Integrating XR Content in X3DOM: Supporting Navigation and Custom Functions in X3D Scenes

Raghav Sethi
Virginia Polytechnic Institute and State University
Blacksburg, VA, USA
raghavsethi96@gmail.com

Andreas Plesch
Harvard University
Cambridge, MA, USA
plesch@fas.harvard.edu

Timo Sturm
Threedy GmbH
Darmstadt, Hessen, Germany
timo.sturm@threedy.io

Nicholas Polys
Virginia Polytechnic Institute and State University
Blacksburg, VA, USA
npolys@vt.edu

ABSTRACT

This paper explores the integration of eXtended Reality (XR) content within X3DOM, a popular framework for displaying 3D content in web browsers. The importance of Web3D and the prevalent use of the X3D file format are discussed. With the deprecation of WebVR and the adoption of WebXR in web browsers, X3DOM has emerged as one of the pioneering adaptors of WebXR APIs. This paper highlights the current capabilities of X3DOM, which enable users to explore 3D scenes on regular screens and seamlessly transition into Virtual Reality (VR) mode. It showcases the use of controllers for navigation and the execution of custom functions within the X3D scenes. Additionally, the paper presents a series of developed 3D scenes that demonstrate the effectiveness of X3DOM in rendering VR content, ranging from indoor to outdoor environments, utilizing X3D nodes to display images and videos to create immersive photospheres and rich interactive scenes.

CCS CONCEPTS

• **Computing methodologies** → *Mixed / augmented reality*; **Virtual reality**; **Graphics file formats**; • **Human-centered computing** → **Web-based interaction**; **Virtual reality**.

KEYWORDS

X3D, X3DOM, WebXR, VR, XR

ACM Reference Format:
Raghav Sethi, Andreas Plesch, Timo Sturm, and Nicholas Polys. 2023. Integrating XR Content in X3DOM: Supporting Navigation and Custom Functions in X3D Scenes. In *The 28th International ACM Conference on 3D Web Technology (Web3D '23), October 09–11, 2023, San Sebastian, Spain.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3611314.3615918

Web3D '23, October 09–11, 2023, San Sebastian, Spain

ACM ISBN 979-8-4007-0324-9/23/10…$15.00
https://doi.org/10.1145/3611314.3615918

1 INTRODUCTION

The emergence of 3D web technologies has revolutionized the way 3D content is presented and experienced on web browsers. It enables users to explore and interact with immersive virtual environments without the need for specialized software or plugins. As the demand for rich and interactive 3D content grows, the importance of Web3D becomes increasingly evident.

One of the widely adopted standards for representing 3D scenes is the Extensible 3D (X3D) file format. X3D [Brutzman and Daly 2007] provides a versatile and extensible framework for describing complex 3D scenes, offering a range of features and capabilities. It has gained popularity among developers and content creators due to its interoperability and broad support across different platforms and devices.

One such framework is X3DOM [Behr et al. 2009], which has been at the forefront of integrating WebXR APIs into the display of 3D content. X3DOM provides a seamless transition from traditional 3D exploration on regular screens to immersive Virtual Reality (VR) experiences within the same browser window. This integration enables users to effortlessly engage with 3D scenes and explore virtual environments using compatible VR devices.

In this paper, we delve into the integration of VR content in X3DOM, focusing on its ability to support navigation and custom functions within X3D scenes. We explore the advancements and capabilities of X3DOM in providing a seamless and intuitive VR experience, allowing users to navigate and interact with 3D scenes using controllers and execute custom functions within the web page. Additionally, we present a series of developed 3D scenes that demonstrate the practicality and effectiveness of X3DOM in rendering VR content, ranging from indoor environments to expansive outdoor landscapes.

By examining the integration of VR capabilities within X3DOM, this research aims to contribute to the growing body of knowledge on delivering immersive 3D experiences on web browsers. We highlight the significance of X3DOM's approach in embracing the WebXR API and its potential to enhance the accessibility and usability of VR content. Furthermore, we discuss future possibilities and potential advancements in integrating VR capabilities in X3DOM, paving the way for even more engaging and interactive web-based virtual reality experiences.

2 CHALLENGE: 3D USER INTERFACES

2.1 The X3D Standard

The field of 3D user interfaces presents unique challenges in terms of designing intuitive and immersive experiences. The X3D standard, as an extensible 3D framework, addresses these challenges by providing device-agnostic solutions for viewing, navigation, and sensor interactions.

One of the key strengths of X3D lies in its ability to adapt to various viewing environments. It supports a wide range of display configurations, from traditional screens to immersive VR headsets. This flexibility allows content creators to design experiences that cater to different user preferences and device capabilities. By providing a standardized approach to rendering 3D scenes, X3D enables developers to focus on crafting user experiences rather than worrying about specific device intricacies.

The extensibility of X3D further contributes to its effectiveness in addressing the challenges of 3D user interfaces. Through its modular and open architecture, X3D enables developers to extend and customize the standard to suit their specific needs. This extensibility allows for the integration of novel interaction techniques, advanced rendering algorithms, and domain-specific functionalities, pushing the boundaries of what can be achieved in 3D user interfaces.

2.2 The WebXR Standard

In recent years, there has been a transition from the WebVR API to the WebXR API [Group 2023]. WebVR, which was initially used for virtual reality experiences in web browsers, has been deprecated in favor of WebXR, a more encompassing and versatile API that supports both Virtual Reality (VR) and Augmented Reality (AR) experiences. This transition has led to the emergence of frameworks and libraries that embrace the WebXR API, allowing developers to create immersive experiences across a wider range of devices.

2.3 X3DOM: Enabling XR with HTML5

X3DOM is a powerful framework for displaying 3D content in web browsers, offering immersive and interactive experiences beyond traditional 2D media. Leveraging HTML5, JavaScript, and the X3D file format, X3DOM provides a comprehensive set of tools for rendering and manipulating 3D scenes. At its core, X3DOM provides a comprehensive set of tools and functionalities for displaying and manipulating 3D content on the web (for example [Plesch and McCann 2015], [Paviot et al. 2020]).

X3DOM ensures accessibility by allowing users to engage with 3D content through their existing web browsers, without requiring additional installations or plugins. Its compatibility with various platforms and devices enables smooth transitions between desktop, mobile, and VR devices, ensuring a consistent experience. Through X3D's API, developers can create novel user interfaces. For example, with a few lines of JavaScript code, a user can look into the X3DOM scene through their phone using X3D Viewpoint's orientation with data from the gyroscope of the mobile device.

Prior research has explored various applications utilizing X3DOM to showcase 3D content on web browsers. For instance, students have designed web applications that incorporated X3DOM for environmental awareness [Pittarello 2013]. Hassadee et al. [Pimsuwan et al. 2012] implemented a virtual reality bookstore using X3D, demonstrating its potential for immersive e-commerce experiences. Moreover, discussions around the 'Metaverse' concept have emerged [Havele et al. 2022], emphasizing open technologies for cooperative development.

However, a key aspect remained unaddressed until recently: seamless integration of X3D content with VR headsets. The advent of WebXR now allows direct consumption of 3D content from web browsers on VR headsets, enhancing immersion and interaction possibilities in X3DOM-powered environments.

The framework prioritizes user convenience by providing intuitive navigation across different modes. Whether users are on a regular screen or in VR mode, X3DOM offers familiar controls and interactions. In VR mode, controllers enable users to navigate, manipulate objects, and perform custom functions seamlessly. This coherence enhances engagement and facilitates a smooth transition between interaction modes.

3 APPROACH AND IMPLEMENTATION

3.1 WebXR Sessions

Figure 1 illustrates the integration of X3DOM with WebXR APIs for the presentation of 3D scenes in XR format. (a) The process begins by verifying hardware and user agent support for the desired XR content. (b) If support is detected, a button is presented to users, indicating the availability of XR experiences. (c) Upon user interaction with the button, an XR Session is requested. (d) Successful acquisition of the session allows X3DOM to establish the reference space and input sources required for XR interaction. (e) Subsequently, a frame loop is initiated to handle XR input and generate images for display on the XR device. (f) This loop continues until the session is terminated either by the user agent or by the user expressing a desire to exit the XR content.

3.2 Stereo Rendering in X3DOM: Optimizing Performance

Traditional stereo rendering involves rendering the entire 3D scene twice, resulting in duplicated workload and decreased performance. This scenario is exacerbated in stereo rendering, demanding a minimum of 120 images per second to ensure a satisfactory user experience.

To overcome these challenges and enhance performance, X3DOM implements a specialized technique called single-pass stereo rendering. Instead of rendering the scene separately for each eye, X3DOM leverages simultaneous rendering of both left and right eye images into a single packed texture that is twice the width of a single eye texture.

This optimization is achieved through the utilization of hardware instancing feature available in WebGL. X3DOM takes advantage of this feature to duplicate individual objects within the scene. By using a single draw call for both objects, X3DOM achieves efficient rendering. Each instance of the object possesses unique properties that are directly utilized within the shaders to determine the appropriate projection. Additionally, these properties identify which portions of the image are unrelated to the current eye and can be discarded, further optimizing the rendering process.

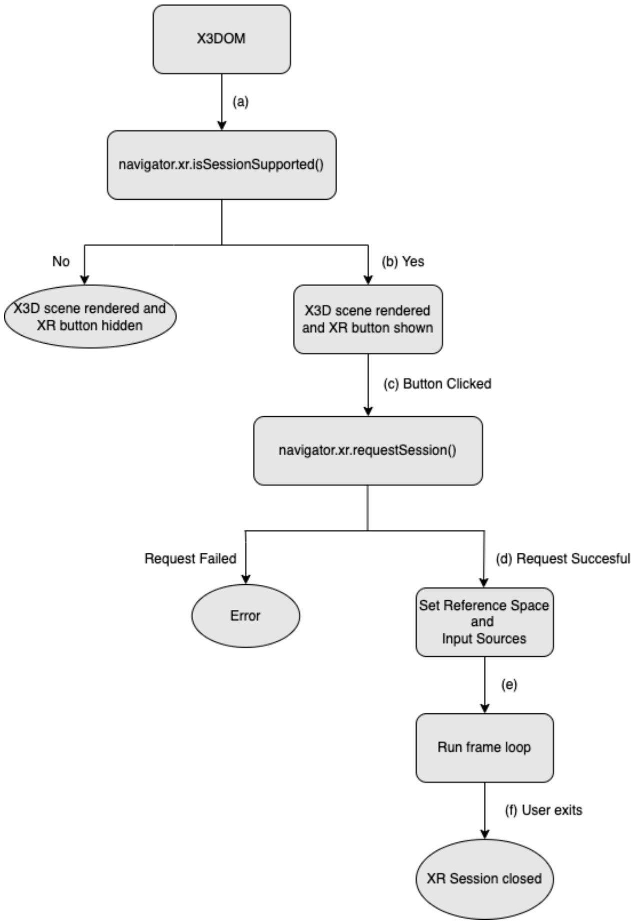

Figure 1: X3DOM WebXR API integration flow

3.3 Navigation Enhancement

While X3DOM streamlines the process of transitioning various X3D scenes into VR mode with a single click, earlier versions of X3DOM faced limitations in terms of adaptable navigation techniques that could cater to the diverse scenarios within VR mode.

To overcome this challenge, we undertook an enhancement initiative by extending the source code and integrating a navigation technique that utilizes controllers to guide navigation, enabling users to navigate through the virtual environment seamlessly.

In the navigation process, we leverage the orientation of the controller to determine the user's intended direction of movement within the VR space. By calculating the required distance for the user to travel within the VR world, based on the minimum and maximum viewpoints of the current scene, we ensure a consistent and immersive navigation experience. In addition, to enable a user-friendly navigation experience, we implemented a VR-specific zooming technique that adapts to the user's navigation speed. By employing this technique, we dynamically calculate the appropriate zoom factor based on various parameters, including the distance between minimum and maximum viewpoints, navigation speed, and the frames per second (fps) of the VR environment. The resulting calculated zoom amount allows users to seamlessly and comfortably adjust their view, enhancing immersion and the overall quality of the VR experience.

The x3dom code is publicly available at: https://github.com/x3dom/x3dom.

3.4 Custom Functions in 3DUIs

Apart from using controllers for navigation, the controller buttons can be configured to perform specific actions, enabling the execution of customized functions within the X3D scene. Leveraging the Navigator Gamepad API [MDN 2023], we can generate custom events that can be captured by a web page, allowing the execution of JavaScript code for enhanced interactivity and dynamic functionality.

4 DEMONSTRATING USABILITY: DEVELOPED 3D SCENES

To showcase the effectiveness of X3DOM in rendering VR content and its versatility in creating immersive web-based experiences, we have developed a series of X3D scenes that highlight the framework's capabilities. These scenes encompass a range of environments, including indoor and outdoor settings, to demonstrate the versatility and adaptability of X3DOM. The scenes have been tested using a WebXR emulator extension [Aoyagi et al. 2023] which enables users to responsively run WebXR applications on their desktop browser without the need for any XR devices. Additionally, the scenes underwent testing using VR Headsets such as Oculus Rift tethered to a desktop and a standalone Pico Neo3 Pro Eye headset. Figures 2, 3, and 4 represent different X3D scenes tested using the WebXR emulator extension. The extension divides the primary display into left and right views, emulating the effect of a headset.

4.1 Stationary Camera

In the indoor scenes, we have created virtual representations of real-world spaces such as museums, art galleries, and architectural interiors. Users can navigate through these environments, exploring the intricate details and experiencing the sense of presence that immersion through headsets enables. By utilizing X3D nodes to render high-resolution images and videos, we have crafted immersive photo and video spheres that allow users to feel as if they are physically present within these spaces.

4.2 Moving Camera

In contrast, the outdoor scenes showcase the vast potential of VR in X3DOM in rendering expansive landscapes and natural environments. Users can move around the scene using controllers. By incorporating X3D nodes to display panoramic images and videos, we have created panoramic views that simulate being present at those locations, providing users with a sense of awe and wonder.

Furthermore, these developed and fully functional scenes demonstrate the potential of X3DOM in various domains, including education, tourism, architecture, and entertainment. By leveraging the power of X3DOM, developers and content creators can deliver compelling and interactive experiences that engage and captivate users.

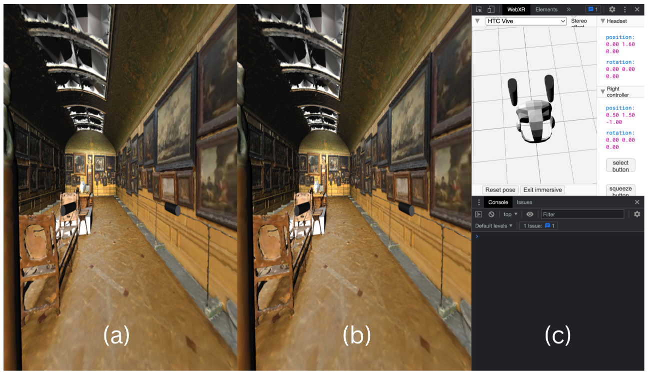

Figure 2: An X3D scene showcasing an art gallery (a) left eye view (b) right eye view (c) WebXR Emulator extension tab

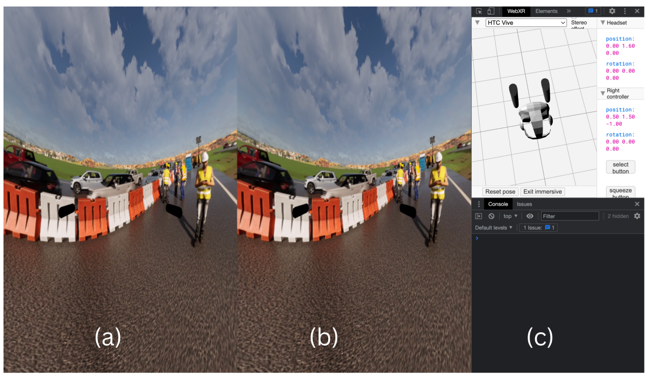

Figure 3: 360 video of roadside construction projected on a sphere (a) left eye view (b) right eye view (c) WebXR Emulator extension tab

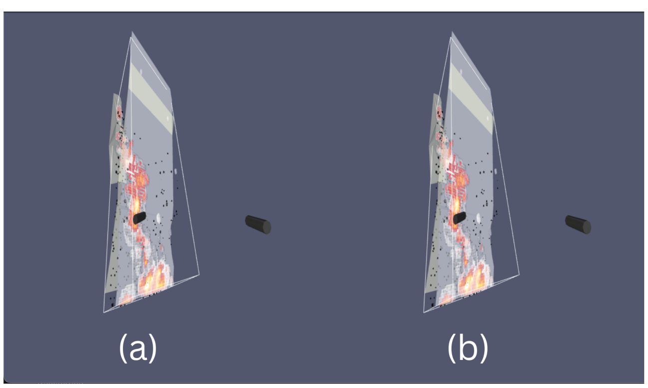

Figure 4: 3D Mining data (a) left eye view (b) right eye view

5 CONCLUSION AND FUTURE WORK

In this paper, we have explored the integration of VR content in X3DOM, focusing on its support for navigation and custom functions within X3D scenes. By integrating WebXR APIs, X3DOM enables users to transition effortlessly from regular screen-based exploration to immersive VR experiences within the same browser window. This integration empowers users to navigate and interact with 3D scenes using controllers, while custom functions add dynamic interactions and enhance user experiences.

The integration of XR capabilities within X3DOM holds significant implications for the delivery of immersive 3D content on the web. X3DOM's approach provides a seamless and intuitive user experience, allowing users to engage with 3D scenes using familiar controls and interactions. The accessibility and convenience offered by X3DOM eliminate barriers to entry and enable a wider audience to access and enjoy virtual reality experiences. The integration of custom functions enhances interactivity, allowing developers to create dynamic and personalized experiences that respond to user actions and choices.

Looking forward, there are several future possibilities and potential advancements in integrating XR capabilities in X3DOM. One area of exploration is the refinement of navigation techniques within VR environments. Advancements in controller tracking and input methods like picking objects in VR can enhance the ease and precision of user interactions, further blurring the boundary between the physical and virtual worlds. Another avenue for future work lies in expanding the range of custom functions and interactions within X3D scenes targeting see-through XR. Finally, these lessons around XR integration can be integrated into future International Standards as X3Dv4 evolves to X3D 4.1.

REFERENCES

Takahiro Aoyagi, Fernando Serrano, and Mozilla Mixed Reality Team. 2023. *WebXR emulator extension.* Retrieved June, 2023 from https://blog.mozvr.com/webxr-emulator-extension/

Johannes Behr, Peter Eschler, Yvonne Jung, and Michael Zöllner. 2009. X3DOM: A DOM-Based HTML5/X3D Integration Model. In *Proceedings of the 14th International Conference on 3D Web Technology* (Darmstadt, Germany) *(Web3D '09)*. Association for Computing Machinery, New York, NY, USA, 127–135. https://doi.org/10.1145/1559764.1559784

Don Brutzman and Leonard Daly. 2007. *X3D: Extensible 3D Graphics for Web Authors*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

W3C Community Group. 2023. *WebXR.* Retrieved June, 2023 from https://immersiveweb.dev/

Anita Havele, Nicholas Polys, William Benman, and Donald Brutzman. 2022. The Keys to an Open, Interoperable Metaverse. In *Proceedings of the 27th International Conference on 3D Web Technology* (Evry-Courcouronnes, France) *(Web3D '22)*. Association for Computing Machinery, New York, NY, USA, Article 9, 7 pages. https://doi.org/10.1145/3564533.3564575

Mozilla Developer Network MDN. 2023. *Gamepad API.* Retrieved June, 2023 from https://developer.mozilla.org/en-US/docs/Web/API/Gamepad_API

Thomas Paviot, Andreas Plesch, Lea Sattler, and Samir Lamouri. 2020. STEP and IFC Export to X3D. In *The 25th International Conference on 3D Web Technology* (Virtual Event, Republic of Korea) *(Web3D '20)*. Association for Computing Machinery, New York, NY, USA, Article 11, 13 pages. https://doi.org/10.1145/3424616.3424706

Hassadee Pimsuwan, Satidchoke Phosaard, Pimmanee Rattanawicha, and Wachara Chantatub. 2012. X3DOM Virtual Reality Book Store. In *Proceedings of the 17th International Conference on 3D Web Technology* (Los Angeles, California) *(Web3D '12)*. Association for Computing Machinery, New York, NY, USA, 183. https://doi.org/10.1145/2338714.2338750

Fabio Pittarello. 2013. Testing the X3DOM Framework for the Development of Web3D Applications. In *Proceedings of the 18th International Conference on 3D Web Technology* (San Sebastian, Spain) *(Web3D '13)*. Association for Computing Machinery, New York, NY, USA, 191–194. https://doi.org/10.1145/2466533.2466557

Andreas Plesch and Mike McCann. 2015. The X3D Geospatial Component: X3DOM Implementation of GeoOrigin, GeoLocation, GeoViewpoint, and GeoPositionInterpolator Nodes. In *Proceedings of the 20th International Conference on 3D Web Technology* (Heraklion, Crete, Greece) *(Web3D '15)*. Association for Computing Machinery, New York, NY, USA, 31–37. https://doi.org/10.1145/2775292.2775315