MPEG-4 Based Recording and Replay of Collaborative Virtual Reality Sessions

Mojtaba Hosseini and Nicolas D. Georganas

Multimedia Communications Research Laboratory
School of Information Technology and Engineering
University of Ottawa, Ottawa, Canada
[mojtaba, georganas]@mcrlab.uottawa.ca

Abstract

The growing complexity of Virtual Reality simulations and the inclusion of a greater range and number of events and interactions in such simulations precipitates the need for the recording of these events to allow their subsequent viewing, discussion and perhaps modification. Furthermore, storage of such recordings in a format defined by an international standard allows their easy use and accessibility as well as distribution. This paper presents a framework for MPEG-4 based recording and replay of Virtual Reality sessions.

1. Introduction

As Virtual Reality (VR) simulations evolve, they more closely mimic the richness, complexity and abundance of events in the real world. Just as video recorders are used to capture events in the real world, recorders of VR simulations are necessary in order to allow subsequent replay of past events in the virtual world. Such recordings can be used to discuss training simulations, serve as a form of latecomer support in collaborative VR or as new content to be used inside other simulations.

Event logging has already been employed in CVEs such as the DIVE system [1] but perhaps the most flexible of recording schemes lies in the concept of "temporal links" as proposed and implemented in the MASSIVE-3 system [2]. However, these recordings can only be replayed by the specific system in which they were recorded. As a result, a DIVE or MASSIVE session recording can not be used by other systems. In order for the recordings of CVE sessions of different systems to be usable by a variety of players, they must adhere to a recognized standard. This paper presents MPEG-4 [3] based recording of CVE sessions but in fact it could be extended to the recording of stand-alone VR simulations as well.

2. CVE Session Recording

In order for the recorder to be application independent participating applications are left recording-unaware and the recorder joins the collaborative session as only a receiver and others collaborate as they would normally (Figure 1).

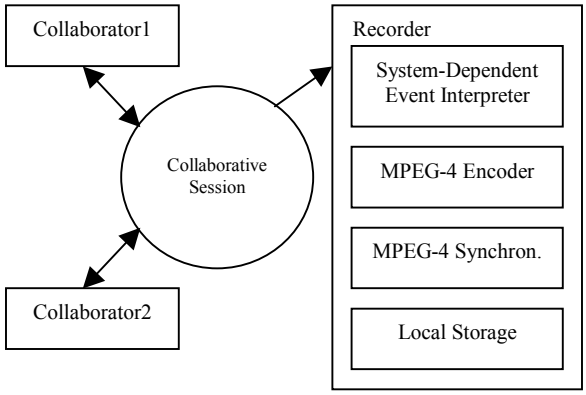

Figure 1. Recording approach

The recorder receives events (events in this context imply both 3D objects and the corresponding updates) during the session, as would any other collaborator. If the system supporting collaboration is not MPEG-4 compliant, an event interpreter component must make a mapping between system events and MPEG-4 supported updates. The mapped updates are then encoded and appended with temporal properties according to the standard and then stored.

2.1 Recording and Replay Processes

The recording begins with the receipt of the first event from the session. (Note: all capitalized words in this section are defined in the MPEG-4 standard). A Clock

Reference Stream, containing an Object Clock Reference (OCR) that will be referenced by all streams, is defined, encoded and stored along with the SLConfigDescriptor. The first event is appended with a Decoding Time Stamp (DTS) equal to the OCR, indicating that the first event should be decoded as soon as it becomes available. All subsequent events are appended with a DTS equal to the OCR plus the time duration between the receipt of the first event and the current event. Hence the sequencing and timing of events in a stream are measured as they were received. By referencing the same OCR, the temporal relationship between various streams is also preserved.

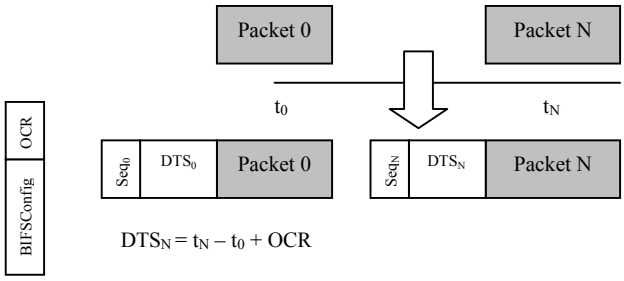

Figure 2. Recording process

For replay, the MPEG-4 Synchronization Layer decodes the OCR and the various time stamps and passes the information to the upper layer. The player application uses the time stamps to decode the stream of events at appropriate intervals and hence replay the recording.

2.3 Recording Size

Due to MPEG-4's high compression of both the 3D objects and the updates, we have found BIFS-encoded recordings lasting a few minutes to occupy storage only in the range of 100s of KB as shown.

Session Description	Number of updates	Geometry	File Size
Two users assembling a bicycle from its parts (3 min)	660	298 KB	316 KB
An avatar displaying hand and finger gestures(18 joints continuously animated for 1min)	1014	81 KB	181 KB
Two avatars playing tag (4 min of continuous walking)	2200	182 KB	289 KB

Type of update	Update rate (updates/sec)	Update size (Bytes)
Object manipulation	30	20-24
Avatar position	5	41
Avatar Hand/Finger (18 joints)	20	78

2.4 Interoperability

Recording sessions in the MPEG-4 format opens the door to its usage by not only MPEG-4 players but also a wide variety of other applications. This can be done by storing the recordings in the Extensible MPEG-4 Textual Format (XMT), "a framework for representing MPEG-4 scene description using textual syntax" [5]. XMT provides interoperability of MPEG-4 content with Extensible 3D (X3D and hence VRML) developed by Web3D Consortium and the Synchronized Multimedia Integration Language (SMIL) of W3C Consortium. This flexibility can be the cause of VR recordings becoming major contributors to 3D content production that is easily accessible.

3. Conclusion

The MPEG-4 standard defines the coding of audio/visual objects as well as their multiplexing and synchronization and hence can be used as a common format for all VR recordings. This paper presented recording of collaborative VR sessions based on this standard with the hope that such recordings become as widely used and accessible as those of audio and video.

4. References

[1] E. Frecon, C. Greenhalgh, and M. Senius, "The DIVE-BONE – an application-level network architecture for Internet-based CVEs", *Proceedings of ACM Virtual Reality Software and Technology (VRST'99)*, London, UK, December 1999, pp. 58-85.

[2] C. Greenhalgh, J. Pubrick, S. Benford, M. Craven, A. Drozd, I. Taylor, "Temporal Links: Recording and Replaying Virtual Environments", *Proceedings of AC 8th international conference on Multimedia (ACM MM'00)*, Los Angeles, USA, November 2000, pp. 67-74.

[3] ISO/IEC 14496-1, "Information Technology – Coding of audio-visual objects, Part 1: Systems", January 1999

[4] ISO/IEC 14496-1/Amd.3, "Amendment 3: Extensible MPEG-4 Textual Format (XMT)", July 2000