XR Framework: A software suite for creating and running multi-user studies in virtual and augmented reality

Sandro ROPELATO, Marino MENOZZI

Human Factors Engineering, ETH Zürich
Scheuchzerstraße 7, 8092 Zürich, Switzerland

Abstract. There has been a recent trend of performing user studies in virtual or augmented reality instead of the real world. While this brings a number of advantages, it has several drawbacks, including the high effort necessary to replicate the experimental setup in the virtual environment and the fact that most studies are tailored to a specific hardware configuration, making it difficult to run on a different setup. We present an extended reality (XR) framework that enables laypeople to create and publish XR experiments that can be run by participants on various types of XR equipment. The framework features multi-user interaction and an open asset database to create and share assets such as 3D objects, textures, and behavior scripts. It is currently under development, and we plan to release the first stable version along with the source code in the near future.

Keywords: virtual reality, augmented reality, extended reality, user studies, framework, multi-user

1. Introduction

User studies have always been an important tool in most scientific fields to gain insights into users' interactions with specific environments and verify the acceptance of new technologies.

High availability and reduced costs of virtual reality (VR) and augmented reality (AR) equipment facilitate a recent trend towards performing such user studies in virtual reality (VR) or augmented reality (AR). This brings various advantages, such as user safety when research questions require confronting users with potentially dangerous situations (Shibata & Fujihara 2002), time and cost-effectiveness in studies requiring a difficult and time-consuming setup (Mottura et al. 2003), or ethical questions when involving humans or animals (Balcombe 2004). VR and AR technologies further enable studies to be executed in a controlled environment and under replicable settings (Miller & Bugnariu 2016).

In this paper, the term extended reality (XR) is used to refer to a combination of VR and AR. Despite the many advantages of performing user studies in XR, there are a number of limitations to be considered when transferring user studies from real life into a virtual environment. For example, the setup of a believable XR environment involves a high effort, as, in most settings, a team of software engineers needs to replicate a real-world scenario in XR and develop mechanisms to enable means of interacting with the virtual objects. This greatly limits the accessibility of this technology for researchers who do not have expertise in creating such XR experiences themselves. Furthermore, most XR experiments are tailored to one single use case and bound to a specific hardware setup, making it difficult to replicate experiments in different locations, which directly affects the number of participants that can be recruited for a study.

Freely available game engines, such as Unity (Unity Technologies, San Francisco, California, USA) and Unreal (Epic Games, Cary, North Carolina, USA), have greatly increased the accessibility of game development, including XR integration. However, hardware-specific subtleties and performance-related concerns still exclude laypeople from creating and running XR studies.

Through our proposed framework, we aim to enable all users to create their own interactive XR environments without the need to write code or configure hardware equipment.

2. Concept

Requirements

As a first step, we identified the specific requirements of a framework that enables laypeople to create their own XR experiments. We accomplished this in a series of small focus groups consisting of non-computer scientists who are experienced in running user studies but have little or no experience in setting up virtual reality environments, as well as experienced XR developers and users.

We worked out the following requirements for the basic version of the XR framework:

• It must support the creation of a 3D room and allow custom objects to be placed within that room.

• At least two people need to be able to be present at the same time, and interactions between users and objects in the virtual environment need to be visible to the other user(s) and vice versa.

• It has two modes (or two different views). The first is the editor mode, where users set up the environment, define the behavior of objects, and configure the ways in which the participants interact with them. The finished scene is then stored in a central repository from where it can be accessed in the second mode, the play mode. In this mode, the participant wears a head-mounted display or AR goggles and operates the input devices to take control over the avatar defined in the editor mode.

• A repository of predefined assets (e.g., 3D models, textures, and behavior scripts) is available in the editor mode. The user can add them to the scene. The repository can be extended by the developers, and the new content is made available to all users.

• The user defines the number of avatars (players) that can be in the room concurrently. When running the experiments, these players are synchronized via an online server, so the players can be in physically different locations but participate in the experiment by meeting virtually in a collaborative space.

Figure 1 gives an overview of an example workflow within the proposed framework.

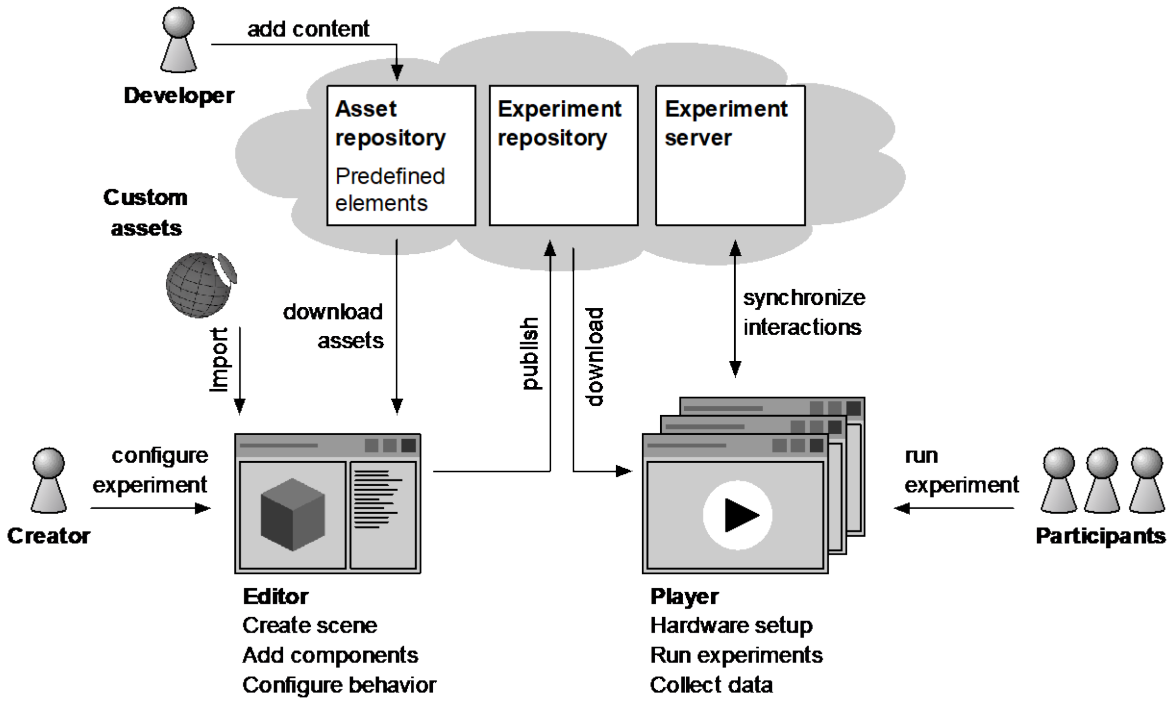

Figure 1. *Overview of the example workflow. The experiment creator designs an XR experiment using the framework's editor mode. The creator can add predefined assets from the asset repository or add custom assets. A finished experiment is published in the experiment repository, from where it can be downloaded by the experiment players on different machines. The experiment server synchronizes the participants' interactions when multiple users participate in the experiment concurrently.*

3. Realization

We used Unity as the foundation for our framework, taking advantage of its elaborate features, especially integration of common VR and AR hardware.

3.1 Editor

As a first step, we built the editor view of the framework, where a user can create a new virtual scene from scratch. As with other 3D editors, users are offered a set of tools to modify the content of the scene. These include adding a new object from the asset database, removing, translating, rotating, and scaling objects, and setting object-specific properties, such as mass and whether the object should collide with others.

Figure 2 shows a screenshot of a scene being edited.

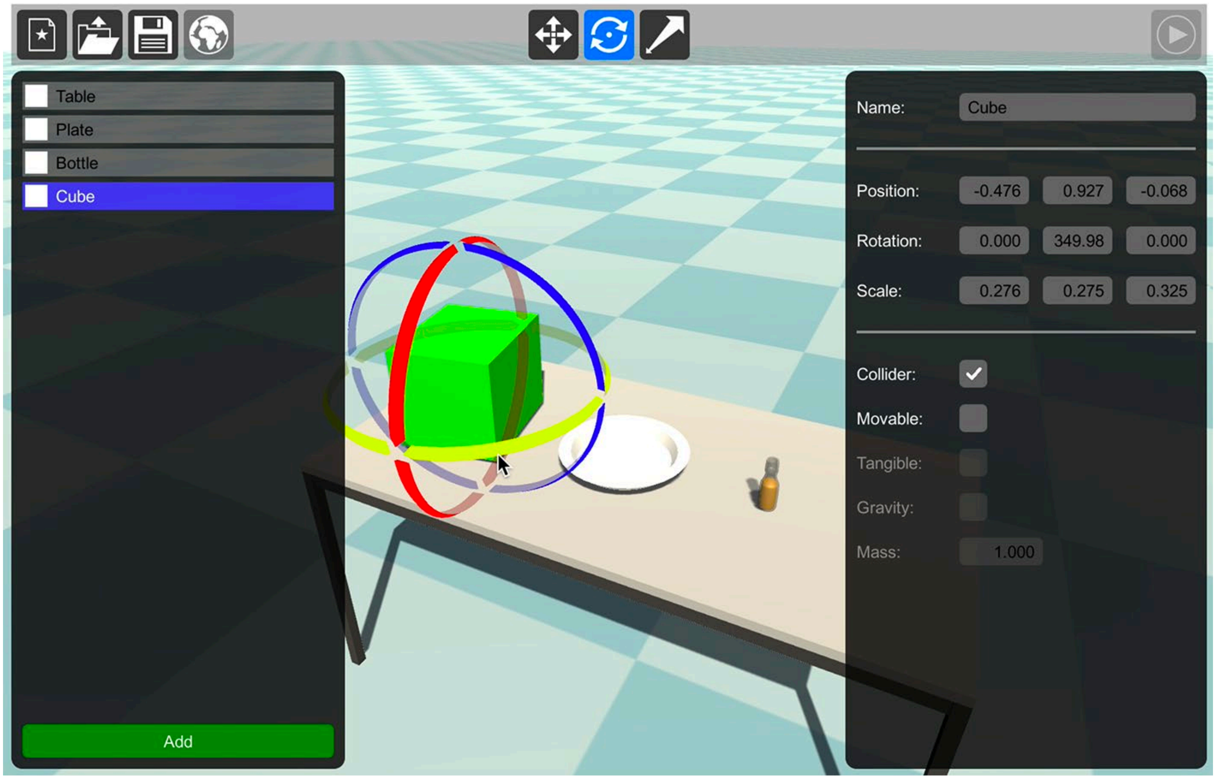

Figure 2. *Editing a scene with the XR framework editor. The scene explorer on the left shows a list of all objects present in the scene. One or multiple objects can be selected and translated, rotated, and scaled using the respective handles directly in the 3D scene. The right panel shows properties of the selected object(s) and allows the user to change them.*

The scene can be locally saved as an XML file. The file includes all objects, properties, and scene settings and does not depend on the platform running the editor. In addition, the scene can be published to the cloud server from where it can be run with the XR framework player.

3.2 Player

The XR framework player is used to run scenes previously created with the editor. A scene can be loaded from a local file or directly accessed by connecting to a published scene on the server. While loading an object from a local file allows only one local user in the scene, multiple users can be present concurrently in the scene running on the server. The player also provides a test environment to locally test scenes. This is possible even without appropriate VR or AR equipment. Keyboard and mouse inputs simulate actions such as looking and walking around or grabbing and moving an object.

Figure 3 shows an object being moved using a mouse.

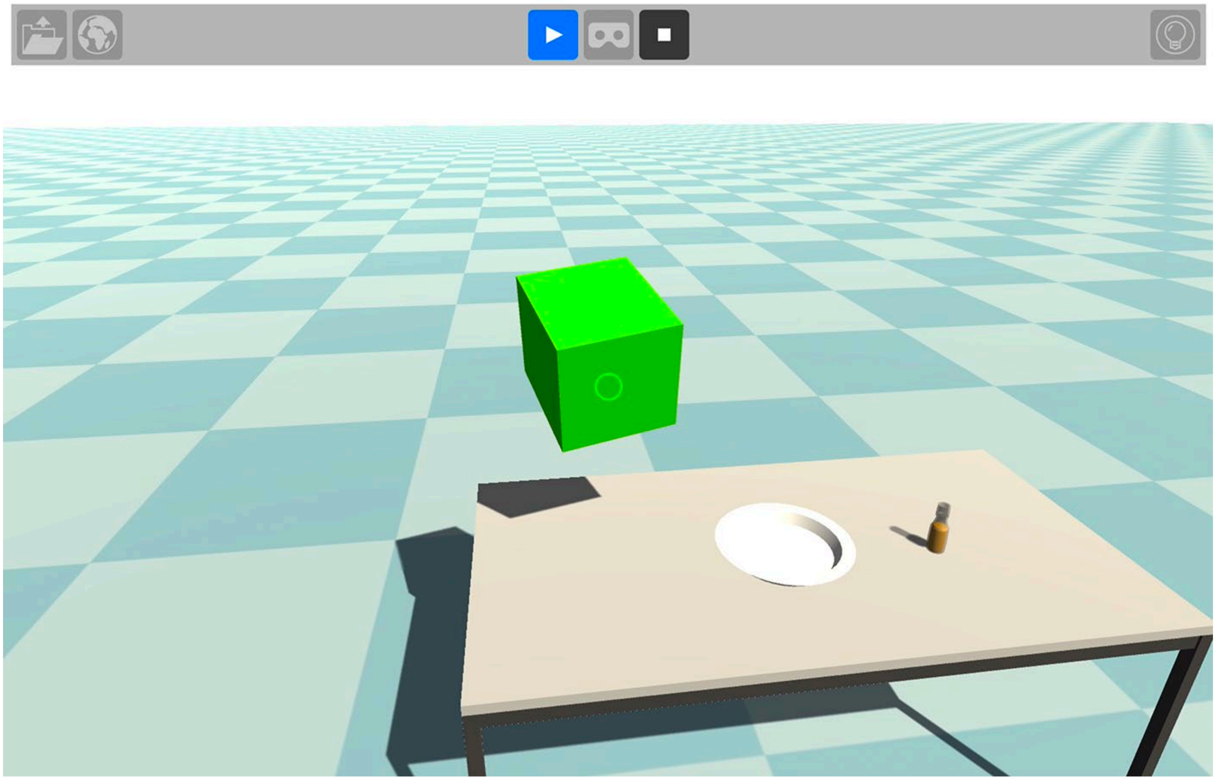

Figure 3. *A local scene being tested with the XR framework player. The user can walk through the scene by using the W, A, S, and D keys on the keyboard and look around by moving the mouse. Tangible objects can be grabbed and moved.*

3.3 Multi-user interaction

The XR framework allows multiple users to be present in an online collaborative room. It therefore synchronizes all movements and actions of the users, so they can interact with each other. This requires a server to track each user's actions and send updates to the other players.

Figure 4 shows an excerpt from a session with two clients running a scene.

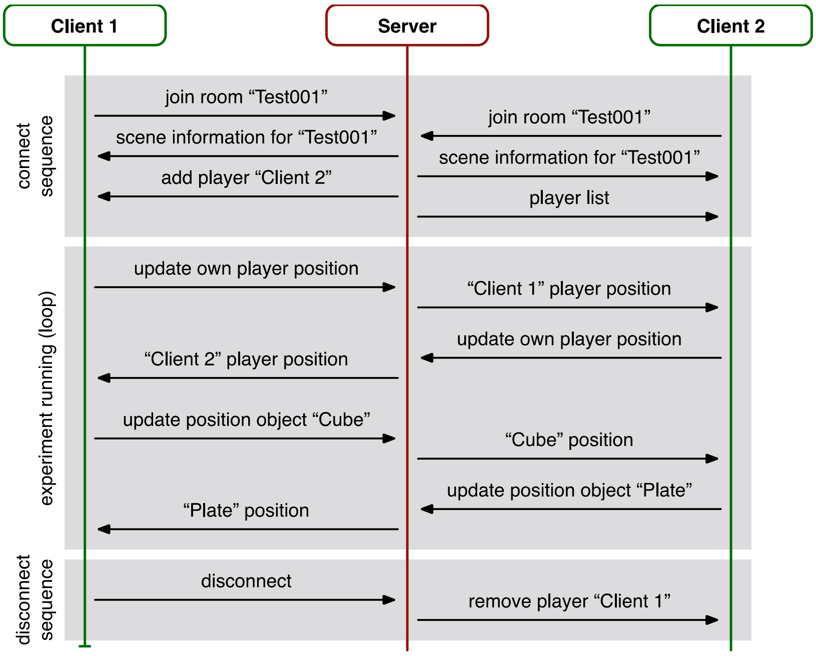

Figure 4. *Communication between two clients and a server. The top part shows the connection sequence where both clients join an online room. The middle part is the main loop where clients continuously update their position and the positions of objects they interact with (e.g., “Cube” or “Plate”). The bottom part shows the end of the experiment where a client disconnects.*

4. Current status and next steps

The prototype of the XR framework offers a helpful, yet very limited, set of features. The current version has proven to have stable multi-user synchronization and seamless integration with current VR equipment.

We are working on extending the framework to include other types of hardware, such as AR headsets and hand and eye trackers, and integrate more tools to create and customize XR experiments.

Our goal is to make the XR framework and its source code available as soon as possible. Updates are published on our research platform *https://spaghettivar.org*. In this way, we hope that users and creators will join us in creating content and features and contributing to the development of the framework.

5. References

Balcombe J (2004) Medical Training Using Simulation: Toward Fewer Animals and Safer Patients. Alternatives to laboratory animals 32 Suppl 1B:553-560

Miller HL, Bugnariu NL (2016) Level of Immersion in Virtual Environments Impacts the Ability to Assess and Teach Social Skills in Autism Spectrum Disorder. Cyberpsychology, Behavior, and Social Networking 19(4):246-256

Mottura S, Viganò GP, Sacco M (2003) Virtual Reality for Product Layout Configuration. Proceedings of the 1st International Conference on Research in Virtual and Rapid Prototyping, VRAP 2003. Leiria, Portugal

Shibata T, Fujihara H (2002) Development of Railway VR Safety Simulation System. Quarterly Report of RTRI 43:87-89

Gesellschaft für
Arbeitswissenschaft e.V.

Arbeit HUMAINE gestalten

67. Kongress der
Gesellschaft für Arbeitswissenschaft

Lehrstuhl Wirtschaftspsychologie (WiPs)
Ruhr-Universität Bochum

Institut für Arbeitswissenschaft (IAW)
Ruhr-Universität Bochum

3. - 5. März 2021

GfA-Press

Bericht zum 67. Arbeitswissenschaftlichen Kongress vom 3. - 5. März 2021

Lehrstuhl Wirtschaftspsychologie, Ruhr-Universität Bochum
Institut für Arbeitswissenschaft, Ruhr-Universität Bochum

Herausgegeben von der Gesellschaft für Arbeitswissenschaft e.V.
Dortmund: GfA-Press, 2021
ISBN 978-3-936804-29-4

NE: Gesellschaft für Arbeitswissenschaft: Jahresdokumentation

Als Manuskript zusammengestellt. Diese Jahresdokumentation ist nur in der Geschäftsstelle erhältlich.

Alle Rechte vorbehalten.

© GfA-Press, Dortmund
Schriftleitung: Matthias Jäger
im Auftrag der Gesellschaft für Arbeitswissenschaft e.V.

Ohne ausdrückliche Genehmigung der Gesellschaft für Arbeitswissenschaft e.V. ist es nicht gestattet:

- den Kongressband oder Teile daraus in irgendeiner Form (durch Fotokopie, Mikrofilm oder ein anderes Verfahren) zu vervielfältigen,
- den Kongressband oder Teile daraus in Print- und/oder Nonprint-Medien (Webseiten, Blog, Social Media) zu verbreiten.

Die Verantwortung für die Inhalte der Beiträge tragen alleine die jeweiligen Verfasser; die GfA haftet nicht für die weitere Verwendung der darin enthaltenen Angaben.

Screen design und Umsetzung
© 2021 fröse multimedia, Frank Fröse
office@internetkundenservice.de · www.internetkundenservice.de