

«Сейфуллин окулары – 18: « Жастар және ғылым – болашаққа көзқарас» халықаралық ғылыми -практикалық конференция материалдары = Материалы международной научно-практической конференции «Сейфуллинские чтения – 18: « Молодежь и наука – взгляд в будущее» - 2022.- Т.І, Ч.ІV. - С. 9-11

ИНСТРУМЕНТ ПРОЦЕДУРНОГО МОДЕЛИРОВАНИЯ ДЛЯ AUTODESK MAYA

*Аханова Д.К., магистрант 2 курса
Казахский агротехнический университет им. С. Сейфуллина, г. Нур-Султан.*

Введение.

В индустрии компьютерной графики создание трехмерных сред остается одной из самых трудоемких задач в трехмерном конвейере. Развитие вычислительной мощности часто ускоряет выполнение таких задач, как создание виртуальных наборов, но с этой эффективностью часто борются завышенные ожидания в отношении масштабов производства и визуальной точности. Один из многих методов, с помощью которых компании идут в ногу с этими требованиями — это разработка или приобретение передовых инструментов и программного обеспечения [1]. В случае производства, требующего все более массивных или густонаселенных виртуальных наборов, инструменты процедурного моделирования автоматизируют и ускоряют многие задачи, необходимые художникам по окружению. Виртуальные трехмерные локации часто используются в игровых фильмах, компьютерных функциях, видеоиграх, телешоу и рекламных роликах. Преимущества внедрения таких инструментов применимы к каждой из этих отраслей. Хотя инструменты процедурного моделирования предлагают преимущества множеству разработчиков, их использование по-прежнему в основном ограничивается ускорением дорогостоящего производства, созданного студиями высокого класса. Из-за растянутых сроков и значительных финансовых ресурсов, связанных с этими предприятиями, наиболее рентабельным вложением для студии является разработка специализированных инструментов, которые могут решить многие проблемы, характерные для этих проектов. Такие компании часто обладают командой инженеров-программистов, способных создавать такие инструменты, что приводит к созданию закрытых решений, которые можно без больших затрат модифицировать и повторно использовать для сиквелов или будущих проектов. Несмотря на то, что несколько вариантов процедурного моделирования стали доступны для использования с коммерческим программным обеспечением 3D, включая стандартное приложение Autodesk Maya [2].

В последние годы было разработано больше плагинов для внедрения в Maya фундаментальных функций процедурного моделирования. Одним из таких плагинов является – XGen. С помощью этого подключаемого модуля

объекты размещаются на геометрии с помощью специальных локаторов, называемых «guides», в качестве прокси или путем рисования карт распределения на базовой геометрии («XGen Geometry Instancer»). К сожалению, все ресурсы, назначенные карте распределения, заполняются с одинаковой плотностью друг-друга и равномерно распределены по назначенной поверхности. Это может стать проблематичным, если придется изменить плотность одного актива по сравнению с другим и попытка распределить каждый актив индивидуально приводит к дальнейшим проблемам, поскольку XGen также не обнаруживает и не предотвращает столкновение объектов. Эти ограничения могут привести к неестественному или чрезмерно систематическому размещению сетки, а также к перекрытию ресурсов [3]. Но XGen имеет преимущество размещения и масштабирования объектов с помощью серии математических выражений на основе MEL, встроенного языка сценариев Maya («Vary geometry size»).

ProceduRule был написан в редакторе сценариев Maya с использованием встроенного языка Maya Embedded Language (MEL). Это решение было принято, чтобы ускорить разработку и отладку, поскольку Autodesk предлагает дополнительную документацию и поддержку для сценариев MEL. Разработка с использованием MEL в редакторе сценариев значительно сокращает время исследования и позволяет ускорить итерацию кода. Альтернативным подходом к созданию инструмента для Autodesk Maya было бы создание сценария с использованием Python. Maya поддерживает три разные оболочки Python, каждая из которых имеет свой набор преимуществ и недостатков. Первый из них, maya.cmds, позволяет пользователю напрямую вызывать большинство команд MEL с помощью Python [4].

PyMel имеет тенденцию быстро оценивать при выполнении базовых операций, но требует гораздо больше времени, когда он создает экземпляры предопределенных классов PyMel. Поскольку этот экземпляр происходит при выполнении многих функций, PyMel часто может работать в целом медленнее, чем MEL или maya.cmds. Преимущества PyMel наиболее очевидны в его объектно-ориентированном дизайне и организованном синтаксисе, которые делают его самой простой в использовании оболочкой Python. Объектно-ориентированное программирование позволяет PyMel выполнять аналогичные задачи с меньшим количеством строк кода, что делает его наиболее организованным и читаемым вариантом. Оставшаяся оболочка Python - это OpenMaya, API C ++, обычно используемый для разработки подключаемых модулей и высокопроизводительных расширений программного обеспечения Maya. Хотя эта оболочка оценивает и выполняет намного быстрее, чем любой из предыдущих вариантов, такая скорость достигается за счет времени разработки. OpenMaya - это технический язык низкого уровня, который обычно требует значительно большего количества кода для разработки той же функциональности, которую некоторые команды MEL инкапсулируют в одну строку. Как и maya.cmds, OpenMaya пытается смешать Python с другим языком кодирования, что приводит к громоздкому

синтаксису, который может еще больше усложнить процесс разработки («PyMEL для Maya»). Эта альтернатива, как и другие оболочки Python, предлагает меньше встроенной поддержки и документации, чем MEL. OpenMaya отлично подходит для самых технических проектов, но также занимает больше всего времени и труднее всего использовать эффективно [5].

В заключении, ProceduRule демонстрирует, что продуманные инструменты автоматизации могут успешно удовлетворить потребности небольших студий, которым не хватает ресурсов более крупных студий. Эти утилиты позволяют создавать массивные, густонаселенные миры в разумные сроки, что может означать разницу между жизнеспособностью проекта или невозможностью. Расширяя возможности того, что может быть выполнено небольшими студиями или в рамках более жестких графиков, возможности индустрии компьютерной графики также расширяются, особенно в коммерческой или вещательной среде, где временные ограничения могут резко ограничить объем или качество проекта. ProceduRule помогает продемонстрировать эффективность работы с процедурно сгенерированным контентом и то, как инструменты автоматизации позволяют 3D – студиям выходить за рамки границ.

Список используемой литературы

1. Aloys Baillet, Eoin Murphy, Oliver Dunn, and Miguel Gao. Forging a new animation pipeline with usd, pages 54:1–54:2, New York, NY, USA, 2018.
2. Jankowski J. Advances in Interaction with 3D Environments, 2015, no. 1, pp. 152-190. ISSN 0167-7055
3. "PyMEL for Maya." Autodesk Knowledge Network. Autodesk, 2009. Web. 19 Jan. 2017.
4. "SideFX Releases Houdini 15.5." SideFX, Side Effects Software, 19 May 2016. Web. 24 Jan. 2017. Houser, Radek 'ElDonaldo'. "Performance Comparison: MEL vs PyMel vs OpenMaya vs Maya.cmds vs C++." Cautiously Optimistic. Blogspot, 17 Aug. 2012. Web. 19 Jan. 2017.
5. Pymel documentation: pymel.core.modeling.polymovevertex. Retrieved from: <https://help.autodesk.com/cloudhelp/2018/JPN/Maya-Tech-Docs/PyMel/generated/functions/pymel.core.modeling/pymel.core.modelin.polyMoveVertex.html>, note=Last Accessed: 2019-05-19.