30, number 1, pages. 303-305, Oxford University Press, 2002.