

Building a Knowledge Model of Cayo Santiago Rhesus Macaques: Engaging Undergraduate Students in Developing Graphical User Interfaces for an NSF Funded Research Project

Martin Q. Zhao^{*1}, Ethan R. Widener¹, George Francis², Qian Wang².

¹Department of Computer Science, Mercer University

²Department of Biomedical Sciences, Texas A&M University School of Dentistry

Zhao_mq@mercer.edu, ethan.ransom.widener@live.mercer.edu,
gfrancis@exchange.tamu.edu, qian.wang@tamu.edu

Abstract:

In this paper, we introduce an NSF funded project that aims to develop a database that integrates genetic, environmental and age-related information to study their effects on health conditions of a rhesus monkey colony at Cayo Santiago, Puerto Rico, which has been founded since 1938. In this project, we will combine the osteology data with the rich genealogy and demographic information into a searchable and computer-interoperable knowledge model accessible through user-friendly interfaces. Backed by the integrated database, this system will provide researchers and the public information from the Cayo Santiago rhesus colony and the derived skeletal collection, a powerful non-human model for datamining to study human disease. Undergraduate and graduate students from diverse communities have been incorporated into research and development activities. Related materials are used as case studies in relevant classes at Mercer University to help train these undergraduate students into problem solvers.

Keywords:

Problem-based learning, computer science education, data management for analytics, information integration, graphical user interface

Section I - Introduction:

An interdisciplinary team led by Qian Wang of Texas A&M University, funded by NSF, is developing a database that integrates genetic, environmental and age-related information to study their effects on health conditions of a rhesus monkey colony at Cayo Santiago, Puerto Rico. Founded in 1938, the Cayo Santiago colony is the source of a rare skeletal collection with associated details about each individual's sex, age, and pedigree (up to eight generations) (Sade et al., 1985; Rawlins and Kessler, 1986; Wang, 2012; Kessler and Rawlins, 2016). This skeletal collection, housed at the Caribbean Primate Research Center, has been highly useful in anthropological and biomedical studies (i.e., Wang et al, 2012; Kessler et al., 2016; Wang and Dechow, 2016; Wang et al., 2016a,b; Li et al., 2018; Guatelli-Steinberg et al., 2022).

In this project, bone dimensions, bone density, body mass, tooth eruption, and observable disease conditions of the rhesus monkey will be incorporated into the database with details on each individual's sex, birth and death dates, parentage information, and social rank (when available). Once the integrated database is available to researchers and the public, information from the Cayo Santiago rhesus colony and the derived skeletal collection will provide a powerful non-human model for datamining to study human disease, family history, development, individual experiences, and aging.

The team brings together expertise in anthropology, biology, biomedical sciences, and computer sciences. Database development incorporates undergraduate and graduate students from collaborating institutions, providing unique learning opportunities. For building the Cayo Santiago rhesus health database, we will combine the osteology data with the rich genealogy and demographic information into a searchable and computer-interoperable Knowledge Model.

"Using the program opportunity in the project to train undergraduate students at Mercer University", one of the Broad Impacts in the original proposal to NSF, has been planned since the project began. Courses relevant to the database and user interfaces development and data analytics are taught at the Computer Science Department at Mercer University through its Computer Science, Information science and Technology, and Data Science programs. Martin Q. Zhao (M.Q.Z.) has been using projects based on real world problems in his Software Engineering, Programming, Database Systems, and Data Science Applications courses. This paper will focus on how students have been involved in the design and implementation of the proposed database system, especially the front-end application codenamed as CSViewer, now in v1.0.

Section II - Database and Application System Design:

A conceptual design for the CS rhesus database and the graphical user interfaces (GUI), which support data management and analytics needs, has been discussed in Zhao et al., 2021. This section will focus on system design aspects for the database and

especially graphical user interfaces provided in the CSViewer frontend application. Implementation of the current version (v1.0) and redesign/enhancement effort for v1.1 will be discussed in the subsequent sections.

A. **Database design.** As shown in Figure 1, a relational data model (Zhao et al., 2021) has been proposed for the integrative database to manage subject genealogy information of the rhesus families maintained by the CPRC, plus health data (to be obtained in this project (including scans, measurements, and observation data). A relational design similar to a human health database (Seo et al, 2013) is used, which shows great extensibility. More tables need to be added to keep track of valid users, access control, user activities, etc. in a later stage.

B. **Data collection and preparation.** Various datasets have been collected from historical records at CPRC and bone measure activities conducted by George Francis for this project. Measuring and picture taking are still ongoing after being delayed by the pandemics and maintenance of CPRC facilities.

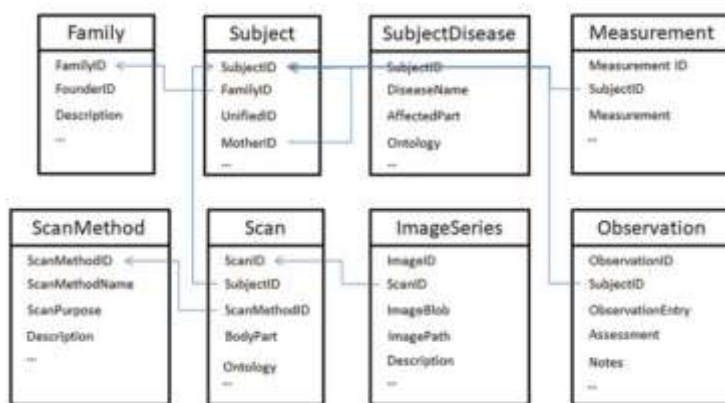


Fig. 1 - Conceptual data model for the Cayo Santiago Rhesus Health Database.

Data quality analysis have been conducted for all incoming datasets received for this project, including

- 1) Census data from 2020 and 2013: The two datasets were merged to generate a complete animal dataset with 10949 distinct animals. Another dataset is available for 77 founding mothers with only the year they were born.
- 2) Catalog data for skeletons curated at CPRC: This dataset has more than 4800 entries, including 2292 entries from the CS population.
- 3) Bone measure data: Data collection using cataloged CS specimens has been conducted since 2020. The raw data including dimensions of skull, upper and lower limbs, suture and joint conditions, as well as bone density for 1509 CS specimens (by the end of 2021) are used in the initial building of the database and application system .

- 4) A small collection of bone pictures has been included in the initial database for testing purposes. Full-scale imagery data collection is currently conducted at CPRC.

C. **Software architecture.** The proposed Knowledge Model consists of a comprehensive database managing existing and new collected data from the CS population. The CSViewer application is the frontend that provides friendly user interfaces for analysts to access the data and perform data analytics tasks.

A standard layered architecture is adopted (as shown in Figure 2), which consists of multiple layers of abstraction coalescing to facilitate the transfer and display of data from the database. For version 1.0, the CSViewer system is based on local files with more stable data structures (such as matrilineal and patrilineal family trees) extracted from the censor and catalog datasets.

Design of the middle layer uses the metaphor of Business Managers to help make the design flexible such that functions for processing different kinds of data (family, measure, image, etc.) can be supported separately to be suitable for an incremental development approach. The managers (that are key objects in functional partitions in the middle layer) are responsible for accessing data using the relevant data accessor objects (DAOs), and create value objects (VOs, such as animal or measure objects) that map to the relational data entries. Information to be displayed to the user will be prepared by the corresponding manager in the form of a page bean and feed to the GUI component that consumes it.

Some design details regarding “managers” will be given in Section III Part B, with descriptions on how Ethan Widener and his teammates in M.Q.Z.’s Software Engineering class contributed to the development of CSViewer v0.1.

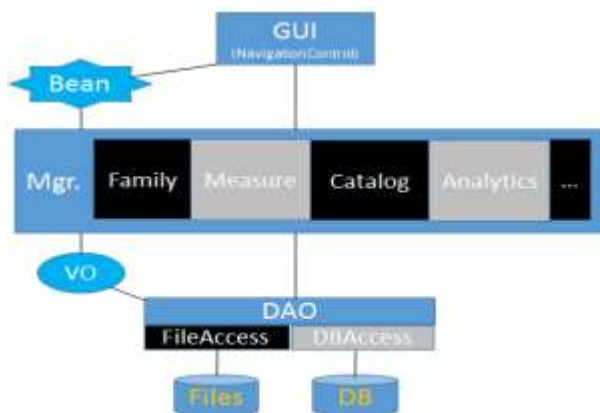


Fig. 2 – Layered architecture with Data accessibility, Business Managers, and GUI.

With such a template, each layer is responsible for a distinct set of operations and is in contact with the layers directly above and below it, ensuring a stable chain of command for efficient and clear communication of data from the back end to the front end of the system.

D. **Graphical User Interfaces.** The graphical user interface displayed upon launch of the application is the user's means of interacting with and navigating the program. The CSViewer window (Figure 3) features a myriad of menus to support various visualization and analytical tasks. The main results of the current task are displayed in the main content panel (matrilineal family trees as in the screenshot). Views for multiple tasks can be displayed as tabbed pane structures for easy navigation.



Fig. 3 – Screenshot of the Main Window of CSViewer v1.0. (with the About Box laid over)

Below the tabbed pane structure, a summary pane is used to display animal summary information on the selected subject when interacting with one of the tabbed pane views. Any additional relevant information is displayed to the right of the tabbed pane structure. This information can include measure data in table format, and/or images relating to the subject and measure data.

E. **Data Structures.** As mentioned above, this version (v1.0) reads data from local files. Animal records are already sorted in a sequential order based on mother or father bloodlines. Internally, the managers use collection classes in the standard Java collections framework (JCF) to store data, and prepare bean objects to feed to various content panes using Java Swing classes, such as JTree and JTable.

The JTree view and related data model, a Java-specific implementation of an n-tree data structure, plays an important role in rendering family trees. In this project, the founders of each matrilineal line serve as top-level nodes, while their offspring are displayed as expandable child nodes. Direct children of founders can further be

expanded to show their children, and so on to provide a complete and navigable view of the dataset and the relationships between each specimen.

The Patrilineal trees are constructed from each animal's father identity and can be traced back to the earliest male subject (with DNA screening) of a respective tree. The construction process also uses the animal information as available in the matrilineal tree input file to reduce redundancy.

Map types as available in JCF are also used to track cross-referencing patterns between animals and their catalog entries, and through which linking to measure/image data when they exist.

Section III - **Software Engineering Education**

Martin Q. Zhao has been teaching various software and database courses at Mercer University in the undergraduate Computer Science (CSC) and Information Science and Technology (IST) programs, and recently also teaching for the graduate Software and System Engineering (SSE) program.

A. ***Teaching pedagogies and student engagement.*** We believe that software and database development topics can be best learned by applying design principles and programming techniques in practice; and ideally, in real-world projects. The case studies and project topics used in M.Q.Z.'s classes evolved from workshops (Zhao and White, 2006) extended from popular textbooks (Horstmann, 2004), to externally funded research and development projects (Zhao, 2010; 2011; Zhao et al 2021; Zhao and Allen 2023).

Courses involved and teaching/learning activities:

- 1) Database implementation and essential queries have been assigned to student teams in several database and software engineering classes since the NSF grant was awarded in 2019.
- 2) The recent CSViewer release was refactored and extended from the codebase finished by a team in the latest Software Engineering I (CSC 480). The student project was based on CSV files listing animal entries sorted by father/mother in chronicle order that are prepared by M.Q.Z. and a layered software architectural template introduced in class.
- 3) Analytical result visualization (using the JFreeChart API) and related selection features are then added to form version 1.0. As a member of the CSC 480 team, Ethan Widener continues to contribute to source control and redesign for v1.1 (to be detailed in the next section), while taking research and internship courses.

Implementation, testing, and continuation among classes:

- 4) Providing DB and GUI design and codebase to student teams with specific tasks seems to be more effective. Incremental development process and source and document control can help establish a foundation securing continued feature building across many classes during subsequent academic terms. Students can contribute through class assignments, volunteer and paid commitments in

multiple (software engineering, database systems, research and internship) courses.

B. Student contributions in building the CSViewer app. Certain details about contributions made by the student team in a Software Engineering class in Fall of 2022 will be presented here. Their work helped with the integration of partially working modules into CSViewer (version 0):

- 1) Merged separate tree views for matrilineal and patrilineal lineage, using a single CSFamilyTreePanel class as provided by M.Q.Z.
- 2) Laid out four major content panes in the CSViewer main window, as described in Section I Part D.
- 3) Implemented the “measure partition” in the middle layer, which includes
 - a. A MeasureHash class using the DAO object to access measure data in a CSV file, and
 - b. Generating “measure bean” objects with attributes holding values that can be used by the JTable in the MeasureTablePanel.
- 4) Established cross-references among the CSFamilyTreePanel, the subject summary pane (beneath the main content pane), and the MeasureTablePanel.

The students were trained with in several workshops demonstrating topics like layered architectural design, GUI and JDBC programming. A team of five helped with this project, including Ethan A. Widener, who continues to work in this project with additional research and internship courses.

What impressed EAW the most is to work closely with a "client" on a complex system with evolving needs. Instead of being given with a clearly defined problem and a fixed set of steps to write a program, students need to communicate with the client to analyze what exactly is needed, and translate that model into a product. In addition, using third APIs (such as JFreeChart and Tablesaw) significantly extended the capability of the software solution, and exposed students to component-based design and development.

C. Enhancements leading to CSViewer v1.0. The integrated system (Version 0 of CSViewer) was then refactored by M.Q.Z. to give it a consistent look. Additional features have been added to make the app to take the shape that can be a useful tool for researchers managing and analyzing the valuable data collected from the CS colony.

- 1) A menu structure has been added, with a tabbed panel to hold major contents.
- 2) A ChartPanel (opened in the Analytics menu) is added to display analytical results.
- 3) Professional graphics design elements are used, as app logo and on the About box, etc.
- 4) An easy to deploy set of deliverables are zipped and sent out for user testing.

Significant features of CSViewer v1.0 will be summarized in the next section.

Section IV - Results and Redesign:

A. **Results of the Current Design.** By the time of writing this paper, CSViewer is now in v1.0.3, and just started user testing. Essential functions in this version can be accessed through its menu structure, which include:

- 2) Matrilineal and patrilineal family trees starting from 77 founders or first male ancestors with DNA tracing data. The trees are expandable, and a search function can automatically expand and show the path leading to a chosen animal in either a matrilineal or patrilineal family tree.
- 3) Cross-reference between animal family information (in the tree panel and the summary pane beneath it) and bone measure data (in table pane on the right) is established. Animal summary and available measure data can be displayed when a tree node is selected.
- 4) For selected measure data (using the Search menu), scatter plots of measure vs. animal's estimated age at death (EAAD) can be displayed through the Analytics menu (Figure 4). Analytic plots are generated using JFreeChart API and support
 - a. Tooltips when moused over and popup menus for left mouse click;
 - b. Short menu with built-in functions for right mouse click. These functions include Save chart as an image (PNG) file, Copy it to a Word file or send to print, as well as Zoom in and Zoom out.
 - c. Planned redesign using Tablesaw API (to be discussed next) will make selection and chart generation more streamlined.
- 5) While large-scale picture taking is still ongoing, linkage between the family tree and corner image pane is established. An available image (though limited in amount so far) can be displayed when the corresponding tree node is selected.
- 6) A small number of group transfer entries and images taken from earlier studies are included in this current version for proof of concept and are planned into the next release (as v1.0.4).

B. **New Design with Tablesaw.** While enhancements are still being added to the current version (v1.0), a redesign is carried out to streamline data analytic support. One important goal is to use the Tablesaw API (which was initiated in 2017 and is still in version 0) to provide a generic DataFrame-like construct that is widely used in data analytics. This addition will make the CSViewer application not just a visualization tool, but also a real toolkit for Analysts, supporting endless analytical and modeling tasks upon this comprehensive CS rhesus monkey data collection.

The Tablesaw API allows software developers to organize data into tables and manipulate those tables in ways that core Java does not support. Given the vast amount of data involved in this project, as well as the myriad of categories the data falls under which often intersect and need to be cross-referenced with one another, an underlying table structure is preferred. Tablesaw API will allow the data to be placed into large tables that can then be cut down into relevant subsets without additional overhead.

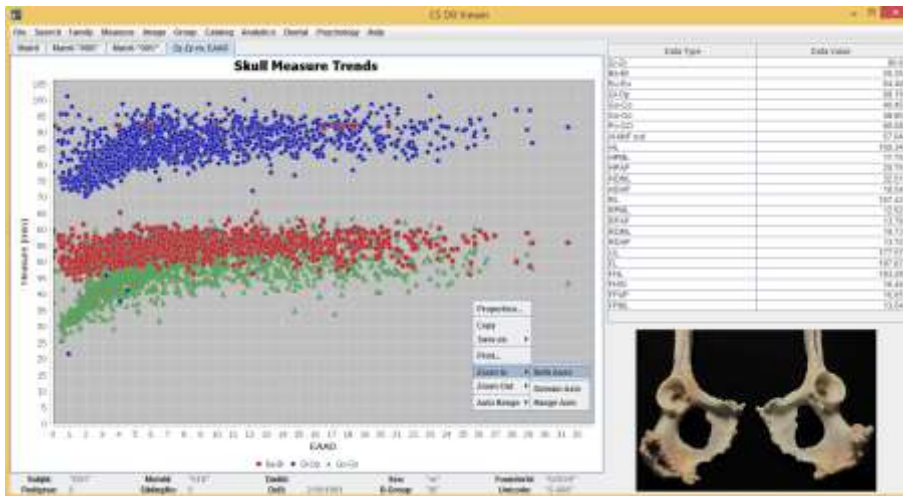


Fig. 4 – Screenshot of a chart tab for analytical results, with features like copy, save, and print.

The new design will be included in version 1.1, which aims to utilize Tablesaw and related new development (such as the XChart API) to help make the system more modular and extensible. It will pave the way for additional features in future iterations and fulfill the goal of a Knowledge Model for hypothesis-based investigations.

Section V - Conclusions and Future Work:

An initial version of the CSViewer for Analysts (v1.0) has been developed and sent for user testing. This user-friendly application helps make it easy to access and analyze data from the Cayo Santiago rhesus colony, one of the most useful primate sources in biomedical and anthropological research. Essential visualization features are now available for family trees based on historical records and recent census data, as well as dimension measures collected from the derived skeletons.

Design and development topics of this real world application have been used in related courses (including programming, database, software engineering, and data science) offered by the Computer Science Department at Mercer University. Undergraduate students in those courses have been involved through lectures, hands-on exercises, and contributing to building parts of the system. Teams of students worked to analyze the needs of the client, design a solution using third party APIs, and translate their models into working software.

Future work: With more existing datasets (such as dental data) and new data being collected (such as imagery and pathological data), we will continue to build both the backend database system and add more features to CSViewer. Efforts will be made to

make cross-referencing data from different sources consistent and to provide more data analysis and modeling support to establish a unique Knowledge Model for hypothesis-based investigations.

Acknowledgments:

The CPRC Skeletal Collection has been supported by National Institutes of Health NIH contracts NIH 5P40OD012217. This project is supported by NSF grants to M.Q.Z. and Q.W. (#1926402, #1926601). Melween I. Martinez Rodriguez (Current CPRC Director), Bonn V. Liong Aure, Terry B. Kensler, Elizabeth Maldonado, Giselle Caraballo Cruz, and other CPRC staff members for their support and help.

Andres Boulosa, Adrian Faircloth, Jesus Rijo, and Zaina Khutliwala worked on the team together with E.R.W. that contributed to CSViewer 1.0. Rajwol Chapagain helped in system testing and preparing user instructions. Robert Allen, Jesse Sowell and the students at the Mercer University's Computer Science Department provided their supports and contributions. Special thanks go to Pegasus Vertex, Inc. in Houston, TX for providing graphics design used in the user interfaces.

Author Contributions:

MQZ: Conceptualization (co-lead); formal analysis (support); investigation (equal); methodology (lead); project administration (co-lead); visualization (lead); writing—original draft (co-lead); writing—review and editing (co-lead).

ERW (Undergraduate Student): Data curation (equal); formal analysis (support); visualization (support); writing—review and editing (support).

GF (Graduate Student): Data curation (equal); formal analysis (support); visualization (support); writing—review and editing (support).

QW: Conceptualization (lead); data curation (equal); investigation (equal); methodology (co-lead); project administration (co-lead); visualization (co-lead); writing—original draft (equal); writing—review and editing (co-lead).

References:

C. Horstmann, "Object-Oriented Design and Patterns," 2nd Edition, Wiley, 2005.

D. Guatelli-Steinberg, T. Guerrieri, T. Kensler, E. Maldonado, G. Francis, L. Kohn, M. Zhao, J. Turnquist, Q. Wang, "Male Cayo Santiago rhesus macaques (*Macaca mulatta*) tend to have greater molar wear than females at comparable ages: Exploring two possible reasons why," 2022, <https://doi.org/10.1002/ajpa.24519>.

D. Guatelli-Steinberg, T. Guerrieri, T. Kensler, E. Maldonado, G. Francis, L. Kohn, M. Zhao, Q. Wang, "Male Cayo Santiago rhesus macaques (*Macaca mulatta*) tend to have greater molar wear than females at comparable ages: exploring two possible reasons why," *American Journal of Biological Anthropology*, vol. 178, pp. 437-447, 2022.

Commented [WQ1]: Order of references?

- D. Sade, B. Chepko-Sade, J. Schneider, S. Roberts, J. Richtsmeier, "Basic demographic observations on free-ranging rhesus monkeys," New Haven, Human Relations Area Files Press: 1985.
- D. Seo, S. Lee, S. Lee, H. Jung, W.K. Sung. "Construction of Korean spine database with degenerative spinal diseases for realizing e-spine, KSII," The 8th Asian Pacific international conference on information science and technology (APIC-IST), Jeju, Republic of Korea, 2013.
- H. Li, W. Luo, A. Feng, M. Tang, T. Kensler, E. Maldonado, O. Gonzalez, M. Kessler, P. Dechow, J. Ebersole, Q. Wang, "The odontogenic abscess in Rhesus macaques (*Macaca mulatta*) from Cayo Santiago," *American Journal of Physical Anthropology*, vol. 167, pp. 441-457, 2018.
- M. Kessler, R. Rawlins, "A 75-year pictorial history of the Cayo Santiago rhesus monkey colony," *American Journal of Primatology*, vol. 78, pp. 6-43, 2016.
- M. Kessler, Q. Wang, A. Cerroni, M. Grynypas, O. Velez, R. Rawlins, K. Ethan, J. Wimsatt, T. Kensler, K. Pritzker, "Long-term effects of castration on the skeleton of male rhesus monkeys (*Macaca mulatta*)," *American Journal of Primatology*. vol. 78, pp. 152-166, 2016.
- M. Zhao, "Knowledge Models for SA Applications and User Interface Development for the SITA System, Final Report," ARFL/RI, Rome, NY, 2011.
- M. Zhao, T. Kensler, D. Guatelli-Steinberg, L. Kohn, G. Francis, and Q. Wang, "Reproduction of Cayo Santiago Rhesus Colony based on the Patrilineal Family Trees: The Missing Patterns," Poster presentation at AABA '23, Reno, NV, 2023.
- M. Zhao, M. Maldonado, T. Kensler, L. Kohn, D. Guatelli-Steinburg, Q. Wang, "Conceptual Design and Prototyping for a Primate Health History Model," *Advances in Computer Vision and Computational Biology*, Springer: New York, p.511-522, 2021.
- M. Zhao, and L. White, "Engaging Software Engineering Students Using a Series of OOAD Workshops," *Proceeding of ASEE*, Chicago, IL, 2006.
- Q. Wang, (Editor), "Bones, Genetics, and Behavior of Rhesus Macaques: *Macaca mulatta* of Cayo Santiago and Beyond," Springer: New York, 2012.
- Q. Wang, "Dental maturity and the ontogeny of sex-based differences in the dentofacial complex of rhesus macaques from Cayo Santiago," Springer: New York, pp. 177-194, 2012.
- Q. Wang, P. Dechow, "Divided zygomatic bone in primates with implications of skull morphology and biomechanics," *Anatomical Record*, vol. 299, pp. 1801-1829, 2016.
- Q. Wang, M. Kessler, T. Kensler, P. Dechow, "The mandibles of castrated male rhesus macaques (*Macaca mulatta*): The effects of orchidectomy on bone and teeth," *American Journal of Physical Anthropology*, vol. 159, pp. 31-51, 2016.

Q. Wang, J. Turnquist, M. Kessler, "Free-ranging Cayo Santiago rhesus monkeys (*Macaca mulatta*): III. Dental eruption Patterns and dental pathology," *American Journal of Primatology*, vol. 78, pp. 127-142, 2016.

R. Rawlins, M. Kessler, (Editors), "The Cayo Santiago Macaques: History, Behavior and Biology," State University of New York Press: Albany 1986.

Tablesaw: A Platform for Data Science in Java, <https://github.com/jtablesaw/tablesaw>, [Retrieved: Feb 16, 2023].

The JFreeChart Project, <https://www.jfree.org/jfreechart/>, [Retrieved: Feb 16, 2023].

The XChart API, <https://knowm.org/open-source/xchart/>, [Retrieved: Feb 28, 2023].