

NASA KSC – Internship Final Report



**Immersive Simulations and Engineering Environment (iSEE)**

Brianna Hodge

Kennedy Space Center

Major: Visualization

Modeling and Simulation Fall Session

12/11/2018

## Increasing the Realism of Simulations

Brianna Hodge  
*Texas A&M, College Station, TX, 77840*

### Nomenclature

*iSEE* = Immersive Simulation and Engineering Environment  
LCC = Launch Control Center  
NASA = National Aeronautics and Space Administration

## I. Introduction

### A. Abstract

The Immersive Simulations and Engineering Environment (iSEE) lab at Kennedy Space Center has been operating with its current team since 2017. The lab provides National Aeronautics and Space Administration (NASA) with simulations that show how a strenuous task can cause stress or injury on a body. Previously the lab was using software that was efficient in ergonomic analysis of a human body, but limited in creating a realistic simulation. To produce a more realistic simulation, the iSEE invested in new software that could potentially create an environment that is closer to reality.

### B. Introduction

The iSEE lab focuses on using virtual simulation to stress the importance of ergonomics and its effect on the human body. The team accomplishes this by using a variety of software that mimics the performed movement while collecting data on how that movement effects the actor. Using a virtual environment allows the actor to be unaffected by the action being performed while letting the team monitor the risk of the given task. Analytic simulation software, a program that implements ergonomics and creates simulations, was originally the only simulation creation software used. But the iSEE lab is now implementing the use of game engines. Game engines help create simulations that appear more realistic and display detailed body movements as the actor performs the task. The implementation of new software requires trouble shooting and research to ensure it performs nominally. To ensure performance, the software was thoroughly tested for errors and how it reacts to the current equipment already in the lab.

### C. Overview

The new software required the iSEE lab to learn a number of new grams and software packages, while making sure that each would work cohesively with software that was already used in the lab area. To do this, understanding how the current lab and software reacted to the new program was required. This involved making sure the cameras were calibrated and updating current software.

The primary objective was to ensure the new software would work correctly with the old software. This created many issues, which later required troubleshooting and updating both the new and old software. Once the issues were resolved, a virtual environment had to be created with which an actor could interact with. Using a game engine to create an environment, the actor could walk around in that environment and interact with objects in the simulation.

In order to produce a more realistic simulation, the implementation of new software onto the current lab computer and designing the realistic simulation was the main goal.

## **II. Introduction to the iSEE Lab**

### **D. Layout of the lab**

The iSEE lab is located in the Launch Control Center (LCC) and contains a vast amount of equipment that collects motion capture data for use in creating a simulation. The main equipment consists of markers, cameras, and computers. The markers are small ball like objects that helps the cameras capture data from the actors as they move around. This data will be used to form the bone structure for the actors.

The main room consist of 32 cameras mounted on metal poles throughout the room. The cameras are set up in way to help capture each marker on the actors or objects. The software required to process the simulation runs on three main computers set up in the lab, each with different software.

## **III. Specific Software used in this project**

### **E. Software Description**

On the lab computers, there are five important pieces of software that are beneficial when creating simulations. The first two have been implemented in the lab since the beginning:

Motion Capture interacts with the data coming from the cameras and forms the bones. Bones are used to define the figures anatomy, and once in a simulation allows the user to see exactly what part of the actor is currently moving. These bones are also formed on static objects so they can be constrained to a virtual object.

Analytic Simulation Software creates simulations using data from Motion Capture, and allows the user to place the actor in a virtual environment. It uses the ergonomic data created in the software to figure out if the given task would cause too much strain on a person performing the task in real life. Analytic Simulation Software takes in to account every characteristic of the actor, such as gender, size, and height to determine if the task needs to be performed by a more qualified person.

The next set of software was just implemented this fall, and provides the simulation with a more realistic view. While the new software is not as efficient with ergonomic analysis, it provides a more detailed and realistic capture of real life scenarios.

Game Engine One creates a virtual environment for the actor to view and interact with if they have the correct equipment. The software is normally used as a game engine, and is mainly used by programmers who use C++ and C#. Game Engine One has a variety of free programs and assets that allow the user to create an interactive world efficiently. By using Game Engine One, the iSEE team is able to create a base simulation that is more realistic than Analytic Simulation Software.

Game Engine Two, like Game Engine One, creates virtual environments but is mainly used by visual students. It allows for more plugins thus helping develop the world quicker than Game Engine One. It also implements a form of coding called Blueprint that resembles a spider web. As the coding is linked together, it notifies the player if the code is active. It also provides a realistic simulation, but doesn't have ergonomic features as Analytic Simulation Software does.

Connector is created by the same company as Motion Capture. It helps connect Motion Captures data to a game engine. It does this by taking the bones that are created from Motion Capture and constrains it to a 3D model of a human or object. While the software requires a specific setup, it is needed to connect Motion Capture to the game engine.

Character creator creates 3D humans and 3D objects with premade rigs. Normally rigging and human models could take a few weeks to produce but Character creator creates these models in less than two hours. It also allows the user to alter the 3D model to resemble the captured object's measurements. The software can use multiple export extensions, but primarily uses Filmbox (FBX), an extension required in Connector.

## F. Testing the software's pros and cons

Each simulation software has its own advantages to the development of the lab, table 1 and 2 shows the pros and cons of each of the software that was seen while completing the projects.

Software Pros		
Analytic Simulation Software	Game Engine One	Game Engine Two
Allows for the use of understanding ergonomics when creating a simulation	Allows for slight realistic simulations	Creates detailed and realistic simulations.
UI design of the program allows for the user to connect Motion Capture and Analytic Simulation Software easily	Has multiple free assets for a user to create the environment with	Provides thousands of plugins to allow for the user to create the simulation in an efficient time frame.
Did not require troubleshooting due to the software being implemented when the Motion Capture was first used	Provides vast tutorials built in with the software	Blueprint is easy for the user to learn

**Table 1. Represents the positive traits of each software.**

Software Cons		
Analytic Simulation Software	Game Engine One	Game Engine Two
Simulations are not realistic	Requires extensive amount of coding for an action to run	Main UI design can sometimes be overly complicated to new users
No detailed bone hierarchy on individual areas, such as hands.	Can take a while to place a scene together	Requires updated plugins to efficiently work.

**Table 2. Represents the negative traits of each software.**

The main disadvantage of using Game Engine One and Game Engine Two, was that the new versions had compatibility issues with the older software the lab was using, such as Motion Capture and Connector. To overcome this, the team went through troubleshooting for the software to connect each efficiently without any malfunctions occurring.

## IV. Project Setup

### G. Software compatibility

To create a more realistic simulation the lab performed connection test on the new and old software to make sure it worked properly. This required using Connector as a middle link between both of the game engines and Motion Capture. For Connector to work, you have to make sure the data from the Motion Capture is smooth or it will affect

the quality of the connection to the game engines. The Motion Capture was made to run at 60 frame rates per second (FPS), so it could stay on the same frame rate as Connector. While Game Engine One could handle a higher frame rate, Connector would crash if the FPS was over 60.

## **H. Using 3D Models**

After the software was connected, the lab had to create multiple models to place into Connector and the game engine. Character creator was used to create a character model that had a skeleton attached to it. Character creator would also allow the user to change the body structure of the 3D character to make it resemble the captured actor. Modeling Software was used to create 3D objects for the scene that was created in the game engine. The 3D models had to follow a strict set up as they were created with a low poly count, anything too high could potentially crash the game engine. This required creating the model with simplistic forms and then making subtle details in the end.

## **I. Testing the System**

Small tests were formed to make sure that connecting the software would not create errors in the system. To calculate errors, the lab tested multiple small projects. The first one was to make sure that a human being would work in Game Engine One. After troubleshooting the software, the lab was able to connect an actor into a game engine and watch the avatar walk around the virtual environment. The second project involved setting up a human with an object to work together. This involved having the actor move around a box in real life, and have the feed run smoothly in Game Engine One. To do this, the 3D box had to have a precise bone structure to work in Connector. Using Modeling Software, the lab was able to make the skeleton needed. Both test projects allowed for the user to test the system, and make sure it ran efficiently.

## **J. Using the game engines**

The main game engine that was tested and worked correctly was Game Engine One. It did not have as many issues when it came to connecting to Connector. Game Engine One also allowed for the user to start up a template project that had all the needed documents in it. While Game Engine Two was tested, errors occurred and did not let the lab test its maximum capabilities it obtained. While Game Engine One was well tested, the lab's next step is to connect a virtual reality headset and make sure it connects properly.

# **V. Troubleshooting**

## **K. Resolved Issues**

Connector had previously not been implemented in the lab until it was discovered that realistic simulations could be created. The software not only required you to collect data from Motion Capture, but also have access to a 3D model. To create a 3D model efficiently, objects were created in Modeling software, and humanoid figures were created using Character creator. Each model had to be created with a specific rigging hierarchy for recognition by Connector. At first this caused issues, since any objects created in Modeling Software had to be rigged a certain way for it to be imported into Connector. After researching the Connector documents that the Motion Capture Company provided, the program started to work after the correct rig was implemented.

Game Engine One was the first game engine used in the lab, so it had to go through troubleshooting to efficiently connect to Motion Capture. When Game Engine One was first connected it experienced issues where the actor would rag doll around the game scene. To fix this issue the team had to change the settings of the computer to high performance. This allowed for the processor to handle the amount of data it was receiving and worked smoothly.

## **L. Unresolved issues**

Game Engine Two is having connection issues either due to the bone hierarchy or old plugins. The game characters animation would not move fluently and be stuck in a T-pose. As of now the issue has not been resolved but the team will keep working on it.

Character creator has many standard base models you can use that have a premade rig in them. As the project progressed it was noticed that depending on the character model, specific parts of the 3D model did not always export correctly such as clothes or other details. As with Game Engine Two, the issue has not been resolved but the team will keep working on it.

## **VI. Conclusions**

### **M. New software**

The lab is designed to create simulations that allow the user to see exactly what issues arise as an actor is completing a task. The goal of adding in new programs is to create realistic simulation that can be easier to understand by third party groups. By having it more realistic the team will be able to see specific movements compared to Analytic Simulation Software, such as fluid hand gestures. This also allows the viewer to experience an environment that more closely mimics the real world.

### **N. Future Goals of iSEE**

All unresolved issues with both Game Engine Two and Connector will be addressed. Motion Capture Company will be keeping in contact with the iSEE lab to help resolve any issues. Additionally, testing is being performed for each issue. The importance of the iSEE lab will increase as the new software is integrated into future projects. As technology advances, simulations that could potentially replicate the real world interactions will be implemented. Additionally, the software can also be used to show the user safety hazards before the tasks are officially performed, thus increasing the safety for the user.

## **VII. Acknowledgments**

I would like to thank everyone that supported me in my work, helping me learn valuable information to create a simulation. In addition to the main iSEE team, Mr. Antonio Pego, my mentor, who provided support in the lab as well as provided me with new knowledge and Mr. Deon Williams. There is Mr. Bob Ferrell. He and Deon were there to provide me with important feedback and equipment that I needed to accomplish my objective. Finally, I would like to thank Dr. Gena Henderson for giving me this wonderful opportunity. I have been able to gain so much knowledge since interning at the Kennedy Space Center and enjoyed working with everyone.