

Contents

Figures

Glossary

Bone

In 3D graphics, part of the virtual skeleton used to deform the shape of the object.

Boss

Special game character used in boss battles, often with unique shape, behavior or abilities.

Bounding sphere

Bounding volume in the shape of a sphere.

Bounding volume

Volume, which contains the bounded object.

Collider

Shape which represents the object in collision calculations.

Collision engine

In simulation, system used for detection and resolution of collisions between objects.

Colossus

Colossal creature from the Shadow of the Colossus video game.

Frame

Point in time, when the game state is updated.

Game engine

Software framework used by video game, which contains various subsystems, like graphics rendering, physics calculation, input handling or networking.

Mesh

Data structure, which describes 3D object with vertices, edges, and triangles. Most commonly used representation of 3D objects in video games.

Platforming game

Video game focused to movement mechanics, such as running, jumping or climbing, where the main task of the player character is to overcome various terrain obstacles.

Player character

Character in game, which is controlled by the player.

Polygon

Closed shape defined by multiple points.

Skinning

In 3D graphics, process, where the mesh shape is deformed in relation to the movement of its bones.

Triangle

Polygon formed by three points.

Unity

Modern and advanced game engine commonly used today.

Vertex

Point in three or two dimensional space, which is used to define polygons.

1 Introduction

One of the typical gameplay mechanics in the video games are boss battles. The boss battle is usually a special event where the player character battles against an uncommon enemy. This can be either much stronger enemy with more health and attack power than common enemies or the fight can use special rules and be divided to several phases.

In some games the boss battles are designed as a puzzle. Boss can attack or behave in some predefined pattern and reacts to player actions. Player is expected to study how this mechanics works and to figure from the observation how to defeat the boss. Some requirements have to be respected; for example, the behavior pattern of the boss have to be observable and predictable (to some degree). In bad designs, the gameplay can become too chaotic or random, negating the requirement of the observation of the boss. The observation phase can be simplified by directly provided hints to the player, as is more common in newer games, or indirectly by clues which can be found during the game walk through.

As the game bosses usually differs from the common enemies, one of the recurring characteristic of them is the size difference in comparison to the player character, where the boss can be much larger. Another characteristic can be the amount of its raw power, either in damage output of his weapons or the amount of health.

This is also often the case when the boss fight is designed as a puzzle, so the frontal assault against the boss is not possible or feasible. For example, in the video game Jamestown, the fight against the final boss is divided to two parts. In the first part, the boss alters between a phase when he is vulnerable and a phase when he covers itself in an impenetrable armor and launches an attack against the player character. After he is defeated, he changes the form and second phase began. This phase can be seen in the figure ?? (page ??). The boss is located in the center and location of the player character is in right bottom. In current situation, the boss is invincible. The player have to attack the cubes which floats around the boss. When the cube is damaged, it will collide with the boss and damages him. When this happens the boss retaliates back with a strong attack against the player. In this case, the cubes can be instead used by the player as a shield. The battle ends when the boss health (yellow bar near the top) is depleted or when the player character dies.

As bosses in boss battles can be quite large and their behavior much more complex in comparison to common enemies, they also brings some technical challenges. For example, Nintendo Entertainment System game console allows only limited amount of sprites to be shown at the same time and their maximum size was constrained. To overcome this limitations, the boss was usually drawn using background tiles. Because of this, the actual background of the battle scene was just black color. This can be seen in the figure ?? (page ??), where the boss (green dragon) is quite large in comparison to the player character, the background is black and the blocks on which the player characters stands are actually created using a sprites (D'ANGELO, 2014).

The goal of this thesis is to implement the boss battle mechanics from the video game Shadow of the

Colossus. The boss battles in this game were puzzle based and their implementation marks it as one of the most advanced video games on its original target platform. The thesis will focus on implementation of the technical aspects of Shadow of the Colossus game in the Unity engine. This work is done for the Zaibatsu Interactive Inc.

2 Shadow of the Colossus

2.1 Introduction

Shadow of the Colossus is an action adventure game, released for PlayStation 2 console in 2005. The gameplay is centered around the fights against large beings, known as colossi, with the player character, a human named Wander. Every fight against a colossus can be classified as a boss battle, with the observation mechanics and boss fight phases.

2.2 Gameplay mechanics

Colossi are huge creatures of various forms, usually shaped as humanoids or animals. Their sizes are between 30 to 300 meters in height or length. They are composed of flesh, usually covered with fur, and stone parts, formed in shape of an armor or weapon. Typical look of a colossus and its comparison against the player character can be seen on the figure ?? (page ??). Every colossus has some number of weak points. These points can be stabbed by sword, thus lowering the amount of colossus hit points. Each weak point can absorb only a limited amount of damage. When this amount is used, the weak point will disappear and is either replaced by another in other location, or the colossus will die, if it was the last one. The player can locate an active weak point by using a mini game: when raising his sword a number of light rays will show, concentrating on the direction to the next point. Because of this, some time is usually spent exploring the body of colossus for finding the active weak point.

For approaching these weak points, a player character has to literally climb the colossus, which is due to its massive size. As every colossus is different in shape or behavior, and the battle occurs in different places, the way to climb a colossus changes as well. Moving on the colossus follows the same principles as moving on terrain, and is similar to mechanics used in other platforming games. The player can move on horizontal surfaces, climb over ledges, formed by a colossus's stone parts, or grab on and climb a vertical surfaces covered with colossus's fur or vegetation. One of the unique feature of the game is in the combination of these mechanics with colossus's dynamic, moving body.

A player can freely navigate on colossus restricted just by its shape and availability of fur. As the colossus is moving, passable surfaces can become impassable. The colossus also actively tries to shake the player down. This can be prevented by either standing on appropriate surface, or tightly holding on its fur. The player is limited by the amount of energy used for continuous holding. If this energy is depleted, the player character will release its grip and usually falls down from the colossus. Energy can be replenished by staying on safe (usually horizontal) surfaces (ADAM SIDDIQUI, 2011, .)

2.3 Target platform

During the time of its release, Shadow of the Colossus was exclusive PlayStation 2 game. It was released in the end of the life cycle of this console, therefore the dev...