

DATA VISUALIZATION SOFTWARES USING JAVA

Vaishali Jabade¹, Shrinath Palwankar²

vaishali.jabade@vit.edu1

shrinath.palwankar19@vit.edu2

Vishwakarma Institute of Technology, Pune¹

Abstract: *The adage "a picture is worth a thousand words" perfectly describes the goal of data visualization. Goal of data visualization is that to present information in a concise, yet understandable manner. The user can use visualizations to emphasize important points and provide useful information, such as summaries of data. Data visualization is the conversion of numbers and raw data into visual objects (points, bars, line charts, maps, etc.). These visualizations, combined with user-friendly and aesthetically pleasing features, can significantly speed up research and data analysis while at the same time being powerful communication tools.*

Keywords: *visualization, visual objects, communication tool*

1. INTRODUCTION

In this paper we have studied all the current available data visualization tools. Data visualization is the process of visualizing information in a graphical or pictorial format. It has become a field of improvement and research that is firmly associated with data structures and perception, factual illustrations, and logical representation. The goal the paper is to provide a review of plans and usage strategies for information perception. Users can gain insight into their huge amounts of data through data visualization. This allows them to identify new patterns and bugs in the data. Making sense of these patterns allows the users pay attention to areas that represent changes or process. They can identify new patterns and bugs in the data as a result. Understanding these patterns allows the user to pay attention to areas that represent red flags or progress.

2. RELATED WORK

Software visualization Recent research. Magnify closest Displays information about the infrastructure of the software. The level of the component and the code it contains. Enlarge the model. A backend data store database but collect Source code information only. Cannot capture the entire software workflow. JCVHT made a decent contribution to A view of the ranking of software components. that is Relationships in software components but take the whole picture. Tools like CodeCrawler and JIVE also Stargate are also very gripping attempts. All below tools try to examine the software code. However, neither of them could be delivered at the same time Information about the organization of a software entity Software workflow patterns.

3. BACKGROUND

The data visualization software node connectivity in networking is displayed as visualization form and can be located at dissimilar locations. It is possible to communicate raw data efficiently, and users can design different types of data visualization patterns depending on their requirements. The Java data visualization software has the following goals and objectives: Graphical or pictorial representation of information that is clear and effective. Functionality and aesthetic form should go together; neither should look boring or overly sophisticated. Convey ideas effectively by providing insight into complex sets of data and information. Achieve a balance between the function and the form of software. Ensure viewers' awareness and engagement while effectively communicating info. Data visualization used for many different fields, including biotechnology, education as well as

research, and many more, to communicate and understand information more easily. Keep this software's features and scopes broad rather than limiting them.

4. WHY JAVA

Since the repository data was collected using the GitHub API, the language used in that data is returned for each repository. Figure 1 shows the distribution of the top 20 languages by a set of repositories.. It is visible that JavaScript is the most popular language among to the distribution which has followed by Ruby then Python and Java and so on. Data was collected randomly; it was the distribution in random years. The languages whom are used more within the project are displayed and that is the automatic field got by the GitHub.[4]

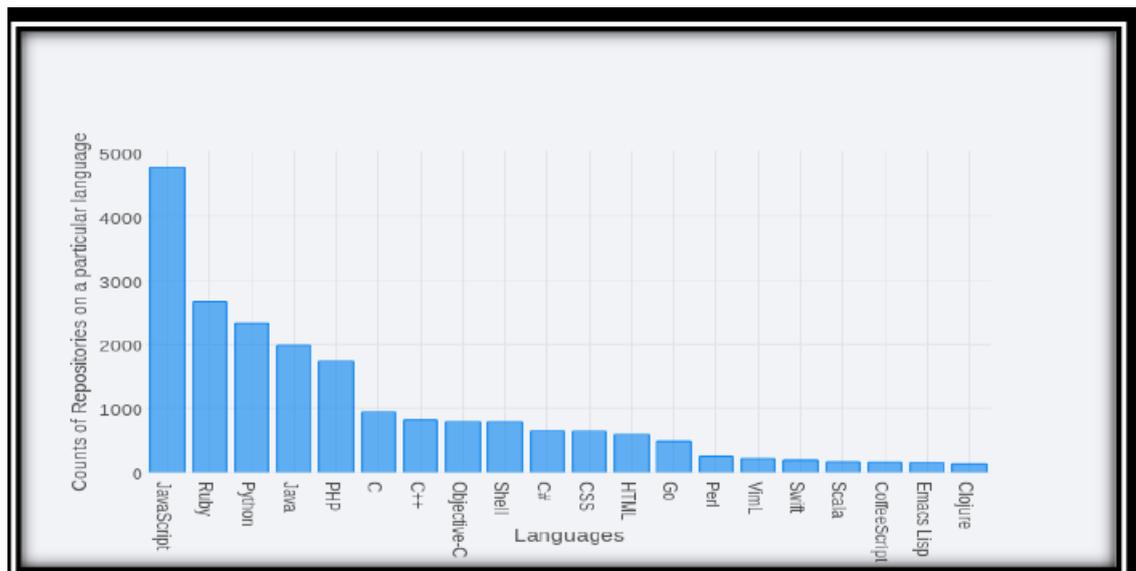


Figure 1. Top 20 Languages by number of repositories [4]

As you can see, JavaScript also Python, and Java, Go, shell scripts, Obj C, and Care are more prominent languages, making them more comfortable for users to interact with. Users are less interested or slow in adapting the language. This is because there are a few highlights like perl, viml, scala, haskell, swift, Languages such as. Assembly Prolog, Jupyter , Cuda is too old, new or infrequently used, so it rarely shows its size. These languages are extracted from the Description field.

5. DATA VISUALIZATION TOOLS

5.1) Data visualization tools are software applications that render information in visual formats such as graphs, charts, heatmaps, etc. for the purpose of data analysis. These tools make it easier to understand and manipulate large amounts of data. Data visualization tools make it easier to understand and manage large amounts of data.

5.2). Benefits of visualizing numerical data: There are many practical parameters or variables that are subject to important data collection, and it is not easy to observe them.

3.1C). The problem with physical phenomena is that the human eye cannot see objects smallest than the visual resolution. For e.g., you can't see atoms or molecules, but electronic sensors can collect important physical calculation by experimenting with operate them. The powerlessness of human perception to resolve time intervals for fast events e.g. Flying bullets. The moment you think you hear the gunshot the bullet is already inside in the target.

Engineers also prefer another tech such as B. High-speed digital photo-graph that captures the path of a bullet frame by frame. The vortex sensor records the path the bullet allows for it. Every data is sent to the computer software visualization package for analysis. The numerical data captured by the gauge is too broad to be fully analyzed by hand or non-visual software, so without visual analysis, the engineer would need a complete understanding of the path of the ball. Visualization gives you a wider view of the problem group you are investigating. That is you can observe the problem from completely different angles. Non visual software's returns what the search algorithm finds, but it is unclear whether the solution is local. It doesn't shows whether the slight difference between solution is still nearly sufficient or incorrect. The figure (Figure 2) illustrates this point.

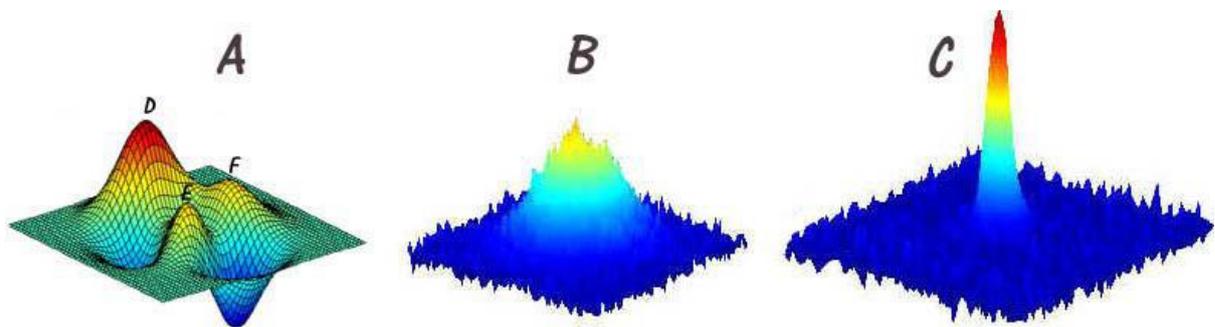


Figure 2. A, B, C Peaks []

All plots in the figure are peak plots. Plato has 3 peaks D, F, E and 2 valleys. Both Plot B and Plot C are one peaks with dissimilar sharpness as well as height. If the algorithm for finding peaks specifies an initial range (x and y) and specified range does not have the highest peak Plot D in Plot A in the figure 2. This info produces the following small peaks: Solution if Peak E, Peak F, depending on the search. However, if you look at Chart A, you can clearly see that Peak E, Peak F is the local maximum, not the global maximum Peak D (in Chart A). (PeakD) was outside in the search estimation range. For regions (domains) containing global maximums, it is fastest to visualize the data before fitting, as no further effort can be made to infer that the entire XY plane is infinite. Get the best solution an optimal search algorithm is the basis of the neural n/w in which the search starts at the first estimation point and descends in a direction that reduces the penalty of its motion unless a solution is found.

6. JAVA TOOLS – VI-SAD (Visualization Algo Development)

Vis-AD is a Java library API for immersive and unite visualization of numerical data. The advantage of the library is that it doesn't require any knowledge of Java 3D. it uses Java 3D. This reduces the requirement to handle low-level features.

The VisAD system combines the following reinforce for data exchange and real-time alliance between dispersed users. Distributed computing is added into the lowest level of the system uses distributed Java R.M.I objects A easy math data model that can suitable virtually any numerical data and support data exchange between dissimilar users as well as dissimilar data sources. A popular display model that reinforce interactive 3D, data fusion and multiple data as well as in place editing, collaboration, and VR.

Display model is consistent with Java 3D and Java 2D. Data analysis as well as computation are integrated into a visualization to help computer-controlled and other compound interchange modes. Extensibility for developers in as possible ways.

7. JIVE: JAVA INTERACTIVE SOFTWARE VISUALIZATION ENVIRONMENT [1]

It is the possible solutions to minimize the cost of animation The algorithm is used enhance re-usability. In other words, it saves effort. Spent as much as possible reusing and animating the algorithm's recent work. For e.g. that is advantageous to Reuse animation code that developed to debug complex things and Animated algorithms while teaching the same Virtual classroom algorithm [1].

In our opinion, this goal is Normally not possible with already available animation systems Because these are usually used for application area. Considering JIVE, an animation work focused on presentations Of data structures that try to define a common platform To create visualizations for a large range of applications scenario. [1]

An important novelty lies in the adoption of a visualization framework that can co-exist with the different architectural patterns found in different application domains of algorithmic animation.

This is possible because JIVE's approach to animation treats the algorithm and its visualization as two separate entities that interact through the transparent use of common communication channels. In addition, JIVE tells a visualization framework based on a zoomable interface that can be easily adapted to the needs that arise from different contexts..[1]

8. VERXCOMBO: AN INTERACTIVE DATA VISUALIZATION OF POPULAR LIBRARY VERSION COMBINATIONS [2]

In software development the acquisition of 3rd party software libraries is became more common. Benefits are including addition quality assurance to minimize the time and costs of re-inventing the wheel. Extensions to library hosting sites for e.g. Maven 2 Central Repository¹ for Java and RubyGems² give system administrators free access to many of open-source systems.[2]

Adopting third-party OSS libraries is common. The use of the OSS library has been validated for use in both open and industrial environments. As the library evolves, maintenance issues arise. It includes new error fixes, latest improvements, and features. A part of software maintenance system administrators is considering when and how to upgrade existing libraries to which library version is to be upgraded. [2]

Sometimes we got to see introduction of new libraries. In that case, you have to consider how this new library will work for system. Great for existing dependency environments. This is evidenced by a survey of reported concerns about incompatibilities in updates such as Heartbleedbug³. The widespread impact of ignoring updates. When it comes to documentation, the task can be very difficult. Due to personnel changes in the project, system knowledge may be insufficient, or system knowledge may be lost.[2]

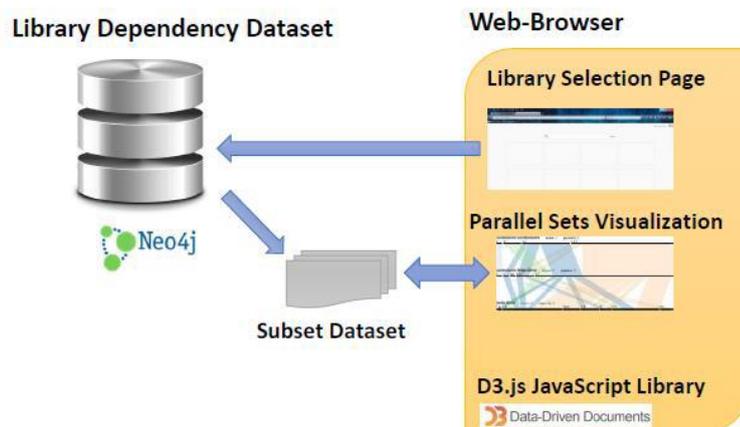


Figure 3. AN ARCHITECTURAL OVERVIEW OF VERXCOMBO [2]

9. DOC-TOOL

Software complexity with respect to code continues to increase as long as it is maintained and updated. Without the latest Doc-Tool, you can't maintain large software. It is not possible for one man to track all the components of huge software and the dependencies as well as relation between them. Many developers at different levels within an organization typically maintain and develop software code.[3]

This delays communication in between development teams as they address and review software components. This article describes Doc-Tool, a tool that tracks components present in software and their relation and presents this info to users in an easy way. This tool is intended to get both a "bird's eye view" and a "drill down view" of the software and their components. This tool used to set of JSON files and a chart database to get solutions to 2 problems in the simplest way. The first problem is it tackles and keeping big software in one place. The 2nd issue to address is the delay in communicating with different person individual at different level.[3]

The 1st phase checks the server side source code. The 2nd phase below this is where the developer infor about the components are entered. This will create a graph that looks like this: It is maintained and updated and will be used later for visualization. Alternatively, diagram nodes are software entities and components. The edges of the chart are entities. The structure of the dissertation is as follows.[3]

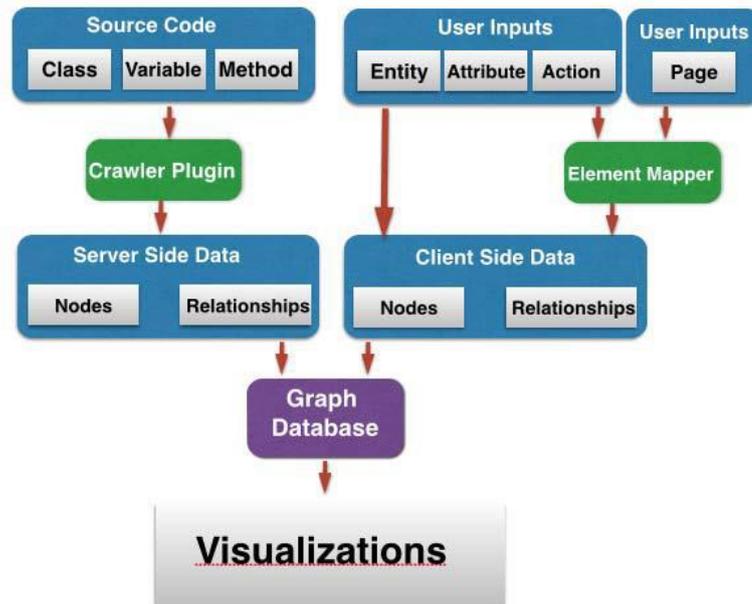


Figure 4. IMPLEMENTATION [3]

Doc-Tool is extremely easy to use and is designed to be language self-reliant. The tools gather data about the software as well as visualizes it to the user in an easy to understand and convenient way. Use a set of JSON files as well as a chart database as the backbone. The tool have of three main modules crawlers, user control , element mapper Plugin parser. The Doc-Tool architecture is shown in the figure 4. In this implementation, the tool is intended to visualize Java web applications. The crawler plugin is an eclipse plugin in fig 4. The plugin makes relationships with the nodes on the server level.

10. FUTURE SCOPE

Data visualization is no longer an art. Evolving cognitive framework, multifaceted imaging, intelligence. Data visualization explores different ranges and recognizes large amounts of complex data. As a digital alternative to visual communication, data visualization has made it easier for businesses to make decisions.

Data visualization allows you to observe your data in visually interactive ways such as patterns, comparisons, and graphs. Create trends to better understand market conditions and address issues and insights. The effects of data visualization are:

- 10.1 Present important data in a visually interactive way.
- 10.2 The pattern may be displayed at any time.
- 10.3 You can collect huge chunks of complex data in a simple visual way.
- 10.4. Prevent the risk of decision errors.
- 10.5. Helps identify key components that affect business outcomes.
- 10.6. Helps create a roadmap for future actions to be implemented.

Visualization of today's data

Data visualization can add value to information by incorporating specific use case or industry-specific themes, objects, and images.

An industry-specific application of data visualization technology to map data-driven wealth management scenarios, from the visualization of agricultural data used in prescriptive crop planning to the AR of financial services. Allows businesses and customers to generate more informed conclusion.

11. BENEFIT

11.1 The emergence of high-fidelity digital twins

A digital twin is a virtual model of a physical object / system created by reading a stream of data related to the physical object in question This enables remote monitoring of performance and health / state parameters, enabling remote analysis and evaluation of physical assets.

11.2. More powerful JavaScript visualization

The web front end is the primary interface between applications and users, as Software as a Service (SaaS) is the best way to consume today's software. Technologies such as Flash and Java have almost disappeared in this area, but JavaScript remains in the top position. Today, popular Java-Script frameworks like React, Angular.js, s Vue.js, are used to stream-line the development of complex front-end visualizations. Framework adds best reality in 3D to JavaScript based Visualization.

11.3 Verticalized data visualization offering

In the process of traditional industry digitization, data visualization is becoming more and more specialized in the needs of target groups in a particular industry. For example, visualization of shipping and shipping data allows shipowners / operators to make improve ship execution and monitor safety as well as operating condition. likewise the automotive industry uses data visualization to streamline automotive product advancement workflows.

11.4 Data visualization for mobile devices

More than half of all pages views coming from cell phones and handheld devices in 2020, website owners are encouraged to focus on optimizing and improving the user experience for mobile devices. Enterprise SaaS offerings as well as business software's platforms are using suit by taking a mobile 1st thing to data visualization. In lay mans words prioritize data visualization as well as design for optimal displays on a tiny screen. For e.g. the mobile version of Salesforce gives feed 1st design, reports, and flexible "component visualization rules" capabilities to define the data visualization components that appear on mobile devices.

12. CONCLUSION

After a brief study of data visualization, it is clear that this area is rich in potential applications in different areas, while at the same time recognizing its practical and ethical complexity. The previous chapter presented some important theoretical and practical principles to keep in mind when designing data visualizations. We also discussed and critiqued some data visualization examples and learned common pitfalls and useful tricks in the process. As we have seen, developing effective and ethical data visualizations is a complex process. This chapter describes the future of data visualization and additional resources for data visualization.

13. REFERENCES

[1] G. Cattaneo, P. Faruolo, U. F. Petrillo and G. F. Italiano, "JIVE: Java Interactive Software Visualization Environment," *2004 IEEE Symposium on Visual Languages - Human-Centric Computing*, 2004, pp. 41-43, doi: 10.1109/VLHCC.2004.34.

[2] Y. Yano, R. G. Kula, T. Ishio and K. Inoue, "VerXCombo: An Interactive Data Visualization of Popular Library Version Combinations," *2015 IEEE 23rd International Conference on Program Comprehension*, 2015, pp. 291-294, doi: 10.1109/ICPC.2015.43.

[3] A. Sadar and Vinitha Panicker J, "DocTool - a tool for visualizing software projects using graph database," *2015 Eighth International Conference on Contemporary Computing (IC3)*, 2015, pp. 439-442, doi: 10.1109/IC3.2015.7346721.

[4] *Data visualization of popular programming languages in GitHub repositories*

[5] M. Byrne, R. Catrambone, and J. T. Stasko. *Examining the effects of animation and predictions in student learning of computer algorithms. Journal of Computers & Education*, 33(7):253–278, 1999.

[6] G. Cattaneo, U. Ferraro Petrillo, and G. F. Italiano. *CATAI: Concurrent algorithms and data types animation over the internet. Journal of Visual Languages and Computing*, 13(4):391–419, August 2002.

[7] P. Crescenzi, C. Demetrescu, I. Finocchi, and R. Petreschi. *Reversible execution and visualization of programs with LEONARDO. Journal of Visual Languages and Computing*, 11(2):125–150, 2000.

[8] P. Gestwicki and B. Jayaraman. *Interactive visualization of java programs. In Proc. of the IEEE 2002 Symposia on Human Centric Computing Languages and Environments (HCC'02)*, pages 226–235, September 2002.

[9] C. M. Kehoe, J. T. Stasko, and A. Talor. *Rethinking the evaluation of algorithm animations as learning aids: an observational study. International Journal of Human Computer Studies*, 54(2):265–284, February 2001.