

скрипта, будет пытаться прийти к тому состоянию, которое было задано в качестве желаемого. Такое поведение системы обусловлено принципом конвергентности, который направлен на приведение системы в нужное состояние при каждом следующем запуске одного и того же скрипта без выполнения операций, успешно выполненных ранее. Т.е. изменения, не выполненные при одной попытке, система будет пытаться выполнить еще раз при повторном запуске.

Список использованных источников:

1. Клеппман. М. Высоконагруженные приложения. Программирование, масштабирование, поддержка/М. Клеппман // СПб: Питер, 2018.
2. Database Migration: What It Is and How to Do It [Электронный ресурс] - Режим доступа: <https://rollout.io/blog/database-migration> - Дата доступа: 22.03.2019

ПРОГРАММНАЯ СИСТЕМА СИМУЛЯЦИИ РАЗРУШАЕМОСТИ НА ОСНОВЕ UNREAL ENGINE 4

Дмитриев А.С.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Золоторевич Л.А., доцент

В данной статье рассматривается одна из возможных реализаций системы симуляции разрушаемости в Unreal Engine 4. Для реализации отдельных компонентов используется внешний плагин Apex Destruction для Unreal Engine и отдельная программа PhysX Lab.

В основе разрабатываемой системы симуляции применяется Unreal Engine 4 как один из наиболее стабильных и производительных движков. При проектировании системы в первую очередь материалы объектов разрушения были разбиты на четыре группы:

- 1) породы – различные каменные образования и структуры;
- 2) металл – тугоплавкие материалы и металлоконструкции;
- 3) дерево – древесина и изделия деревообрабатывающей промышленности;
- 4) стекло – изделия стекольной промышленности и бьющиеся материалы.

Для моделирования разрушаемости объектов первой группы используется алгоритм Воронова, основанный на одноименной диаграмме или так называемом разбиении Дирихле. В качестве его программной реализации используется внешний плагин Apex Destruction для Unreal Engine 4, который заранее делит объекты на сцене на определенное количество элементов, называемых чанками. Все чанки являются параллелепипедами различных размеров, содержащими один из будущих обломков разрушаемого объекта. Так как разделение объекта запекается заранее, разрушаемость является довольно предсказуемой, но с другой стороны позволяет обеспечить высокую скорость моделирования в режиме реального времени (рисунок 1).

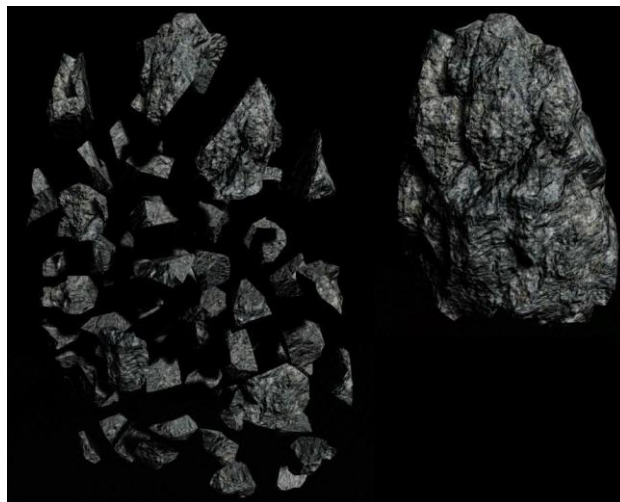


Рисунок 1 – Пример деления объектов в Apex Destruction

Вторая группа основана на трёхмерной разметке модели и её компонентов или так называемом скелетал меше. Каждая такая точка в разметке называется нодом. Положение каждого нода может

быть изменено динамически во время работы программы. Таким образом можно симулировать плавление металлических конструкций путем постепенного смещения узлов и, собственно, сжатия самой модели. Кроме того, при ударах и давлении на объект можно смещением узлов симулировать искривление самой конструкции.

Для моделирования разрушения объекта из дерева используется внешняя программа PhysX Lab, позволяющая делать разметку разрушаемых объектов схожим образом с плагином Arx Destruction, однако используя другой алгоритм. В этом случае задается количество параллельных разрезов по каждой из осей, после чего уже на каждый из кусков накладывается зашумление. Таким образом, получается набор чанков одинакового размера. Все данные по частям декомпозиции хранятся в формате, пригодном для импорта в Unreal Engine 4. При таком делении обломки получаются более вытянутыми и больше похожими на те, которые образуются при разрушении деревянных конструкций (рисунок 2).



Рисунок 2 – Пример деления объектов в PhysX Lab

Моделирование процессов разрушения объектов четвертой группы является наиболее сложным по следующей причине - стекло всегда бьется радиально от точки попадания по всей своей поверхности. Так как точка попадания может быть любой, предусмотреть такое количество заготовленных заранее вариантов невозможно или нерационально. Следовательно, необходимо разработать алгоритм, который способен в реальном времени радиально разбивать модель от точки попадания. Данная задача пока остается нерешенной.

Следующим этапом является объединение предыдущих групп в единые конструкции. Делается это при помощи компонентов в Blueprint-класссах. Каждый Blueprint-класс представляет собой набор визуальных компонентов со встроенной логикой, которая может быть вызвана извне действиями других Blueprint-класссов или событиями на сцене.

Рассмотрим это на примере разрушения железобетонной балки. Если опустить все тонкости, то в центре такой конструкции будет находиться металлический каркас, вокруг которого будут находиться бетонные пластины. При столкновении массивного объекта с объектом данного класса сначала будут разрушаться внешние пластины, а затем уже станет гнуться внутренняя конструкция, причем при сильном изгибе, так как коллизия внутренней конструкции пересечется с коллизией пластин с другой стороны, они также будут разрушены, как это было бы в реальности.

Далее комбинированным компонентам присваивается определенная прочность. Сделано это для того, чтобы симулировать разрушение объектов во времени. Функция изменения прочности имеет вид прямолинейной функции, переходящей в гиперболу. Функция отражает зависимость прочности объекта от времени. Точка перехода между функциями является границей ветхого состояния.

При моделировании разрушения сложных конструкций, например зданий, масса, давящая на объект, извлекается из физической части движка. Оттуда же извлекается и скорость объектов, если она не равна нулю. Также учитывается и ускорение свободного падения. Затем берутся крайние точки соприкосновения объектов, определяется площадь соприкосновения и сила давления на объект.

Когда сила давления немного превышает текущую прочность, объект начинает разрушаться быстрее. При превышении лимита объект теряет прочность и моментально осыпается.

Для визуального отражения старения объектов используются экземпляры материалов. Это копии материалов, позволяющие менять их отдельные параметры. Таким образом, создав экземпляры материалов грязи, плесени и ржавчины, мы можем наложить их поверх текстур объектов сцены и постепенно изменять их степень отрисовки. Выглядит это как постепенное появление грязи, плесени или ржавчины на соответствующих объектах.

Разрабатываемая система может быть использована не только в компьютерных играх, но и при моделировании процессов разрушаемости объектов инженерного типа. Упор в данном подходе сделан на производительность и быстродействие, а не на точность моделей. Однако система может масштабироваться в зависимости от мощности применяемой ЭВМ.

Список использованных источников:

1. Physics Simulation [Электронный ресурс]. — Режим доступа : <https://docs.unrealengine.com/en-us/Engine/Physics> – Дата доступа: 14.03.2019г.
2. Apex [Электронный ресурс]. — Режим доступа : <https://docs.unrealengine.com/en-us/Engine/Physics/Apex> – Дата доступа: 14.03.2019г.
3. Н.Г.Бураго – Институт проблем механики им. А.Ю. Ишлинского РАН, Москва, 119526, Россия – Вычислительная механика сплошных сред. – 2008. – Т. 4, № 4. – С.5-20

ПРОГРАММА ЛОЯЛЬНОСТИ ДЛЯ КЛИЕНТОВ СПОРТИВНЫХ КЛУБОВ НА БАЗЕ ТЕХНОЛОГИИ БЛОКЧЕЙН

Евсаев П.В., Беликова Т.О.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Фролов И.И. – к.т.н., доцент

Исследования рынка клиентской лояльности показывают, что эффективность программ лояльности падает. Это связывают с неудобствами, связанными с хранением большого количества дисконтных карт/установкой большого количества мобильных приложений. Разработка системы, являющейся единой точкой управления программами лояльности для организаций и их клиентов видится своевременным и актуальным.

По прогнозам экспертов, объем рынка клиентской лояльности к 2020 году подойдет к отметке в 48 миллиардов долларов США. Несмотря на внушительный оборот, исследования показывают, что эффективность программ лояльности снижается. Это связывают с тем, что каждая программа лояльности требует использования карты либо мобильного приложения, что снижает привлекательность каждой следующей программы для потребителя. Создание единой системы для запуска программ лояльности снизит поставщикам товаров и услуг издержки на обслуживание собственной CRM системы и позволит иметь единый канал связи с аудиторией потребителей всех организаций-участников сети. С точки зрения пользователя, наличие подобной системы позволит иметь единую точку управления всеми бонусными баллами и акционными предложениями.

При выборе инструментов для разработки проекта, выбор был сделан в пользу стека Hyperledger. Стек Hyperledger позволяет создавать блокчейны с ограничением прав доступа для участников сети, определять собственную модель взаимодействия между участниками сети через строго определенный набор транзакций. Возможность обновления чейнкода дает возможность проводить разработку в итерационном формате, постепенно расширяя функциональности системы. Использование привилегированного блокчейна и обязательный процесс идентификации пользователя снижает риск атаки на сеть и снимает необходимость взимать плату за проведение транзакции.

В соответствии с бизнес-целями, в сети необходимо и достаточно наличие двух типов участников: организация и потребитель. Организации должны иметь право управлять программами лояльности, участники - иметь возможность вступать/выходить из программ лояльности. Оба типа участников должны иметь право осуществлять переводы токенов программ лояльности другим участникам сети.

Процесс запуска программы лояльности может выглядеть следующим образом: при создании организацией программы лояльности, у нее создается кошелек, на который поступает весь объем эмитированных в рамках программы лояльности токенов. Далее, организация высылает клиентам приглашения. Когда потребитель принимает приглашение, у него создается привязанный к этой программе лояльности кошелек, на который/с которого ему могут поступать токены. Токены являются уникальными для программы лояльности и могут быть реализованы либо переданы другому клиенту-участнику программы.

Для доступа к управлению своими ресурсами участникам сети необходимо пройти процесс идентификации. При регистрации, новому участнику сети ему выдается персональный сертификат, выпущенный доверенным для проекта центром выдачи сертификатов. Соответствующая сертификату пара ключей сохраняется локально и будет использована при следующем входе в систему.

В качестве направлений для развития создаваемой системы можно рассмотреть возможность запуска совместных программ лояльности для нескольких организаций, реализовать интеграцию с действующими CRM системами. Наличие гибкого, функционирующего по описанному алгоритму проекта поможет решить существующие проблемы на рынке клиентской лояльности, а использование распределенного реестра для хранения данных об операциях дает хороший фундамент для развития проекта с точки зрения финансового аудита.