SAGE3 - the Smart Amplified Group Environment

1st Roderick Tabalba
University of Hawaii at Manoa
Honolulu, HI, USA
tabalbar@hawaii.edu

2nd Nurit Kirshenbaum
University of Hawaii at Manoa
Honolulu, HI, USA
nuritk@hawaii.edu

3rd Jesse Harden
Virginia Tech
Blacksburg, Virginia, USA
jessemh@vt.edu

4th Michael Rogers
University of Hawaii at Manoa
Honolulu, HI, USA
mlr2010@hawaii.edu

5th Arthur Nishimoto
University of Illinois, Chicago
Chicago, IL, USA
anishi2@uic.edu

6th Elizabeth Christman
Virginia Tech
Blacksburg, Virginia, USA
elizabethc99@vt.edu

7th Andy Yu
University of Hawaii at Manoa
Honolulu, HI, USA
andyyu@hawaii.edu

8th Ryan Theriot
University of Hawaii at Manoa
Honolulu, HI, USA
rtheriot@hawaii.edu

9th Lance Long
University of Illinois, Chicago
Chicago, IL, USA
llong4@uic.edu

10th Luc Renambot
University of Illinois, Chicago
Chicago, IL, USA
renambot@uic.edu

11th Mahdi Belcaid
University of Hawaii at Manoa
Honolulu, HI, USA
mahdi@hawaii.edu

12th Chris North
Virginia Tech
Blacksburg, Virginia, USA
north@vt.edu

13th Andrew Johnson
University of Illinois, Chicago
Chicago, IL, USA
ajohnson@uic.edu

14th Jason Leigh
University of Hawaii at Manoa
Honolulu, HI, USA
leighj@hawaii.edu

***Abstract*—SAGE3 is software to augment the cyberinfrastructure-enhanced research and education enterprise by supporting data-intensive collaboration across a wide range of display devices from high-resolution display walls to laptops. This paper provides insight into SAGE3's implementation, which significantly improves on prior generations of SAGE by leveraging emerging advancements in Web technologies and Artificial Intelligence. We also provide an overview of new usage patterns that we observed with SAGE3.**

***Index Terms*—SAGE3, Collaboration, AI, Display Wall**

I. Introduction

The Scalable Adaptive Graphics Environment (SAGE) and the Scalable Amplified Group Environment (SAGE2) are software [1] that enable scientists, researchers and students to collaborate with their colleagues and their data in front of scalable tiled display walls, sharing information and digital media data, particularly large-scale visualizations and animations, to make discoveries, reach agreement, and come to decisions with greater speed, accuracy, comprehensiveness and confidence [2]. The SAGE approach to collaboration enables multiple users to simultaneously drag and drop information - e.g., numeric data files, images, movies, PDFs, and web pages - from their laptops onto a shared display walls where it can be interactively juxtaposed with other related information. SAGE's collaboration features enable multiple users to point, interact, and compare features, while also allowing simultaneous screen sharing from active applications.

The SAGE3 project is funded in part by the National Science Foundation awards 2004014, 2003800, 2003387, 2149133.

This approach has successfully supported eScience collaborations for over 5000 users across 600 institutions equipped with tiled display walls, encompassing 21 disciplines including Archaeology, Architecture, Art, Atmospheric Science, Biology, Chemistry, Civil Engineering, Communications, Computer Science, Education, Geoscience, Health, Library Science, Mathematics, Medical, Meteorology, Network Engineering, Neuroscience, Physics, Psychology, and Statistics.

SAGE3 (sage3.sagecommons.org) represents our comprehensive redesign and re-implementation of its predecessors, to endeavor to endow its users with two fundamental and intricate functionalities: 1. the seamless exchange of substantial varieties and quantities of content across both expansive display walls *and* collaborative laptops/computers; and 2. the augmentation of data-intensive collaboration leveraging artificial intelligence (AI).

This paper provides a brief overview of the SAGE3 platform, from its user-interface to its system architecture. First, we describe SAGE3's end-user motivated capabilities. Then, we describe SAGE3's system and software architecture design and implementation that are needed to support the capabilities. Lastly, we describe SAGE3's usage patterns and a use case that showcases SAGE3's capabilities.

II. SAGE3 End-User Capabilities

Based on our 20 years of experience with SAGE1 and 2, scientists need a variety of software tools to support their work, namely:

- *Collaborative file sharing* - To share data file types in the form of text and data tables, to three-dimensional imaging and videos. Example tools: Google Drive, Microsoft One Drive, Box and Dropbox.

- *Collaborative brainstorming* - To communicate ideas and organize thoughts among one or more people. Example tools: Mindmap, Google Jamboard, and infinite canvas boards such as Miro, Freeform, Figma, Mezzanine, Thinkhub.
- *Presentation* - To share ongoing or finished data products. Example tools: Google Slides, Prezi and the many infinite canvas tools mentioned above, as well as communication tools like Zoom.
- *Performing computation* - To conduct data transformation and analysis. Example tools: Jupyter, Google Colab, Tableau, Spreadsheets.
- *Spatial Computation* - Most notably, recent workflows (e.g., Einblick.ai and CoCalc) opt to overcome the limitations of a linear computational notebook [3], [4] by developing computational environments that use some of the spatial features of board tools.

With SAGE3, we set out to create an **all-in-one toolset** that incorporates rich media content all in a desktop-like setting that harnesses the power of meaningful spatial placement of information [5]–[8]. Any web application can run as an application within SAGE3, bringing users together in one unified work environment, enabling them to harness and amplify the capabilities of existing scientific gateways and portals [9]. SAGE3 surpasses its competitors by the sheer flexibility and extensibility of the system, its design for facilitating collaboration among distributed groups with different display devices and sizes (harking on SAGE1 and SAGE2 historical support for large, wall-sized displays), and an AI core called the Foresight Engine with countless potential applications [10]–[12],.

Like many modern collaboration tools, SAGE3 supports users placing content on an infinite canvas called a *board*. Boards are organized within *rooms*, and rooms and boards can be password protected by the user that creates them. At the room level, SAGE3 manages assets (files uploaded or created on any board in the room) and kernels (both public and private), giving every board in a given room access to related content.

Users can use any number of SAGE3 applications within boards, most commonly viewers for multimedia files (images, videos, pdfs, etc..), web viewers for a fully functional browser within SAGE3, screen shares for as many as supported by one's network bandwith, and sticky notes. Many of these start by simply dragging content onto the board or clicking on them from the application menu. Every opened application can be positioned and resized anywhere on the board. Other users on the board can also interact and manipulate application windows; for example, they may resize each other's screen share or edit each other's notes. Other users can also download assets from the board.

SAGE3 incorporates AI capabilities, including a spatialized computational notebook (SAGECell), an AI-enhanced chart drawing application (ChartSAGE) inspired by [13], and an AI backend for applying algorithms (such as sentiment analysis, summarization, or translation) to document groups on the boards.

A board's user interface enables direct manipulation of application windows and a set of control widgets, similar to toolbars, that can be moved about and called/dismissed as needed. These widgets let users: launch applications, track or follow other users on the board, access the assets manager, launch plugins (a form of additional external applications), add annotations, and navigate across the board. The full list of SAGE3 features is available on the SAGE3 website.

III. SAGE3 Architecture

The SAGE3 architecture, shown in Figure 1, contains the following core components: Homebase (the central server), Webapp (web client), Foresight (AI functionality), JupyterLab (to run Python kernels), and Redis (in-memory database).

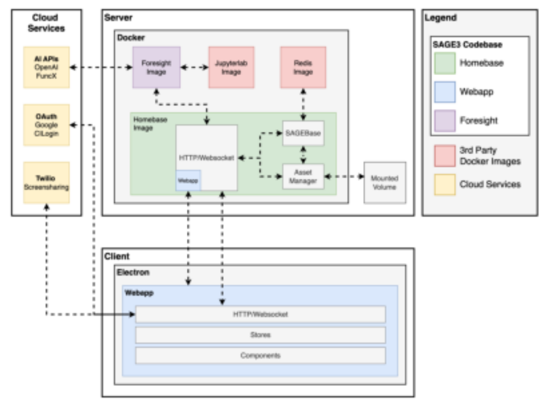

Fig. 1. SAGE3 Architecture with server, clients and services

Homebase is a backend web server that manages SAGE3's state information in SAGEBase, an interface to Redis- a fast and persistent in-memory database that enables us to achieve near real-time synchronization between clients. Homebase provides endpoints that enable users to update various SAGE3 states through a REST API or over websocket messages. Additionally, Homebase allows users to upload and download files and serves the Webapp to the clients.

The Webapp is a graphical user interface that allows users to manipulate the current state of SAGE3. It communicates with Homebase to keep the user's local SAGE3 state synchronized with the server's state. The Webapp is contained within an Electron desktop application.

Foresight is a backend service that enables users to execute Python code within SAGE3, as predetermined functions (Smartbit) or user code in a SAGECell. SAGECell allows users to write Python code concurrently. The code is executed on the backend within dynamically created Python kernels inside JupyterLab. Additionally, Foresight enables users to leverage AI functionality and tools within SAGE3.

A. Backend

1) Homebase: Homebase is a NodeJs and Express web server that serves as the routing layer, controller layer, and

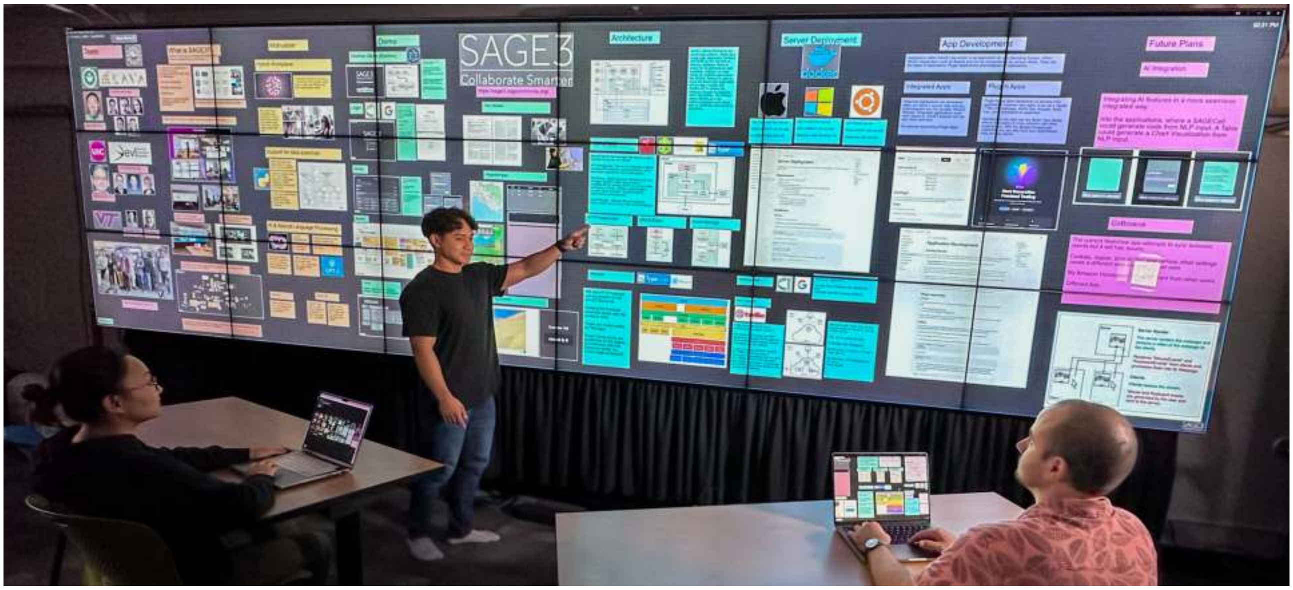

Fig. 2. SAGE3 training presentation given to a European Union funded project (landsage.info) on landslide monitoring, management, and decision making.

data access layer. It provides a persistent single source of truth, with a comprehensive set of tools and features to enable users to work together on projects, communicate in real time, and manage tasks and documents efficiently. Homebase handles asset uploads and serves the frontend web application to connected clients. Homebase exposes REST APIs to data collections, handles authentication and access, manages session information, and processes data uploads.

The asset manager is responsible for handling file uploads to Homebase. Homebase stores uploaded assets on the server's local storage (cloud storage options are available too) and maintains references to them within SAGEBase. Clients have the ability to upload a variety of file formats to SAGE3, such as images, videos, PDFs, text files and others. To optimize performance, the asset manager pre-processes PDF files before exposing them to clients.

Homebase employs SAGEBase's authentication module to handle user logins and session information. To manage different login strategies, SAGEBase utilizes PassportJS. Currently, SAGE3 supports Google OAuth and CILogon, with additional strategies easily integrated.

2) Foresight: Foresight is SAGE3's AI processing module written in Python. It handles AI requests from clients via Homebase. Foresight is also a proxy to the JupyterLab backend, providing Python kernels for SAGE3. It creates, deletes and maintains the kernels on behalf of SAGE3 users.

3) Redis: Redis is an in-memory data-structure store, used as a database, cache, and message broker. It supports a wide variety of data structures, including strings, hashes, lists, sets, sorted sets, and JSON. Homebase leverages SAGEBase as a data-access communication layer to communicate with Redis. SAGEBase also stores data within Redis as a collection and document style database.

4) JupyterLab: JupyterLab is the dominant web-based interactive development environment for users working with notebooks, code, and data. It is a flexible platform for scientific computing, data analysis, and machine learning in Python and other languages. In the context of SAGE3, JupyterLab provides Python kernels that enable clients to run Python code directly within SAGE3's boards. This is useful for data analysis and machine learning workflows, as clients can manipulate data and run models directly within the SAGE3 interface.

B. Frontend

The SAGE3 frontend consists of two components: a React web application called Webapp, and an Electron desktop application.

1) Webapp: Webapp is written in Typescript and is a React web application. It communicates with Homebase to synchronize a client's local state with the server state. To achieve this, the application uses local data stores (implemented with Zustand) to maintain a local copy of the state, which is then passed to various React components to render the graphical user interface. Components interact only with the stores, which maintain synchronization with Homebase. The stores act as the intermediary between the components and the server, ensuring that the state of the application remains consistent across all parts of the system. This technique is applied to users, rooms, boards, assets, and SAGE3 applications.

2) Electron: Clients use the SAGE3 Electron desktop application to access Webapp by connecting to Homebase. Electron is an open-source framework that enables developers to build cross-platform desktop applications using web technologies. While Electron integrates the Chromium browser and Nodejs in a single platform, it gives the developers access to custom and native APIs over a standard browser and makes it easy for a small team to deploy one application to many environments and architectures (Windows, MacOS, x64, arm64, etc).

IV. Applications

SAGE3 provides a variety of applications to enable users to begin collaborating immediately. When users upload files of various formats to a board, SAGE3 provides support applications to display those uploaded files. PDFs, images, videos, text, CSV, and geojson are currently supported, each with their respective "Viewer," such as PDFViewer, ImageViewer, VideoViewer, and CSVViewer. In addition, SAGE3 provides various collaboration applications such as Notepad, Screenshare, and Web Browser.

Applications within SAGE3 are similar to applications within a desktop environment, in that they have their own encapsulated internal logic and are displayed within a movable window. However, unlike an operating system environment where only one user can interact with an application at a time, the SAGE3 environment enables multiple users to interact with the same application simultaneously. As a result, the state of an application must be synchronized in real-time across multiple clients.

A. Plugin System

We designed a Plugin system that enables users to extend the functionality of SAGE3 by installing third-party plugins. A SAGE3 Plugin is any web application that can run inside an 'iframe' component with its own HTML/JS/CSS or even WebAssembly. An iframe is an isolated context of execution with its own sandboxing and content security policies, providing a relatively safe way to execute unknown code with unknown dependencies. To leverage SAGE3's features and API, we provide a JavaScript communication NPM library (@sage3/sageplugin) to provide state synchronization using a data communication channel between the Plugin's iframe and the SAGE3 runtime. A user now can package a web application into an compressed archive (.zip file) and upload it to the SAGE3 server at runtime, after specifying a little metadata (plugin name and description).

V. Artificial Intelligence - SAGECell

SAGE3 integrates data science and AI capabilities, enhancing previous SAGE iterations. It provides familiar tools like computational notebooks while leveraging spatialized information (SAGECells). In today's Data Science era, Jupyter Notebook is widely used, but our user study found that arranging cells spatially improves productivity [3]. This led to the development of SAGECell, a code cell that can be freely positioned on SAGE boards. Users interact with SAGECell by typing and editing the Code Input Box, and the Output Box displays text and rich media-types like PNG, JPG, SVG, HTML, with interaction for Plotly and Vega charts.

VI. New Usage Patterns

Previously, when observing how our user community worked with SAGE2, we discovered a number of signature usage patterns. These include: conceptualization (brainstorming), data collection and wrangling, data analysis and visualization, knowledge crystallization, and presentation [2]. SAGE3 enhanced these existing patterns, but has also resulted in new usage patterns.

A. Enhancing Existing Usage Patterns

1) Across Space: The classical SAGE and SAGE2 collaboration mode was a group of users all sitting in front of a display wall and using their laptops to manipulate documents on the wall. They could only see the content on the display wall, not on their laptops. With SAGE3, however, users can see the same content on every device, and as a result, have exhibited new modes of collaboration: (1) remote users working on laptops/desktops can collaborate with users in front of display walls; (2) remote users on laptops/desktops can collaborate with other remote users also using laptops/desktops, and (3) locally situated users on laptops/desktops can collaborate with other locally situated users on laptops/desktops.

2) Across Time: In addition to accommodating use scenarios across space, SAGE3 can accommodate Computer Supported Collaborative Work (CSCW) across time. While most work we have seen falls under synchronous collaboration, the permanence of each SAGE3 board, means that we also see asynchronous work patterns. Group members can load or update content in between work sessions in order to prepare, communicate, and add referents on the board.

3) Across Boards: While SAGE2's servers could only support a single group meeting at a time, SAGE3 can support multiple meetings concurrently, as long as there is sufficient computational capacity to support those meetings.

B. Creating New Usage Patterns

1) 2D programming: SAGE3 enables data collection, wrangling and analysis tasks to be performed using two-dimensionally-arranged code cells (SAGECells) that can be collaboratively viewed and edited. This enables users to simultaneously contribute code to different parts of the data analysis pipeline at the same time.

2) Viewport Guide: SAGE2's workspace was limited to the resolution of the physical display wall. SAGE3 provides infinitely sized screen spaces (rooms and boards) as well as an infinite number of rooms/boards where users can spread out their content. Often, when a display wall is filled with content, users will search the entire screen to find more open space in which to work. Alternatively, they can create a whole new room/board to work in.

3) Virtual Reality: The SAGE3 team is frequently asked whether SAGE can support Virtual Reality (VR). We conducted research and prototyping in this area [14], and SAGE3's boards can indeed be viewed in VR headsets, such as the Meta Quest. This enables users to view boards as if they are standing in front of a large physical display wall, even though in actuality they may be in a small confined space, like a hotel room, or sitting in an airplane. They can also be surrounded by a multitude of boards at the same time, each of varying physical size - something difficult to achieve in the real world. However, this is not necessarily the only and best use of VR. Our long-term goal is to support interactions that can better take advantage of 3D environments, such as viewing inherently three-dimensional datasets.

C. SAGE3 Use Case

While SAGE3 has many use cases, here we will briefly showcase the **Hawaii Mesonet application**, a SAGE3 application meant to provide a way for farmers, ranchers, and climate scientists to create their own personal weather-dashboards without knowing how to code. The Hawaii Mesonet application retrieves near real-time climate data from the Hawaii Climate Data Portal [15] and provides a simple way to generate multiple charts and widgets for selected Mesonet stations. By using this application to create specialized dashboards, farmers and ranchers are capable of conducting strategic planning - make logistical decisions for crops, management of natural resources, and cattle that minimize their losses for any foreseen shifts in Hawaii's weather. For climate scientists, the Hawaii Mesonet application provides a way to create dashboards to study Hawaii's diverse weather conditions; With Hawaii being one of the most isolated group of islands in the world and possessing a wide range of climate zones, Hawaii can provide unique climate data for scientists to study climate change. Both user groups can enhance their dashboards using other SAGE3 applications, for example, using webviews, users can enrich their dashboards with additional sources of climate data like Windy.com. This example use case illustrates how SAGE3 can serve as a substrate for diverse web portals. Additional use cases can be found in our previous paper [16].

VII. Conclusion

SAGE3 is software cyberinfrastructure that empowers users to organize information and applications in a spatial layout, collaboratively work with both the data and one another, synchronously or asynchronously, regardless of whether they are in the same location or remote. Additionally, SAGE3 integrates with and democratizes AI, thereby accelerating the scientific discovery process. Commercial systems are emerging that attempt to address some of SAGE3's capabilities, but they are all closed source and do not reveal their approach. This paper contributes to the state of practice by shedding light on how one might go about developing collaborative cyberinfrastructure systems such as SAGE3. Specifically, we describe the design and development of a software architecture and platform that can support current and future highly interactive applications that need to constantly maintain their state while being used collaboratively by multiple users. SAGE3 can be downloaded from sage3.sagecommons.org and the source code has recently been made public at https://github.com/SAGE-3/next.

References

[1] B. Jeong, L. Renambot, R. Jagodic, R. Singh, J. Aguilera, A. Johnson, and J. Leigh, "High-performance dynamic graphics streaming for scalable adaptive graphics environment," in *Proceedings of the 2006 ACM/IEEE Conference on Supercomputing*, ser. SC '06. New York, NY, USA: Association for Computing Machinery, 2006, p. 108–es. [Online]. Available: https://doi.org/10.1145/1188455.1188568

[2] J. Leigh, D. Kobayashi, N. Kirshenbaum, T. Wooton, A. Gonzalez, L. Renambot, A. Johnson, M. Brown, A. Burks, K. Bharadwaj *et al.*, "Usage patterns of wideband display environments in e-science research, development and training," in *2019 15th International Conference on eScience (eScience)*, IEEE. San Diego, California, USA: IEEE, 2019, pp. 301–310.

[3] J. Harden, E. Christman, N. Kirshenbaum, J. Wenskovitch, J. Leigh, and C. North, "Exploring Organization of Computational Notebook Cells in 2D Space," in *2022 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. Rome, Italy: IEEE, 2022, pp. 1–6.

[4] J. Harden, E. Christman, N. Kirshenbaum, M. Belcaid, J. Leigh, and C. North, "There is no reason anybody should be using 1D anymore: Design and Evaluation of 2D Jupyter Notebooks," in *Graphics Interface 2023-second deadline*, 2023.

[5] D. S. Tan, D. Gergle, P. Scupelli, and R. Pausch, "With similar visual angles, larger displays improve spatial performance," in *Proceedings of the 2003 Conference on Human Factors in Computing Systems, CHI 2003, April 5-10, 2003*, G. Cockton and P. Korhonen, Eds. Ft. Lauderdale, Florida, USA: ACM, 2003, pp. 217–224. [Online]. Available: https://doi.org/10.1145/642611.642650

[6] R. Ball and C. North, "Effects of tiled high-resolution display on basic visualization and navigation tasks," in *Extended Abstracts Proceedings of the 2005 Conference on Human Factors in Computing Systems, CHI 2005, April 2-7, 2005*, G. C. van der Veer and C. Gale, Eds. Portland, Oregon, USA: ACM, 2005, pp. 1196–1199. [Online]. Available: https://doi.org/10.1145/1056808.1056875

[7] C. Andrews, A. Endert, and C. North, "Space to think: large high-resolution displays for sensemaking," in *Proceedings of the 28th International Conference on Human Factors in Computing Systems, CHI 2010, April 10-15, 2010*, E. D. Mynatt, D. Schoner, G. Fitzpatrick, S. E. Hudson, W. K. Edwards, and T. Rodden, Eds. Atlanta, Georgia, USA: ACM, 2010, pp. 55–64. [Online]. Available: https://doi.org/10.1145/1753326.1753336

[8] N. Kirshenbaum, K. Davidson, J. Harden, C. North, D. Kobayashi, R. Theriot, R. S. Tabalba Jr, M. L. Rogers, M. Belcaid, A. T. Burks *et al.*, "Traces of time through space: Advantages of creating complex canvases in collaborative meetings," *Proceedings of the ACM on Human-Computer Interaction*, vol. 5, no. ISS, pp. 1–20, 2021.

[9] SGCI. (2023) Science gateway community institute. SGX3. [Online]. Available: https://sciencegateways.org

[10] T. Cook, "Apple launches freeform: A powerful new app designed for creative collaboration," Aug 2022. [Online]. Available: https://www.apple.com/newsroom/2022/12/apple-launches-freeform-a-powerful-new-app-designed-for-creative-collaboration/

[11] A. Khusid, "The visual collaboration platform for every team: Miro," 2011. [Online]. Available: https://miro.com/

[12] T. Krasa, "Solve any data problem with just one sentence," 2019. [Online]. Available: https://www.einblick.ai/

[13] R. Tabalba, N. Kirshenbaum, J. Leigh, A. Bhatacharya, V. Grosso, A. Johnson, B. Di Eugenio, and M. Zellner, "An investigation into an always listening interface to support data exploration," in *2023 28th International Conference on Intelligent User Interfaces (IUI '23)*, ACM. New York, NY, USA: ACM, 2023.

[14] D. Kobayashi, N. Kirshenbaum, R. S. Tabalba, R. Theriot, and J. Leigh, "Translating the benefits of wide-band display environments into an xr space," in *Symposium on Spatial User Interaction*. New York, NY, USA: ACM, 2021, pp. 1–11.

[15] T. Giambelluca. (2023) Hawai'i mesonet. Hawai'i Climate Data Portal. [Online]. Available: https://www.hawaii.edu/climate-data-portal/hawaii-mesonet

[16] J. Leigh, D. Kobayashi, N. Kirshenbaum, T. Wooton, A. Gonzalez, L. Renambot, A. Johnson, M. Brown, A. Burks, K. Bharadwaj, A. Nishimoto, L. Long, J. Haga, J. Burns, F. Cristobal, J. McLean, R. Pelayo, and M. Belcaid, "Usage patterns of wideband display environments in e-science research, development and training," in *2019 15th International Conference on eScience (eScience)*, 2019, pp. 301–310.