

Original software publication



BabiaXR: Facilitating experiments about XR data visualization

David Moreno-Lumbreras^{*}, Jesus M. Gonzalez-Barahona, Gregorio Robles

Universidad Rey Juan Carlos, Madrid, Spain

ARTICLE INFO

Keywords:

Extended reality
Empirical experiment
Data visualization
Web browsers
Virtual reality
Augmented reality

ABSTRACT

BabiaXR is a toolset for conducting experiments about 3D data visualizations in extended reality (XR) in the browser. *BabiaXR* provides both components for building complex data visualizations, and for easily transforming them into scenes suitable for running experiments with subjects. For data visualization, it provides components to retrieve, filter, select, and visualize data. To facilitate empirical experiments with human subjects, it provides components for showing information to subjects, controlling their interaction with data, and recording their reactions. This enables the easy transformation of a certain data visualization scene into an experiment directly usable by subjects. *BabiaXR* is extensible, based on the *A-Frame* JavaScript framework for XR. As such, it is easily composable with other *A-Frame* components, and complex data visualization scenes can be created by using only HTML constructs. *BabiaXR* can be used in any XR device supporting *WebXR*, and with limited capabilities also in desktop and mobile devices.

Code metadata

Current code version	1.2.6
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-23-00039
Code Ocean compute capsule	N/A
Legal Code License	GPL-3.0
Code versioning system used	git
Software code languages, tools, and services used	JavaScript
Compilation requirements, operating environments & dependencies	WebXR-compliant web browser (most modern web browsers, including those in XR devices)
If available Link to developer documentation/manual	https://babiaxr.gitlab.io/
Support email for questions	david.morenolu@urjc.es

1. Motivation and significance

In this paper, we introduce *BabiaXR*, a toolset for 3D data visualization in XR (virtual and augmented reality). *BabiaXR* combines data analytics, empirical XR experiments, and immersive XR environments to create an extended reality system for data analysis. It supports diverse analysis methods from scientific literature and offers experiment planning and execution features. Leveraging *WebXR*'s multiplatform capabilities, *BabiaXR* works across different platforms and supports various experiment modes, including on-site, semi-unattended, and remote experiments. Visualization has a rich history, dating back to ancient maps and star charts in 6200 BC and 134 BC, respectively [1]. Recent years have seen increased interest in data visualization due to data accessibility and analytical tools [2,3]. Our work contributes

to empirical XR data visualization, aiming to provide a universal approach for creating and validating XR data visualizations in the InfoVis domain.

XR technology offers a new dimension for data presentation, allowing data to be represented as 3D objects, leveraging our brain's 3D perception advantages. However, there is a need for more research on the advantages, effectiveness of visualizations, and design principles for XR data visualizations. Immersive analytics, utilizing 3D visualizations and immersive technologies, can revolutionize data-driven decision-making [4]. Yet, challenges like intuitive interfaces and privacy concerns must be addressed in this promising research area.

Existing research explores the use of virtual reality (VR) for scientific data visualization [5,6]. Applications range from seismic data,

^{*} Corresponding author.

E-mail address: david.morenolu@urjc.es (David Moreno-Lumbreras).

protein docking, astronomy, to healthcare data [7–10]. VR enables interaction with complex phenomena and their visualizations, as Bryson emphasized [11]. However, caution surrounds the integration of 3D (and XR) visualizations in literature. Munzner cautions against unjustified 3D visualizations [12], and Few advises against “3D displays of quantitative data” [13]. Nonetheless, researchers advocate for the benefits of 3D and XR in data visualization. Batch et al. focus on presence tasks [14], Jacob on embodiment tasks [15], Rosenbaum on involvement tasks [16], and García-Hernández applies it in aerospace [17]. Data visualization and validation often intertwine, involving empirical experiments to assess visualization effectiveness. In XR, conducting such experiments remains an ongoing exploration, with challenges in planning and execution. Challenges encompass defining tasks, participant interaction methods, and validating responses. Creating a data validation environment in XR remains a known academic challenge, with diverse solutions for analyses, environment comparisons, and data validation. Previous studies have conducted XR experiments, such as Raja et al.’s pilot study using CAVE [18], Millais’ 3D scatter plots and parallel coordinate plots [19], and our own experiments comparing CodeCity metaphor visualizations on 2D screens and VR [20,21]. Other studies by Merino, Rüdél, and Romano involved city metaphor visualizations in 3D environments [22–24]. In Big Data, Donalek et al. explored various VR data representation approaches [25]. Simpson et al. examined movement types in VR for data visualizations [26]. In AR, Kurzhals et al. evaluated situated visualizations with eye tracking [27]. Systematic literature reviews by Dey et al. and Merino et al. have identified common features for immersive experiments and evaluations [28,29].

Our goal with *BabiaXR* is to streamline XR experiments, which often involve redundant setups and manual tasks. This framework offers automation for common features, like recording responses, tracking participant movements and gaze, displaying descriptive elements, and managing other aspects of XR experiments. Leveraging technologies like *WebXR* [30] and *WebGL* [31], which facilitate cross-platform development and integration with other applications and APIs, *BabiaXR* enables interactive data visualizations in XR. While some suggest XR can enhance software engineering by reducing learning curves and boosting creativity [32], and virtual prototyping has been explored extensively [33–35], limited evidence supports the superiority or equivalence of XR over on-screen representations in data analytics. In essence, *BabiaXR* bridges the gap between data analytics, XR experiments, and immersive environments, offering diverse experiment control options and aiming to advance empirical research on XR data visualization.

2. Software description

BabiaXR is a toolset for 3D and Virtual Reality data visualization in web browsers, enabling easy scene creation for experiments in 3D and XR environments. It automates and simplifies experiment planning processes by offering a collection of components, including visualizations, data processors, and other tools, all built on top of *A-Frame*—a JavaScript framework for creating *WebXR* scenes. *BabiaXR* aims to streamline scene preparation, automate element creation, record participant responses and movements, generate descriptive texts, and measure time spent. It utilizes *A-Frame*, requiring no specialized knowledge beyond HTML, making it user-friendly. Furthermore, comprehensive step-by-step tutorials for scene creation are available in the *BabiaXR* documentation.

2.1. Software architecture

BabiaXR leverages the *A-Frame* framework, which empowers the development of 3D, AR, and VR experiences within web browsers, using HTML and *Three.js*. It seamlessly integrates with *WebXR* and *WebGL* APIs for XR device interaction and scene rendering, respectively. The tool offers custom *A-Frame* components designed specifically for XR

data visualization and empirical experimentation, covering the entire data cycle, from retrieval and processing to visualization. Additionally, it facilitates integration with other *A-Frame* components for enhancing scenes, such as surrounding environments or shared state management. *BabiaXR* is open source, accessible on *GitLab*, and installable via *npm*.

BabiaXR operates exclusively within web browsers, using web technologies, specifically *WebXR* and *WebGL*. This choice underscores its universality, as any scene created with *BabiaXR* can be accessed on devices with compatible browsers, regardless of processing or pre-rendering needs. This sets it apart from alternatives like Unity and Unreal, which lack established standards for conducting such experiments. In the realm of visual quality and performance, *BabiaXR*’s output relies on the capabilities of *WebGL* and *WebXR*, while device resources directly impact performance and graphics quality. Furthermore, *BabiaXR* functions on the client side of web applications. When users visit a website hosting *BabiaXR* scenes, their browser downloads the necessary resources. If the user’s device supports VR/AR immersion, such as Oculus, Vive, or Hololens, the scene automatically adapts to the device’s environment (VR/AR). Conversely, if the device features a 2D screen, like a mobile device or computer, the scene is presented in a simulated 3D format on the 2D screen. This versatility, coupled with its web-based nature, enhances the accessibility and applicability of *BabiaXR* across a wide range of devices and platforms.

BabiaXR’s components fall into four categories: Charts for rendering visualizations, Queries for data querying, Data Management for processing tasks, and Experiment Management for preparing experiment features. Each component adheres to *A-Frame* standards and provides its own API. The complete API documentation is available on the official *BabiaXR* documentation website. These components follow the pipeline proposed by Chi et al. [36]. Fig. 1 illustrates the various component types within the pipeline, ultimately resulting in a fully equipped scene for XR data visualization experiments.

2.2. Software functionalities

BabiaXR offers a range of functionalities designed for conducting empirical experiments in XR environments, catering to tasks performed by human subjects:

- **Automatic Environment Creation:** Configure scenes, visualizations, data collection elements, and the camera effortlessly using the *babia-experiment* component. Default buttons control experiment initiation and termination, automatically displaying or hiding defined graphs.
- **Auto Camera Configuration:** The *babia-camera* component simplifies scene setup for on-screen and XR by automatically adapting camera settings based on the device, including controllers and interactions.
- **Browser Accessibility:** *BabiaXR* is accessible through a modern browser, requiring no additional deployment beyond hosting the HTML file on an HTTP server.
- **Time Measurement:** The *babia-experiment* component provides various time measurement options, including start and finish buttons. Set time limits and notify participants when time expires.
- **Answer Recording:** Record participants’ responses with default microphone activation. Notifications inform participants, and recorded data can be downloaded for subsequent analysis.
- **Position and Rotation Tracking:** Periodically record the camera’s position and rotation, enabling the analysis of participant movements during experiments. Download recorded data in JSON format.
- **Automatic task positioning:** Explain tasks to participants via text, audio, or video. The *babia-experiment* component allows you to define task details, and an information button in the scene triggers the corresponding explanation. Define multiple tasks using the *babia-task* component.

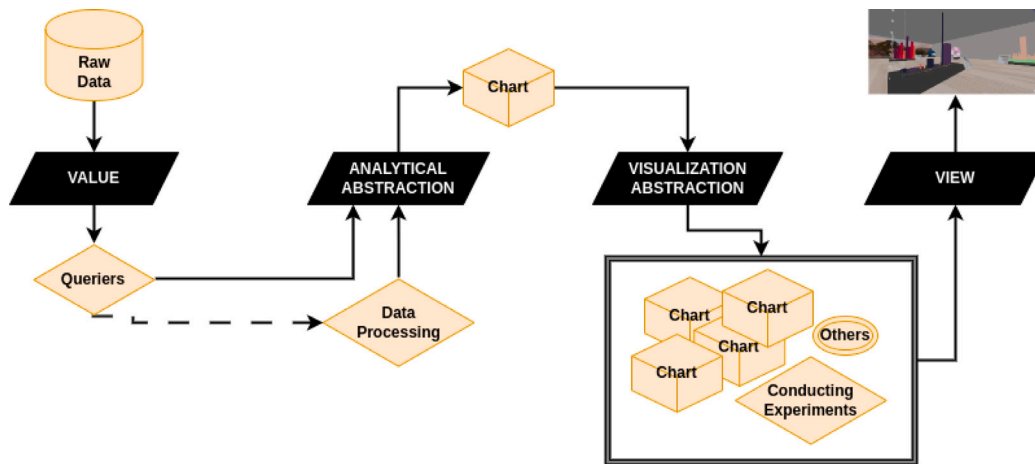


Fig. 1. *BabiaXR* components fitting the visualization pipeline proposed by Chi et al. [36] Each component type is in a different pipeline phase, resulting in a scene with all the necessary elements for conducting XR data visualization experiments.

These functionalities have been developed based on feedback from the authors and insights from related studies [22–24,37], which involve tasks, data collection from participants, and various methods of task presentation.

In terms of data visualization, *BabiaXR* offers the following features:

- **Visualizations:** *BabiaXR* provides a variety of configurable visualizations, including bar displays, cylinders, floating bubbles, doughnuts, graphs, and city displays, with positioning algorithms inherited from the *CodeCity* [38] tool.
- **Queriers:** Components for querying databases, such as CSV files, JSON files, *Elasticsearch* databases, and the GitHub API, are available in *BabiaXR*. Queried data can seamlessly integrate into visualizations.
- **Data Processing:** *BabiaXR* includes components for data manipulation, filtering, hierarchical tree construction, and real-time data updates. Participants can intuitively handle temporal evolution or specific data ranges.

Developers interested in contributing to *BabiaXR*'s development, improvement, or problem-solving can refer to the contribution guide provided in the *BabiaXR* documentation.¹

3. Illustrative examples

This section presents three illustrative examples of how *BabiaXR* improves the workflow when conducting empirical studies that contain XR visualizations. The examples presented correspond to real cases where the workflow for fetching and conducting the experiment improved by using the *BabiaXR* components for simplifying the creation of the scenario for the experiment. These examples correspond to three different cases of XR visualizations experiments, one to compare the same scene in a PC screen to an XR glasses immersion, another one to compare the *BabiaXR* visualizations to *Kibana*, a well-known platform for visualizing data in a web browser for an on-screen device, and a final one including third-party *A-Frame* components along with *BabiaXR* components showing the possibility of conducting experiments with other components.

These examples and others are available in the *BabiaXR*'s web page.²

3.1. CodeCity On-Screen vs. VR

This example compares the *CodeCity* [38] visualization in an on-screen environment and a VR environment through an empirical experiment conducted by the authors [20,21]. *BabiaXR* facilitates the visualization of the city metaphor and provides tools for fetching software metrics from a repository using *GrimoireLab* [39]. Participants perform program comprehension tasks in both environments to analyze the accuracy and efficiency of the results.

The scene is defined using HTML with *A-Frame* syntax, including components for visualizing the city and additional information panels. The scene is depicted in Fig. 2, and is prepared for defining the experimental tasks.

For preparing the experiment, we have to wrap the entire scene in the *experiment* component of *BabiaXR*, with one *babia-task* for each task, as shown in Listing 1. Task descriptions can be added as posters, audios, or videos. Fig. 3 shows the result of adding task descriptions to the scene. The visualization is hidden until the start button is pressed to ensure participants do not see the visualizations before the experiment begins. In this experiment, participants answer the tasks in both environments, with their answers recorded using the microphone.

Listing 1: Scene wrapped by an experiment component with the tasks to conduct, one defined as poster, one defined as audio, and one defined as video

```
<a-scene id="scene">
  <a-entity id="experiment" babia-experiment>
    <a-entity id="task1" babia-task="taskTitle: Task 1; taskDescription: my
      ↪ custom description"></a-entity>
    <a-entity id="task2" babia-task="taskAudio: true; taskAudioUrl: ./myaudio.
      ↪ mp3"></a-entity>
    ...
    <a-entity id="rawdata" babia-queryjson="url: data.json"></a-entity>
    <a-entity id="treedata" babia-treebuilder="field: field_list; split by: /; from:
      ↪ rawdata"></a-entity>
    <a-entity id="city" babia-boats="from: treedata; area: metric1; height:
      ↪ metric2; color: metric3"></a-entity>
    ...
  </a-entity>
</a-scene>
```

Upon finishing the experiment, participants can click the finish button, and the data with voice recordings and camera positions will be downloaded in separate files. The experimenter can analyze the experiment data and extract the answers and their validation at any time, without needing to be present during the experiment. The experiment definition feature in *BabiaXR* simplifies the planning, execution,

¹ Contribution guide: <https://babiaxr.gitlab.io/contributing/>

² <https://babiaxr.gitlab.io/aframe-babia-components/examples/experiment/city.html>

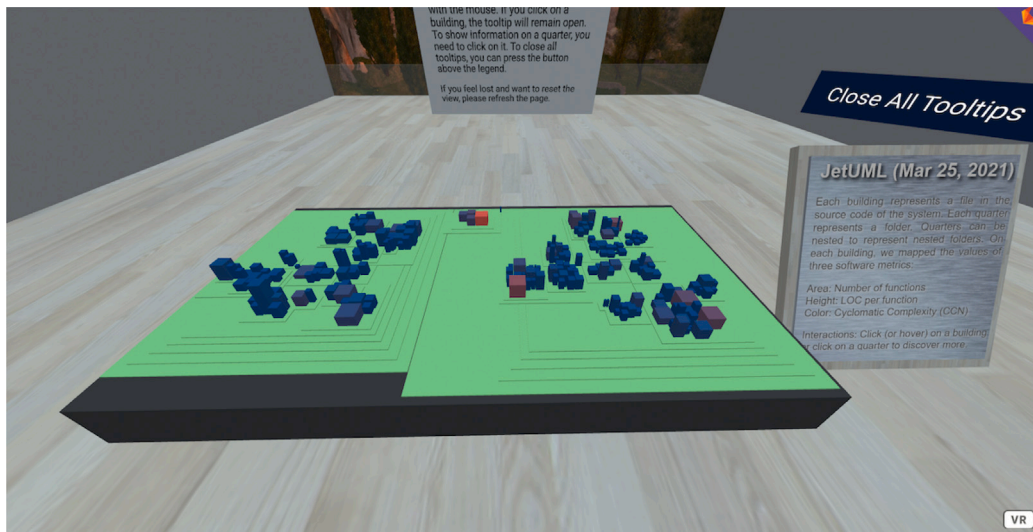


Fig. 2. CodeCity visualization of a real project, prepared to conduct an experiment with the data shown. Along the visualization, there are other entities that complement the scene with information for the experiment.



Fig. 3. Tasks defined with the experiment definition feature of BabiaXR. Each task is one of the *babia-task* entities with the corresponding parameters defined, one as poster, one as video, and one as audio.

development, and data storage process, reducing the overall execution time.

The actual experiment conducted without the experiment definition feature required the experimenter's presence for result validation, significantly increasing the execution time. The implementation of the experiment definition feature in BabiaXR aims to simplify the process and reduce the time required for conducting experiments.

3.2. A comparison between On-Screen vs VR dashboards

This example compares an on-screen web application with a VR environment for visualizing data. The scene replicates a Kibana dashboard using BabiaXR visualizations. The experiment aims to compare the two environments [40]. The scene consists of bar visualizations in a controlled environment with real-time data and a time range filter, similar to Kibana.

Customization is possible in this scene, such as adding a museum metaphor for visualizations. The final scene is depicted in Fig. 4.

The goal is to conduct tasks that require participants to search for information using different visualizations. Again, for preparing the experiment, we have to wrap the entire scene in a *babia-experiment* component with *babia-tasks* entities defining each task, as shown in Listing 2.

Listing 2: Second scene wrapped by an experiment component with tasks to conduct, with N tasks defined as isolated entities

```
<a-scene id="scene">
  <a-entity id="experiment" babia-experiment>
    <a-entity id="task1" babia-task="taskTitle: Task 1; taskDescription: my
      ↪ custom description"></a-entity>
    <a-entity id="task2" babia-task="taskAudio: true; taskAudioUrl: ./myaudio.
      ↪ mp3"></a-entity>
    ...
  <!-- UI for changing the time range -->
```

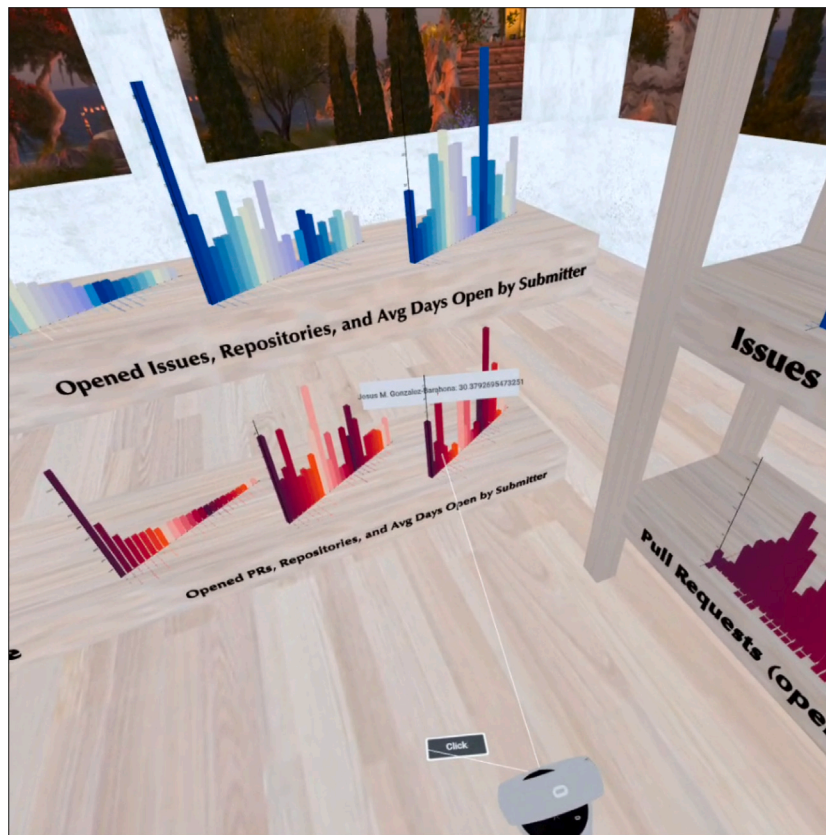


Fig. 4. Scene with several bars visualizations, customized with titles, museum environment, and shelves for locating and making the scene comfortable to participants.

```

<a-entity id="interface" babia-range-selector></a-entity>

<!-- 3d and 2d barschart + querier -->
<a-entity id="vis0" babia-queries="elasticsearch_url: es_url; index: myindex;
  ↪ request: myquery; rangeSelector: interface" babia-barsmap="height:
  ↪ field; z_axis: otherfield; x_axis: field2; animation: false"></a-entity>
<a-entity id="vis1" babia-queries="elasticsearch_url: es_url; index: myindex;
  ↪ request: myquery2; rangeSelector: interface" babia-bars="height: field
  ↪ ; x_axis: field2; animation: false;"></a-entity>
</a-entity>
...
</a-scene>

```

The visualizations are initially hidden and appear when participants press the start button. Fig. 5 illustrates the scene with an open task, showing the hidden and visible visualizations.

Participants record their answers using their voices, along with their position and rotation. The data can be analyzed by the experimenter remotely. The *Kibana* environment also recorded the screen and voice of participants for a fair comparison.

The example is available on the *BabiaXR* examples page.³

3.3. *BabiaXR* along other *A-Frame* components

This example demonstrates the integration of *BabiaXR* components with external *A-Frame* components to conduct experiments. It combines the experimentation elements provided by *BabiaXR* with a third-party *A-Frame* component that renders city streets. The scene includes three tasks defined using *BabiaXR* components: one as text, one as video, and one as audio. The code for this scene can be found in Listing 3.

³ <https://babiaxr.gitlab.io/aframe-babia-components/examples/experiment/>

Listing 3: Third scene wrapped by an experiment component with tasks to conduct, with three tasks defined as isolated entities

```

<script src="street_thirdpartysourcecode.js"></script>
...
<a-scene id="scene">
  <a-entity id="experiment" babia-experiment>
    <a-entity id="task1" babia-task="taskTitle: Task 1; taskDescription: my
      ↪ custom description"></a-entity>
    <a-entity id="task2" babia-task="taskAudio: true; taskAudioUrl: ./myaudio.
      ↪ mp3"></a-entity>
    <a-entity id="task3" babia-task="taskVideo: true; taskVideoId: myassetvideo
      ↪ "></a-entity>
    ...
    <a-entity id="mySimpleStreet" position="0 -3 0" street-streetmix-loader="
      ↪ streetmixStreetURL: <url>" ></a-entity>
    ...
  </a-entity>
  ...
</a-scene>

```

The resulting scene, depicted in Fig. 6, allows users to experience the external component while engaging in the defined tasks facilitated by *BabiaXR*. The combination of *BabiaXR* components and other *A-Frame* components offers endless possibilities for creating immersive experimental scenarios.

This example can be accessed on the *BabiaXR* examples page.⁴

Similar to previous experiments, participants in this scenario record their answers using their voices. Additionally, the position and rotation of the camera are recorded to assess how participants move and navigate within the street environment.

⁴ https://babiaxr.gitlab.io/aframe-babia-components/examples/experiment/third_party.html



Fig. 5. Second scene showing a VR dashboard with several *BabiaXR* visualizations and the experiment features for a comparison between On-Screen vs VR. A scene with the task open and the start button in the middle of the room (left) and a scene with the task open and the finish button in the middle of the room (right).



Fig. 6. Third scene using a third party component that renders city streets, including the experiment components of *BabiaXR* showing three different tasks, to conduct an experiment using the street as the main feature of the scene.

4. Impact

BabiaXR aims to simplify the process of experiment planning, execution, and data visualization in VR environments, whether for scenario comparisons or scene validation. In the experiment planning phase, users can create scenes with various data visualizations, placing and customizing them freely. *BabiaXR* also provides flexible data fetching

and filtering options, streamlining the VR scene creation process. Descriptions of experiment tasks can be added seamlessly, with the option to include audio, video, or text components.

During experiment execution, users have the flexibility to choose whether they want to be physically present, as *BabiaXR* allows for the recording of task responses, demographics, and feedback using the device's microphone. This eliminates the need for in-person experiment control. Camera position and rotation within the scene are

also recorded, enabling users to review participant actions and validate results. Alternatively, experimenters can opt to be present and take notes instead of recording participant voices, offering versatile data retrieval methods.

BabiaXR accommodates a wide range of data types, provided they follow a defined format. This flexibility allows experimenters to tailor visualizations to their specific objectives, whether for data comprehension, user behavior analysis, or visualization comparison.

BabiaXR's visualizations undergo validation, as detailed in Section 3. Multiple experiments involving over 100 participants and more than 10 experimenters from academia and industry have contributed to the tool's feature development. We believe *BabiaXR*'s capabilities are unparalleled, given the nascent state of extended reality data visualization and experimentation. We encourage the community to conduct autonomous experiments with *BabiaXR* and provide feedback. Our future plans include further validation through additional experiments and feedback collection. We are also exploring new methods for participant responses, such as numeric sliders in VR, predefined choices, and voice recognition technologies. Data storage improvements using a cloud platform are also in progress.

5. Conclusions

BabiaXR is a toolset designed for conducting empirical experiments related to data visualizations in 3D and XR environments within web browsers. Built on *WebXR* and *WebGL* standards, *BabiaXR*'s primary goal is to create comprehensive scenarios for data validation in XR. It offers components for querying, managing, and visualizing data, including well-established visualizations and cutting-edge techniques like the city metaphor visualization. These components encompass everything needed for experiment planning and execution, simplifying participant response retrieval and task definition placement within the scene.

Designed with user-friendliness in mind, *BabiaXR* streamlines the process, allowing users to work with HTML, regardless of their familiarity with 3D XR environments. Interactions within the scene are intuitively designed, requiring minimal prior experience. If training is necessary, *BabiaXR* facilitates the creation of mock-up scenes with straightforward HTML, as demonstrated in Listing 1. Comprehensive tutorials and examples are available to guide experimenters in scene creation and exploration. *BabiaXR* excels in its ability to simplify the production of various experiments, whether for data validation, scenario comparisons, or other objectives.

This toolset undergoes continuous development, enhancing its feature set and refining the existing stack based on feedback gathered from each conducted experiment. *BabiaXR* is an open-source tool with extensive documentation, featuring guides for getting started and contributing to the project.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: David Moreno-Lumbreras reports financial support was provided by Community of Madrid. Jesus M. Gonzalez-Barahona reports financial support was provided by Community of Madrid. Gregorio Robles reports financial support was provided by Madrid Regional Government. Gregorio Robles reports financial support was provided by EU Structural Funds.

Data availability

No data was used for the research described in the article.

Acknowledgments

We thank all the experiments participants, researchers and general users of *BabiaXR*.

References

- [1] Friendly M. Milestones in the history of thematic cartography, statistical graphics, and data visualization. 2008.
- [2] Friendly M. A brief history of data visualization. In: Handbook of data visualization. Berlin, Heidelberg: Springer Berlin Heidelberg; 2008, p. 15–56. http://dx.doi.org/10.1007/978-3-540-33037-0_2.
- [3] Liu S, Cui W, Wu Y, Liu M. A survey on information visualization: recent advances and challenges. *Vis Comput* 2014;30(12):1373–93.
- [4] Kraus M, Fuchs J, Sommer B, Klein K, Engelke U, Keim D, et al. Immersive analytics with abstract 3D visualizations: A survey. *Comput Graph Forum* 2022;41(1):201–29. <http://dx.doi.org/10.1111/cgf.14430>, arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.14430>.
- [5] Milgram P, Kishino F. A taxonomy of mixed reality visual displays. *IEICE Trans Inf Syst* 1994;77(12):1321–9.
- [6] Skarbez R, Smith M, Whitton MC. Revisiting milgram and kishino's reality-virtuality continuum. *Front Virtual Real* 2021;2:647997.
- [7] Kaiser P, Vasak P, Suorineni F, Thibodeau D. New dimensions in seismic data interpretation with 3-D virtual reality visualisation for burst-prone mines. 2005, p. 33–45. http://dx.doi.org/10.36487/ACG_repo/574_0.3.
- [8] Anderson A, Weng Z. VRDD: applying virtual reality visualization to protein docking and design. *J Molecul Graph Model* 1999;17(3–4):180–6.
- [9] Djorgovski S, Hut P, Knop R, Longo G, McMillan S, Vesperini E, et al. The MICA experiment: Astrophysics in virtual worlds. 2013, arXiv preprint arXiv:1301.6808.
- [10] Ibrahim Z, Money AG. Computer mediated reality technologies: A conceptual framework and survey of the state of the art in healthcare intervention systems. *J Biomed Inf* 2019;90:103102.
- [11] Bryson S. Virtual reality in scientific visualization. *Commun ACM* 1996;39(5):62–71.
- [12] Munzner T. Visualization analysis and design. CRC Press; 2014.
- [13] Few S. Show me the numbers. Analytics Pres 2004.
- [14] Batch A, Cunningham A, Cordeil M, Elmquist N, Dwyer T, Thomas BH. There is no spoon: Evaluating performance, space use, and presence with expert domain users in immersive analytics. *IEEE Trans Visual Comput Graph* 2019;26(1):536–46.
- [15] Jacob RJ, Girouard A, Hirshfield LM, Horn MS, Shaer O, Solovey ET, et al. Reality-based interaction: a framework for post-WIMP interfaces. In: Proceedings of the SIGCHI conference on human factors in computing systems. 2008, p. 201–10.
- [16] Rosenbaum R, Bottleson J, Liu Z, Hamann B. Involve me and I will understand!—abstract data visualization in immersive environments. In: International symposium on visual computing. Springer; 2011, p. 530–40.
- [17] García-Hernández RJ, Anthes C, Wiedemann M, Kranzlmüller D. Perspectives for using virtual reality to extend visual data mining in information visualization. In: 2016 IEEE aerospace conference. 2016, p. 1–11. <http://dx.doi.org/10.1109/AERO.2016.7500608>.
- [18] Raja D, Bowman D, Lucas J, North C. Exploring the benefits of immersion in abstract information visualization. In: Proc. immersive projection technology workshop, vol. 61. 2004, p. 69.
- [19] Millais P, Jones SL, Kelly R. Exploring data in virtual reality: Comparisons with 2d data visualizations. In: Extended abstracts of the 2018 CHI conference on human factors in computing systems. 2018, p. 1–6.
- [20] Moreno-Lumbreras D, Minelli R, Villaverde A, Gonzalez-Barahona JM, Lanza M. CodeCity: On-screen or in virtual reality? In: Working conference on software visualization. IEEE; 2021, p. 12–22. <http://dx.doi.org/10.1109/VISSOFT52517.2021.00011>.
- [21] Moreno-Lumbreras D, Minelli R, Villaverde A, Gonzalez-Barahona JM, Lanza M. CodeCity: A comparison of on-screen and virtual reality. *Inf Softw Technol* 2023;153:107064. <http://dx.doi.org/10.1016/j.infsof.2022.107064>, URL <https://www.sciencedirect.com/science/article/pii/S0950584922001732>.
- [22] Merino L, Fuchs J, Blumenschein M, Anslow C, Ghafari M, Nierstrasz O, et al. On the impact of the medium in the effectiveness of 3D software visualizations. In: 2017 IEEE working conference on software visualization. 2017, p. 11–21. <http://dx.doi.org/10.1109/VISSOFT.2017.17>.
- [23] Rüdél M-O, Ganser J, Koschke R. A controlled experiment on spatial orientation in VR-based software cities. In: 2018 IEEE working conference on software visualization. 2018, p. 21–31. <http://dx.doi.org/10.1109/VISSOFT.2018.00011>.
- [24] Romano S, Capece N, Erra U, Scanniello G, Lanza M. On the use of virtual reality in software visualization: The case of the city metaphor. *Inf Softw Technol* 2019;114:92–106. <http://dx.doi.org/10.1016/j.infsof.2019.06.007>.
- [25] Donalek C, Djorgovski SG, Cioc A, Wang A, Zhang J, Lawler E, et al. Immersive and collaborative data visualization using virtual reality platforms. In: 2014 IEEE International conference on big data (Big Data). 2014, p. 609–14. <http://dx.doi.org/10.1109/BigData.2014.7004282>.

- [26] Simpson M, Zhao J, Klippel A. Take a walk: Evaluating movement types for data visualization in immersive virtual reality. In: Proceedings of workshop on immersive analytics. 2017, p. 1–6.
- [27] Kurzhals K, Becher M, Pathmanathan N, Reina G. Evaluating situated visualization in AR with eye tracking. 2022, <http://dx.doi.org/10.48550/ARXIV.2209.01846>, URL <https://arxiv.org/abs/2209.01846>.
- [28] Dey A, Billingham M, Lindeman RW, Swan JE. A systematic review of 10 years of augmented reality usability studies: 2005 to 2014. *Front Robot AI* 2018;5. <http://dx.doi.org/10.3389/frobt.2018.00037>, URL <https://www.frontiersin.org/articles/10.3389/frobt.2018.00037>.
- [29] Merino L, Schwarzl M, Kraus M, Sedlmair M, Schmalstieg D, Weiskopf D. Evaluating mixed and augmented reality: A systematic literature review (2009–2019). 2020, p. 438–51. <http://dx.doi.org/10.1109/ISMAR50242.2020.00069>.
- [30] Jones B, Goregaokar M. WebXR device API. 2020, W3C Working Draft.
- [31] Jackson D, Gilbert J. WebGL 2.0 specification. Khronos Group Specification; 2020.
- [32] Elliott A, Peiris B, Parnin C. Virtual reality in software engineering: Affordances, applications, and challenges. In: 2015 IEEE/ACM 37th IEEE international conference on software engineering, vol. 2. IEEE; 2015, p. 547–50.
- [33] Gibson I, Cobb S, Eastgate R. Virtual reality and rapid prototyping: Conflicting or complementary? In: 1993 International solid freeform fabrication symposium. 1993.
- [34] De Sa AG, Zachmann G. Virtual reality as a tool for verification of assembly and maintenance processes. *Comput Graph* 1999;23(3):389–403.
- [35] Mujber T, Szecsi T, Hashmi M. Virtual reality applications in manufacturing process simulation. *J Mater Process Technol* 2004;155–156:1834–8. <http://dx.doi.org/10.1016/j.jmatprotec.2004.04.401>, URL <https://www.sciencedirect.com/science/article/pii/S0924013604005618>, Proceedings of the International Conference on Advances in Materials and Processing Technologies: Part 2.
- [36] Chi E. A taxonomy of visualization techniques using the data state reference model. In: IEEE symposium on information visualization 2000. INFOVIS 2000. proceedings. 2000, p. 69–75. <http://dx.doi.org/10.1109/INFVIS.2000.885092>.
- [37] Wetzel R, Lanza M, Robbes R. Software systems as cities: a controlled experiment. In: 2011 33rd international conference on software engineering. 2011, p. 551–60. <http://dx.doi.org/10.1145/1985793.1985868>.
- [38] Wetzel R, Lanza M. Visualizing software systems as cities. In: 2007 4th IEEE International workshop on visualizing software for understanding and analysis. 2007, p. 92–9. <http://dx.doi.org/10.1109/VISSOF.2007.4290706>.
- [39] Dueñas S, Cosentino V, Gonzalez-Barahona JM, del Castillo San Felix A, Izquierdo-Cortazar D, Cañas-Díaz L, et al. GrimoireLab: A toolset for software development analytics. *PeerJ Comput Sci* 2021;7:e601. <http://dx.doi.org/10.7717/peerj-cs.601>.
- [40] Moreno-Lumbreras D, Robles G, Izquierdo-Cortazar D, González-Barahona JM. To VR or not to VR: is virtual reality suitable to understand software development metrics?, *CoRR*, abs/2109.13768 2021, [arXiv:2109.13768](https://arxiv.org/abs/2109.13768).