2014 IEEE 14th International Conference on Advanced Learning Technologies

A Survey of Frameworks and Game Engines for Serious Game Development

Brent Cowan and Bill Kapralos
Faculty of Business and Information Technology
University of Ontario Institute of Technology
Oshawa, Ontario, Canada. L1H 7K4.

e-mail: brent.cowan@uoit.ca bill.kapralos@uoit.ca

***Abstract*—Given the sparsity of standard game engines and frameworks for serious game development, developers of serious games typically rely on entertainment-based game development tools. However, given the large number of game engines and frameworks dedicated to entertainment game development, deciding on which tool to employ may be difficult. A literature review that examined the frameworks and game engines used to develop serious games was recently conducted. Here, a list of the most commonly identified frameworks and game engines and a summary of their features is provided. The results presented provide insight to those seeking tools to develop serious games.**

Keywords-serious games; game engine; framework; review;

I. INTRODUCTION

Given the ubiquity of video game use across a large demographic (i.e., male, female, from the very young to the very old), and the ability of video games to engage and motivate learners, the popularity of serious gaming (video games whose primary purpose is education and training) has seen a recent surge across all areas of education and training. Serious games "leverage the power of computer games to captivate and engage players for a specific purpose such as to develop new knowledge or skills" [1] and greater engagement within an educational setting leads to higher academic achievement [2] (greater details regarding the benefits of serious games are available by Squire [3] and Bergeron [4]). Despite the popularity of serious games and the benefits they afford, the development of effective serious games is a difficult and complicated process requiring an appropriate balance between game design and instructional design (a lack of proper instructional design can lead to an ineffective serious game) [5,6].

Further complicating matters, there are currently very few development tools (game engines and frameworks in particular), designed specifically for serious game development; serious game developers have generally relied on game engines and frameworks designed for entertainment games despite the inherent differences between serious game and entertainment game development [7]. More specifically, serious games are often produced by small teams in contrast to many of the commercial entertainment games where the development team can include dozens of developers.

There are many commercial game engines that provide advanced rendering technologies, and simple tools for content creation allowing game developers to reuse code thus decreasing development time and costs [8]. Sherrod defines a game engine as "a framework comprised of a collection of different tools, utilities, and interfaces that hide the low-level details of the various tasks that make up a video game" [9]. The terms game engine and framework are often used interchangeably. For the purpose of this paper, the term game engine refers to the functionality and features that become part of the completed game. A framework includes a game engine in addition to external tools and resources that simplify the process of game development.

Here, we present the results of a literature review that was conducted across three databases (*Google Scholar*, the *Institute of Electrical and Electronics Engineers* (IEEE) Xplore digital library, and the *Association for Computing Machinery* (ACM) digital library), to examine the tools commonly used to develop serious games, essentially compiling a list of tools used by serious game developers. This list will allow us to investigate the game engine and framework features that are most important to developers. This may prove to be useful to the developers seeking such tools.

II. BACKGROUND

A list of the primary features provided by the majority of most modern game engines is provided below. The code responsible for providing this functionality becomes a part of the finished game:

- **Scripting**: Simple code that can be written to control game objects and events.
- **Rendering**: The speed and accuracy that a 3D scene is generated, and the visual effects provided.
- **Animation**: The movement and deformation of objects such as characters.
- **Artificial Intelligence**: Steering behaviors such as pursuing, dodging, and fleeing, are combined with path finding.
- **Physics**: Objects respond accurately when collided with or in response to force or pressure applied to them.
- **Audio**: Spatial rendering of audio allows sounds to have a location in the environment. Filtering can add variation and environmental cues such as reverberation.
- **Networking**: Allows players to interact with other players within the game by sharing data through a network.

978-1-4799-4038-7/14 $31.00 © 2014 IEEE
DOI 10.1109/ICALT.2014.194

Game creation frameworks generally provide a graphical user interface (GUI) which often ties together several editors. Listed below are some of the editing tools commonly included with game creation frameworks:

- **Level Editor**: Also known as a world editor, this tool aids in the creation of virtual 2D or 3D environments (game levels, maps, etc.).
- **Script Editor**: Scripts can be attached to objects selected in the level editor to customize their behavior.
- **Material Editor**: Shader code is edited and combined with images to form the surface of objects or to create visual effects.
- **Sound Editor**: Volume, attenuation and other settings can be combined with filter effects provided by the sound engine.

Level and script editors can also be part of a game engine and at times, these components may be purposely included with a finished (shipped) game to encourage modification ("modding") of the game itself by the users.

III. Results: The Surveys

A. Survey Methods

Two surveys were conducted. In the first survey (Survey 1) a search of game engines and frameworks that are popular among serious game developers was conducted by searching three databases: Google Scholar, ACM digital library, and the IEEE Xplore digital library. The search terms "serious game", "educational game", and "simulator" were used to reveal approximately 200 academic publications related to serious games and game development. These publications were then scanned for the names of game engines without regard to the context in which the engine was mentioned. Twenty game engines and frameworks that were mentioned in more than one publication were deemed worthy of further investigation. For each of these 20 game engines, the same three databases were used to survey the engine's popularity based on the number of search results returned. In each case the engine name was combined with the search terms described above ("educational game", "serious game", or "simulator"). Many game engines including Unity, Unreal, Torque, and Olive, have names that are common words in the English language whose meaning may not necessarily have any relation to game engines or frameworks. To separate the search results specific to game engines and frameworks from those that did not relate to game development, the first thirty documents from each search term were reviewed. Dividing the number of documents found that referred to the engine by the total number of documents reviewed provides an approximate percentage of the search results relating to the game engine. The results were then normalized to ensure that each of the three search terms were equally weighted.

After completion of Survey 1, Survey 2 was conducted to determine the engines or frameworks serious game developers are actually using. This survey consisted of conducting a search within the Google Scholar, IEEE digital library, and ACM digital library, with the search term "serious game". The search revealed several thousand papers across each of the three databases. However, the papers were manually reviewed sequentially to determine the first thirty papers from each database that met the requirements. Those papers that did not meet the requirements were disqualified and not considered further. For example, with respect to the search of the ACM digital library, over 200 papers were manually reviewed to find the 30 papers that met the requirements. The papers that did meet the requirements were then analyzed (manually reviewed) to determine which game engine was used. The majority of these papers (approximately 80%) were disqualified since they failed to describe the game engine used, described more than one serious game, or described a serious game that was developed from "scratch" without the use of a specific game engine or framework. Papers were also removed from the list if they featured a game that had been discussed in a previous paper, or if the first author matched that of a previous paper. The first thirty papers for each database that satisfied the above requirements were collected to determine the most popular engines utilized by serious game developers.

B. Results

The results of Survey 1 are summarized in Fig. 1 where the *x*-axis represents the search engine/framework and the *y*-axis represents the normalized occurrence (the number of search results) for each of the three search terms ("educational game", "serious game", and "simulator"). The results of Survey 2 are summarized in Table 1 where the top ten most utilized engines and frameworks are listed in order from left to right. 23.3% of the serious games found in the search were developed with Unity, 13.3% with Unreal, and 8.9% were developed using Torque. In addition, Table 1 compares the features (important to developers of serious games) provided by the top frameworks.

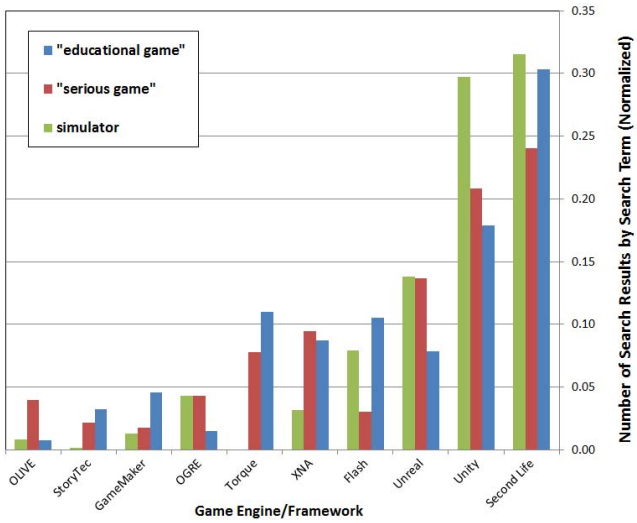

Figure 1. Game engine/framework vs. normalized occurrence (number of search results) by search term. The *x*-axis represents the search engine/framework and the *y*-axis represents the normalized occurrence (the number of search results) for each of the three search terms.

	Unity	Unreal	Torque	HTML5	Flash	Source Engine	Ogre	Second Life	GameMaker	XNA
Level editor	■	■	■		■	■		■	■	
Scripting	■	■	■	■	■			■	■	
C++		■	■			■	■			
Networking	■	■	■	■	■	■		■	■	■
3D Graphics	■	■	■			■	■	■		■
Shader effects	■	■	■			■	■		■	■
Dynamic shadows	■	■	■			■	■			■
Physics	■	■	■			■		■	■	■
Artificial Intelligence	■	■	■			■			■	
Free non-commercial	■	■	■	■		■	■	■		■
Free for commercial				■			■			■
Mobile Devices	■	■		■	■		■		■	■
Web player	■			■	■				■	■

Table 1. Game engines and frameworks and their features.

IV. Discussion and Conclusions

Two surveys were conducted using three academic databases. The first survey (Survey 1) compared the number of papers mentioning the name of each engine in relation to the search terms "serious game", "educational game", and "simulator". The frameworks and engines that scored well in this survey are those that are often discussed in academic writings. Second life was by far the "most talked about" even though Second Life is not specific to serious games. The serious games created with Second Life are essentially modifications ("mods") that exist as a location within the Second Life world. After Second Life, Unity, Unreal, Flash, XNA Game Studio, and the Torque game engine had the greatest number of search results. The second survey attempted to discover which game engines and frameworks were used most often to develop serious games. A search was performed using the search term "serious game". Papers detailing the creation of a serious game were downloaded and read in order to determine which engine was used. Five of the top ten engines are full featured game creations frameworks and eight of the top ten are commercial products. However, six of these commercial game engines are free for non-commercial use. Although there are a few game engines designed specifically for serious game development (e.g., the Delta3D simulation and gaming engine [10]), they do not appear our list of top ten most utilized game engines.

The results presented here suggest that serious game developers are primarily using game engines and frameworks that were designed specifically for the creation of leisure or entertainment games. Given the disparity of available resources to serious game developers in comparison to commercial (entertainment) game developers, it is peculiar that they chose the same tools. This suggests that the currently available serious game engines may be lacking many of the features found in commercial engines. The results may also indicate that game engines designed specifically for serious games do not simplify the process beyond that of commercial entertainment focused engines and frameworks. Each game engine/framework has a number of features that may lend itself to a specific serious game development project. Table I summarizes ten of the most utilized game engines and frameworks along with common features that are important to developers of serious games.

Although a significant amount of time and effort was placed to conduct the searches and review the hundreds of resulting papers, there are limitations to our results. More specifically, our results are specific to papers within three academic databases (Google Scholar, IEEE digital library, and ACM digital library), and as a result, game engines and frameworks that are not described in any research-based publications may have been missed altogether. Our results (Survey 1 and 2) are also subject to any bias Google Scholar, IEEE, or ACM may have relating to the ordering of the search results. Furthermore, although access to the ACM and IEEE digital libraries was available, 46% of the articles revealed in the Google Scholar search were not freely accessible and thus not considered. Despite these limitations, the above surveys do help to remove any personal bias on the part of the authors in selecting, and ranking the frameworks. We are confident that the results presented here will be useful to designers and developers of serious games and may serve as a first step to their development projects.

Acknowledgment

The financial support of the *Natural Sciences and Engineering Research Council of Canada*, the *Social Sciences and Humanities Research Council of Canada* IMMERSe initiative, and the *Canadian Network of Centres of Excellence*, GRAND initiative is appreciated.

References

[1] K. Corti, "Game-based learning; a serious business application," PIXELearning. Coventry, UK, 2006.

[2] V. J. Shute, M. Ventura, M. Bauer, and D.Zapata-Rivera, "Melding the power of serious games and embedded assessment to monitor and foster learning," In U. Ritterfeld, M. Cody, and P. Vorderer (Eds), Serious Games. Mechanisms and Effects.Routedle Publishers, New York, NY. USA, 2009.

[3] K, Squire, and H. Jenkins, "Harnessing the power of games in education," *Insight*, vol. 3, no. 1, pp. 5-33, 2003.

[4] B. Bergeron, "Developing serious games," Thomson Delmar Learning, Hingham, MA. USA, 2006.

[5] K. Becker, and J. Parker, "The Guide to Computer Simulations and Games," John Wiley and Sons, Inc., Indianapolis, IN USA, 2011.

[6] N. Iuppa, and T. Borst, "End-to-End Game Development: Creating Independent Serious Games and Simulations from Start to Finish," Focal Press, Oxford, UK, 2010.

[7] P. Petridis, I. Dunwell, D. Panzoli, S. Arnab, A. Protopsaltis, M. Hendrix, and S. de Freitas, "Game engines selection framework for high-fidelity serious applications," *International Journal of Interactive Worlds*, vol. 2012 (2012), Article ID 418638.

[8] M. Lewis, and J. Jacobson, "Game Engines," *Communications of the ACM*, vol. 45, no. 1, pp. 27, 2002.

[9] A. Sherrod, "Ultimate 3D Game Engine Design & Architecture," Charles River Media, 2007. Boston, MA.

[10] R. Darken, P. McDowell, and E. Johnson, "Projects in VR: the Delta3D open source game engine," *IEEE Computer Graphics and Applications*, vol. 25, no. 3, pp. 10-12, 2005.