
**НОВЫЕ ТЕХНОЛОГИИ
В МАШИНОСТРОЕНИИ**

УДК 004

**РАЗРАБОТКА ЦИФРОВОГО КОНЦЕПТА
НА ОСНОВЕ ТВЕРДОТЕЛЬНЫХ 3D-ОБЪЕКТОВ,
ОБЕСПЕЧИВАЮЩЕГО КОМФОРТНЫЕ УСЛОВИЯ РАБОТЫ
ИНЖЕНЕРА-КОНСТРУКТОРА**© 2023 г. И. Д. Долгий¹, А. М. Ляшенко¹, Д. В. Глазунов^{1,*}¹*Ростовский государственный университет путей сообщения, Ростов-на-Дону, Россия***e-mail: glazunovdm@yandex.ru*

Поступила в редакцию 23.06.2022 г.

После доработки 18.11.2022 г.

Принята к публикации 20.12.2022 г.

В статье приводится описание процесса создания трехмерных моделей на основе метода AAA-пайплайна. Установлено, что старт построения проекта и любой модели начинается с идеи, являющейся интерпретацией желания заказчика, которую необходимо предоставить клиентам путем выполнения поставленных задач и требований, как для целой команды, так и для отдельно взятого разработчика. Предложена инвариантная метрика сравнения изображений. Рассмотрен новейший процесс разработки в IT-сфере – AAA-пайплайн. Приведено поэтапное применение процесса AAA-пайплайна для создания твердотельных 3D-объектов, включающее восемь основных этапов выполнения пайплайна. Проведена разработка моделей цифрового концепта, обеспечивающего комфортные условия работы инженера-конструктора. Выполнено экспортирование созданных 3D-моделей в иной графический редактор.

Ключевые слова: контент, цифровизация, процесс, ресурс, разработка, модель, стиль, карта, анимация, AAA-пайплайн, оптимизация, графика, полигон, блокинг

DOI: 10.31857/S0235711923020049, **EDN:** COYDMI

Современное состояние рассматриваемой проблемы. При разработке проекта планировки пространства очень важно учесть основные факторы: количество сотрудников, схему развития организации, показатели роста, потребности во вспомогательных помещениях, которые сегодня не ограничиваются только комнатами для переговоров. В настоящее время 30–35% конструкторских пространств в организациях не заняты рабочими местами. Как следствие, зарождаются некие хабы – зоны, где инженеры-конструкторы могут пересекаться. Намечается тенденция поощрять общение и взаимодействие среди конструкторов, что очень важно для определения местоположения подобных зон при разработке планировки конструкторского пространства. Универсальная планировка пространств снижает управленческие расходы. Поэтому хорошо продуманное решение о расположении зон в пространстве, о порядке управления пространствами становится важным для таких организаций как конструкторские бюро.

Одним из новейших технологий разработки цифрового концепта, например, конструкторских пространств, является применение процесса AAA-пайплайна для создания твердотельных 3D-объектов. AAA-пайплайн получил свое название от трех латинских букв “А”, каждая буква имеет информационную важность [1–7]. Соответственно, расшифровка аббревиатуры имеет следующий вид: “A lot of time (большое количество

затраченного времени со стороны разработчиков), A lot of resources (множество ресурсов, которые были использованы в процессе разработки), A lot of money (сумма инвестированных денег со стороны спонсорских контор)” [8–14].

Этот процесс является настоящим технологическим прогрессом, благодаря которому проведено оптимизирование процесса создания 3D-моделей, т.к. ускоряются темпы при работе со следующими технически важными вопросами: выбор стиля моделей, подсчет количества полигонов в итоговых моделях; определение необходимости задеирования карты нормалей (normal map); выбор использования “плоских” цветов или переход на современные физически корректные материалы (PBR); определение необходимости использования карт с бликами; выбор разрешения текстур в UV-пространстве (измеряемого в текселях); определение разрешения текстурных карт (измеряемого в пикселях); необходимость смены положения объектов с заднего на передний план и, наоборот, для каждой модели в зависимости от ракурса, который представит перед зрителем; оценка необходимости создания анимации каждой модели; выбор графического редактора для создания предварительных текстурных карт (“запечка” нормалей); оценка необходимости детализации частей объектов или переход на “визуальный обман”, который реализуется в текстурном графическом редакторе; предварительное конвертирование каждого полигона в соединение двух треугольников; определение необходимости задеирования карты прозрачности; оценка необходимости использования “атласов” (использование одной текстурной карты для нескольких объектов); выбор расширения при создании итоговых текстурных карт; создание драфтовых моделей, находящихся на большом расстоянии (“лоды”); оценка необходимости использования геометрии, которая работает с физикой и ее просчетом (“коллижн”).

Таким образом, выполнение пайплайна позволяет создать красивую, качественную, а главное грамотно оптимизированную под графический движок (Unity, Unreal Engine и др.) 3D-модель.

Все этапы пайплайна основываются на соблюдении четырех функций: 1) последовательность — выполнение работы строго пошагово. Разработчик не имеет права нарушать структуры пайплайна, т.к. это приведет к замедлению рабочего процесса и даже к некорректной финальной версии 3D-модели; 2) вариативность — смена некоторых этапов разработки в зависимости от рабочего проекта. Например, разница в использовании технического оборудования при моделировании (для твердотельных Maya 3D или 3D Max, а для мягкотелых Zbrush или Modo); 3) завершенность — невыполнение одного этапа ведет к итоговому провалу. Например, отсутствие корректной UV-развертки не позволит создать разноцветные текстурные карты для модели; 4) универсальность — оптимизация всего этапа разработки под любой набор программного обеспечения (ПО).

Следовательно, использование различных графических редакторов позволит получить итоговый продукт. С целью создания твердотельных 3D-объектов с использованием процесса AAA-пайплайна, необходимо выбрать способ сравнения изображений.

Выбор способа сравнения изображений. Алгоритм нахождения чистой ячейки основан на предположении, что чистые ячейки похожи между собой сильнее, чем ячейки, загороженные объектами переднего плана. Так как существующие способы сравнения изображений не позволяют надежно определять похожие ячейки, была предложена новая метрика. Чтобы сделать метрику инвариантной к изменениям в освещении сравниваются не сами изображения, а градиенты на них. Для компенсации возможных небольших несоответствий в положении ячеек решеток предлагается размывать найденные на изображении края с помощью преобразования расстояний (distance transform). Для каждого пикселя изображения записывается расстояние до ближайшего к нему края. Сравнение полученных изображений ведется попиксельно как сумма

квадратов разностей. Чистой ячейкой объявляется та, сумма расстояний от которой до других по метрике минимальна.

Задачу сегментации можно сформулировать как необходимость присвоения каждому пикселю каждой ячейки метки объекта или метки объектов переднего плана [15]. Для сегментации используется модель Марковского случайного поля (MRF) — графическая модель, состоящая из набора случайных величин и связей между ними. Всем пикселям изображения ставятся в соответствии случайные величины, принимающие значения цветов этих пикселей, и неизвестные случайные величины, соответствующие метке пикселя. Проставляются связи, показывающие непосредственно зависящие друг от друга пиксели. Необходимо максимизировать общую апостериорную вероятность модели за счет выбора значений меток. Задачу максимизации вероятности можно представить как задачу минимизации энергии на графе. Вершины такого графа соответствуют пикселям изображения. Каждой вершине сопоставляется метка. Ребра графа строятся между пикселями, связанными друг с другом в модели. Функция энергии состоит из двух видов слагаемых: бинарных, которые формулируются для каждой связи между пикселями, и унарных, которые формулируются для каждого пикселя

$$E = \sum_p E_p(p, l) + \sum_{p, q} E_b(p, q, l_p, l_q),$$

где p, q — различные пиксели; l — метка, которая может принимать бинарное значение “объект” или “объект переднего плана”.

Используемая функция энергии удовлетворяет условиям регулярности и может быть минимизирована алгоритмом разреза графа. Граф строится для каждой имеющейся ячейки. Значения бинарных слагаемых для соседних точек отражают тот факт, что близкие по значению пиксели, скорее всего, относятся к одному классу. В случае различных меток пикселей энергия зависит от цвета соседних пикселей и обратно пропорциональна их похожести. В случае одинаковых меток энергия равна нулю

$$E_b(p, q, l_p, l_q) = \begin{cases} \exp(-\|I(p) - I(q)\|), & l_p \neq l_q, \\ 0, & l_p = l_q. \end{cases}$$

Унарные слагаемые отражают вероятности пикселей быть частью объекта или частью объектов переднего плана. Известна только одна чистая ячейка, соответственно можно считать, что чем ближе пиксель по цвету к соответствующему пикселю чистой ячейки, тем больше у него шансов быть частью объекта. При метке “объект переднего плана” значение унарного слагаемого является просто некоторой константой

$$E_u(p, l) = \begin{cases} \exp(-\lambda * \|I(p) - I_{\text{best}}(p)\|), & l = \text{facade}, \\ \exp(-\lambda * C), & l = \text{foreground}, \end{cases}$$

где I_{best} — абсолютно чистая ячейка; C — экспериментально подобранная константа. Единственным параметром системы, который показывает важность бинарных слагаемых по сравнению с унарными, остается параметр λ . Изменяя только его, можно добиться приемлемого результата. Единого значения λ для всех случаев не существует. Это обусловлено различиями в освещенности ячеек и характере текстуры. Для восстановления текстуры в найденных областях в них попиксельно копируется чистая ячейка. Для компенсации различий в освещенности ячеек и неточности нахождения решеток, применяется метод вставки части одного изображения в другое, основанный на построении и решении уравнения Пуассона с краевыми условиями Дирихле. Метод учитывает перепады цвета во вставляемой части и краевые условия векторного поля градиентов из целевого изображения. Далее рассмотрим поэтапное применение процесса AAA-пайплайна для создания твердотельных 3D-объектов.

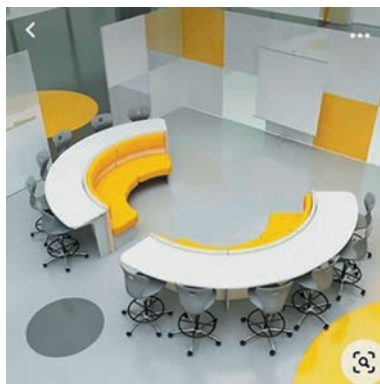


Рис. 1. Референс нового формата для учебной аудитории образовательного кампуса.

Поэтапное применение процесса AAA-пайплайна для создания твердотельных 3D-объектов. Успешное выполнение пайплайна основывается на восьми основных этапах: 1) взаимодействие с референсами (вспомогательными изображениями); 2) создание силуэта и основных форм объекта (“Драфт”); 3) детализация 3D-модели; 4) создание многомиллионной модели и малополигонального аналога (“Сетка”); 5) создание 2D-аналога 3D-объекта в UV-пространстве (“Развертка”); 6) перенос элементов с многомиллионной модели на малополигональный аналог (“Запечка”); 7) создание текстурных карт (“Текстурирование”); 8) создание рендеров со всех необходимых сторон и углов (“Визуализация”).

Первым этапом процесса разработки является взаимодействие с референсами (вспомогательные изображения), которые не только значительно сокращают время работы разработчиков, но и напрямую выступают в качестве необходимого инструмента для того, чтобы работа была выполнена корректно. Референсы оптимизируют работу на всех этапах разработки, т.к. выступают наглядным примером того, какой продукт необходимо получить в конечном итоге. Например, в строительной сфере люди используют эскизы, технические рисунки и чертежи. В IT-сфере референсы применяются повсеместно в Web-дизайне, при обработке баз данных, соответственно в 3D и др. В качестве референса может выступать абсолютно любое изображение или его часть, которое позволяет увеличить темпы разработки.

В 3D-сфере референсы позволяют определить внешний вид объекта, составные динамические и статические части, пересекающиеся и зависимые элементы и что крайне важно, целые детали, которые не требуется моделировать из-за их малых габаритов или отсутствия информационной значимости со стороны пользователей (например, в киноиндустрии — проезжая часть всегда только из верхнего слоя асфальта, т.к. зрители никогда не увидят песок и гравий, которые необходимы для структурного состояния дороги). Процесс получения референсов напрямую зависит от задач, стоящих перед командой или разработчиком, т.е., если задача состоит в создании стилистически точного объекта, то набор изображений будет состоять из фотографий необходимой архитектурной единицы со всех ракурсов, которые сделаны в реальном мире. Это позволит определить габариты всех составных частей объекта. Если цель проекта состоит в создании чего-то совершенно нового, не имеющего аналогов в окружающем мире, то получение референсов будет зависеть от творческих возможностей художников.

Таким образом, получить референс можно следующим образом: поиск в сети, создание фотографий в реальном мире и выполнение художником (рис. 1).



Рис. 2. Демонстрация внешнего вида модели офисного кресла созданного на основе блокинга.

Второй этап разработки включает создание общей формы объекта, иными словами, создается силуэт 3D-модели. Изначально все объекты в сцене имеют угловатую, собирательную форму. Такой формат упрощенных моделей получил название драфт. Соответственно вся работа на данном этапе выполняется по методу блокинга, это позволяет создать будущий некий шаблон каждой модели в кратчайшие сроки относительно всего проекта. На данном этапе разработчики используют только крупные и средние формы стандартных объектов (кубы, цилиндры и сферы) для того, чтобы попасть в стилистику и пропорции перед переходом к обработке деталей. Такой набор примитивов создает силуэт объекта, который передает суть модели. Блокинг позволяет создать читаемую и информационно понятную модель, используя простые и примитивные объекты. Таким образом, это работа с крупными моделями, позволяющими в своем сочетании воплотить соответствие пропорций и эффективный силуэт. Этап занимает в среднем 30–60 мин, что является чрезвычайно малым временным промежутком в 3D-сфере. На данном этапе разработчик беспокоится лишь о внешнем виде объектов, т.к. модели выполняют функцию образца — некая 3D-интерпретация 2D-референса. В итоге многие полигоны имеют низкое качество, т.е. неправильное количество ребер, вершин, а также нормали зачастую имеют некорректное направление. Выполнение булевых операций происходит без соблюдения необходимых правил, что приводит к математически неправильной 2D-сетке моделей. Объяснение этого основывается на продолжительности данного этапа — один рабочий час разработчик затрачивает на создание информационно читаемого силуэта, иначе нарушается последовательность AAA-пайплайна. В свою очередь на данном этапе модель зачастую обладает большим количеством однотонных текстурных материалов, позволяющих заранее продумать какую цветовую палитру использовать при создании текстурных карт, чтоб все необходимые акценты объекта имели презентабельный внешний вид.

Выполнение второго этапа позволяет получить не готовую модель, а лишь ее детальный образец, позволяющий зрителю определить функциональность и назначение объектов. Некорректное выполнение этапа приведет к сильным затруднениям в дальнейшей разработке, т.к. формы драфта являются прямым эскизом для 3D-разработчика, который необходимо выполнить с абсолютной точностью. Соответственно, проявление ответственности в процессе блокинга позволяет создать финальную модель достоверно точной и интересной для зрителя. На следующих этапах разработки модель будет лишь наполняться составными малыми элементами, но общие внешние формы будут неизменны (рис. 2).

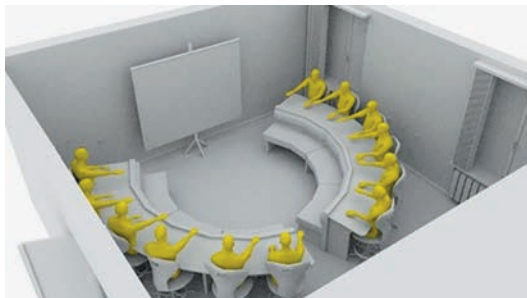


Рис. 3. Коллаборация low-poly (интерьер) и high-poly (студенты) моделей при моделировании нового формата аудитории.

Успешное завершение второго этапа основывается на выполнении следующих правил: итоговый объект понятен для зрителя; силуэт объекта понятен и информационно читаем; соблюдены пропорции, основанные на предоставленном референсе; модель обладает уникальной читаемой функциональностью; объект обладает всеми необходимыми структурными элементами, грамотное сочетание однотонных цветовых материалов, визуальное принятие со стороны разработчика. Этот этап пайплайна в плане творческого подхода и ответственности со стороны разработчика является вторым по сложности после создания текстурных карт, т.к. остальная работа основывается на соблюдении технических и математических правил.

Третий этап разработки 3D-моделей называется детализацией, который обозначает механику движения модели, позволяющую зрителю определить функциональность данного объекта. Необходимо грамотно проработать пересечения и геометрические переходы, отвечающие за смысловые участки детали. Это позволяет придать 3D-модели визуальную интересность и сложность.

Четвертый этап разработки основывается на моделировании малополигональной модели (low-poly) и ее многомиллионной версии (high-poly). Переход к данному этапу нельзя совершить без создания грамотного силуэта и геометрических форм разрабатываемого объекта. В редких ситуациях создается модель средней сложности (mid-poly), в основном используемая в киноиндустрии, т.к. в этой сфере нет высокого приоритета оптимизации 3D-объектов из-за того, что киностудия обладает большими ресурсами, которые позволяют предоставить разработчикам новейшее ПО. Low-poly — низкополигональная модель, отвечающая за силуэт объекта, его блик от светового прибора. Основная задача этого формата заключается в экспортировании в графический редактор текстурных карт и игровой движок, поэтому разработчик создает UV-развертку именно на этой модели. Оптимизация low-poly (современная игровая модель имеет около 20–40 тысяч полигонов) реализуется путем удаления ненужных полигонов или даже целых геометрически отдельных частей (shell). В основном, ненужный полигон это тот, который никогда не увидит зритель. Также в модели сокращается разрядность (например, зритель не сможет отличить 25- и 20-гранник). Главная задача такой модели — полигональная оптимизация и визуальная выразительность с сохранением важных геометрических частей, т.к. именно на них в процессе “запечки” будут перенесены элементы с многомиллионного аналога. High-poly — максимально детализированный формат 3D-модели (количество полигонов модели измеряется в десятках миллионов), задача которого исключительно внешний вид, который будет перенесен на low-poly в процессе “запечки”. Соответственно, отсутствует ограничение в количестве полигонов, т.к. данная модель не обладает корректной UV-разверткой. Единственное что необходимо учитывать — это возможность

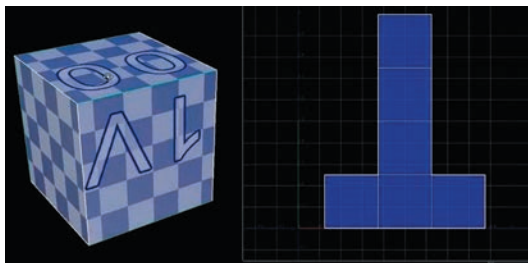


Рис. 4. Стандартная UV-развертка куба.



Рис. 5. Демонстрация внешнего вида low-poly модели офисного кресла с картой нормалей и сохраненной топологией.

ПО для обработки данного объекта. Задача high-poly основывается на сглаживании смежных граней, придании округлости фактурам и важным деталям, а также определении положения блика составных частей. Из-за того, что такая модель является дополнительной, разработчик вправе проявить весь творческий потенциал при ее создании (рис. 3).

Последовательность создания high-poly и low-poly напрямую зависит от физических качеств модели. Твердотельные (здания, мебель, авто и др.) объекты моделируются с low-poly, мягкотелые (флора и фауна) с high-poly, а модели, которые будут созданы исключительно для рендера обладают и вовсе только одним форматом.

На *пятом этапе* разработки создается UV-развертка 3D-модели в 2D-пространстве. Успешное выполнение этого необходимого и важного процесса позволит провести “запечку” и создать текстурные карты, т.к. графические редакторы работают исключительно с картинками, которые являются двухмерной интерпретацией объектов. Задача разработчика на данном этапе заключается в нарезке модели на составные части по одному математическому правилу: если угол пересечения смежных граней меньше, либо равен 90° , то совершается разрез. Далее необходимо сделать развертку каждой детали модели по единому разрешению, измеряемому в текселях. Оптимизация в UV-пространстве реализуется путем наложения (overlap) друг на друга повторяющихся или идентичных объектов (например, левый и правый глаз), а также путем использования корректного расстояния между элементами модели (padding) (рис. 4).

Шестой этап пайплайна основывается на “запечке”, т.е. на создании предварительных текстурных карт. Это осуществляется путем переноса важных визуальных

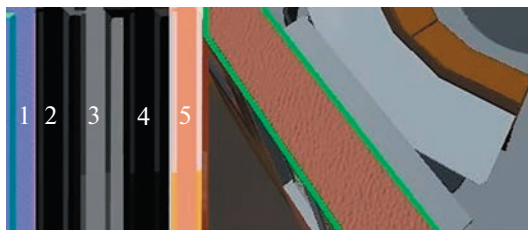


Рис. 6. Визуальное отображение участка стены учебной аудитории при использовании всех необходимых текстурных карт.

элементов с high-poly на low-poly без изменения количества полигонов и UV-развертки. То есть, происходит “визуальный обман”, т.к. низкополигональная модель теперь обладает такими же сглаженными фасками, фактурой, силуэтом, бликом, углублениями как у многомиллионного аналога благодаря карте нормалей (появилась в 2001 г. с выходом киноленты “Властелин колец”) (рис. 5).

Благодаря Normal map при текстурировании low-poly модель на рендере будет иметь внешний вид ничем не отличимый от high-poly.

На *седьмом этапе* AAA-пайплайна происходит текстурирование низкополигональной модели. Данная часть работы является самой сложной с точки зрения творческих возможностей разработчика. 3D-специалист должен обладать художественными навыками (например, иметь понятие о грамотном сочетании цветов). Существуют различные способы визуализации картинок. При создании текстурных карт, которые будут использованы в мультфильме, используется чистый цвет, т.е. без карт шероховатостей (Roughness) и металла (Metallic). Модели компьютерных игр конца 90-х и начала 2000-х годов обладали текстурами с вложенными картами бликов, света, отражений и теней для оптимизации при экспорте в игровой движок. Для моделей в AAA-проектах используют набор текстур, который состоит из карт цвета (Diffuse), высоты и глубины (Height), нормалей (Normal), бликов и затенений (Specular), гладкости и шершавости (Roughness), металла и пластмасса (Metallic), отражения и прозрачности (Opacity), а также свечения (Emission) (рис. 6).

Таким образом, для создания концепта штукатурки стены аудитории необходимо использовать следующие текстурные карты: 1) карта нормалей — задача заключается в визуальном отображении углублений на поверхности стены; 2) карта металла — инвертированная карта позволяет создать шершавый внешний вид; 3) карта высоты — использование данной текстуры позволяет создать воздушные возвышенности штукатурки; 4) карта освещенности — отвечает за отсутствие блика и придание тусклого внешнего вида; 5) карта цвета — задача заключается в окончательном визуальном отображении участка модели.

Восьмой этап AAA-пайплайна является основным, т.к. происходит визуализация всех этапов проделанной работы. Успех на данном этапе зависит не только от выполнения предыдущих шагов, но и от грамотного выбора ракурса, заднего фона и количества яркости и затенения вокруг модели (рис. 7).

Цифровое отображение позволяет придать моделям средней сложности весьма презентабельный внешний вид на основе коллаборации всех текстурных карт. Недостаток состоит в необходимости высококачественного ПО, а также длительности создания рендеров (от десятка минут до нескольких часов). Выполнение всех нюансов на данном этапе работы позволяет предоставить заказчику созданную модель в более выгодной форме с точки зрения визуального отображения.



Рис. 7. Внешний вид нового формата аудитории на основе процесса AAA-пайплайна.

Оптимизация моделей для использования в среде разработки. Многომиллионные модели с подробной, точечной детализацией применяются в тех случаях, когда 3D-специалисты разрабатывают объекты под высокопроизводительное техническое оборудование. Также такие объекты создаются для визуального интерпретирования, т.е. выступают в качестве итогового продукта, который не будет использован в приложениях. В основном разработчики применяют малополигональные аналоги, над которыми перед применением проводится ряд необходимых последовательных действий, позволяющих оптимизировать 3D-модели. Выполнение данных операций позволяет создать графически упрощенные объекты, которые будут применены в программе. Тем самым реализуется меньшая нагрузка на ПО, что приводит к снижению требований к аппаратной части.

Набор наиболее популярных методов по оптимизации моделей: уменьшение общего количества числа полигонов в каждом составном элементе; уменьшение количества используемых и применяемых текстурных материалов; удаление всей истории обработки каждой составной части 3D-модели; обнуление всех модификационных компонентов, которые влияют на внешний вид модели; редактирование участков UV-развертки путем применения наложения элементов друг на друга (overlap); сокращение межэлементного пространства на UV-развертке путем использования корректного расстояния (padding); использование одинаковой текстурной карты для пересекающихся и похожих моделей (“атлас”); снижение уровня качества материалов в UV-пространстве, измеряемого в текселях; удаление тех граней объекта, которые обладают малым информационным приоритетом для зрителя; переход на использование примитивных частей (кубы, сферы) малых элементов в плане габаритов; переход на более низкий уровень качества текстурных карт, измеряемый в пикселях. Количество полигонов играет важную роль, значительно влияющую на объем затрачиваемых ресурсов, со стороны технического оборудования, требуемого для выполнения каждого этапа пайплайна, а также для визуализации всей работы в цифровой сцене. Это особенно важно в тех случаях, когда в одном пространстве находятся много созданных 3D-моделей. В большинстве случаев объекты содержат очень много элементов для фонового пространства, которые являются лишними с точки зрения видимости зрителя, а также имеют скрытые грани, которые разработчик упрощает путем объединения без влияния на визуализацию силуэта деталей.

Оптимизация модели реализуется путем использования меньшего количества материалов, что увеличивает процесс текстурирования, но уменьшает объем файла, кото-

рый измеряется в мегабайтах. Очистка истории процесса создания и обнуление всех модификационных компонентов (move, translate, rotate) позволяет сократить время обработки модели со стороны персонального компьютера. Сокращение общего веса модели, который зависит от текстур в UV-пространстве, проходит с соблюдением следующих правил: расположение одинаковых, зеркальных и повторяющихся частей друг на друге, максимальное сокращение межэлементного пространства, а также применение минимально возможного качества, измеряемого в текселях. Обеспечение меньшей нагрузки на ПО со стороны моделей при текстурировании реализуется путем отказа от использования всего набора текстурных карт. Например, в качестве аналога карты отражения выступает прорисовка бликов поверх основных цветов. Также разработчик использует минимально допустимое разрешение текстурных карт. Например, при работе с мультипликацией используется качество текстур равное 512×512 , т.к. применяются однотонные цвета.

Таким образом, соблюдение правил оптимизации при создании текстурных карт позволяет создать виртуальную визуализацию пола, которая имеет один полигон. Оптимизация модели проводится 3D-специалистом вручную, либо с использованием набора имеющегося функционала и дополнительного инструментария графических редакторов и стандартных программ.

Экспортирование моделей в графический редактор. Непосредственно перед выполнением экспортирования созданных 3D-моделей в иной графический редактор необходимо провести ряд операций. Изначально проводится проверка наличия очищенной истории каждого элемента, а также обнуление всех модификационных компонентов, с помощью метода “заморозки” (“freeze components”). Выполнение данных правил позволяет обеспечить корректное положение и направление трех осей для управляющих элементов “скелета” модели, а также расположить модель в правильных, необходимых координатах в графическом движке. Затем проверяется масштаб модели, т.к. графический движок по стандарту имеет систему, измеряемую в метрах, а трехмерные редакторы в свою очередь обладают сантиметровой. Поэтому необходимо заранее сократить увеличить 3D-модель путем несложной математической операции, чтобы сократить время работы над моделью. Далее проводится обработка всех имеющихся артефактов (темные, некорректные затенения на объекте, которые появляются из-за неправильного положения нормалей), ошибок блика и освещения. Затем происходит распределение по группам в зависимости от назначения и иерархической зависимости. Например, приоритет коленного сустава всегда выше голеностопного. Это позволяет избежать ошибок на следующих этапах работы, таких как: ригинг (создание иерархического опорного “скелета” 3D-модели, обладающего удобными элементами управления для дальнейшего анимирования), скининг (перенос модели на скелетный каркас путем использования некоторых математических операций) и анимация.

При экспортировании 3D-модели, имеющей анимацию и скелетный каркас, необходимо указать в окне экспортирования ключевые кости и анимационные клипы для того, чтобы иметь возможность их использования. Также при экспорте требуется учитывать выделение всех составных частей объекта в иерархической системе, чтобы избежать фатальных ситуаций: потеря кусков объекта и некорректная обработка анимационных клипов. Перенос анимации из рабочего графического редактора Maya 3D проводится путем использования специальной утилиты “Game Exporter”, которая позволяет кадрowo разделить весь анимационный фильм на именные сюжеты и экспортировать каждый независимо друг от друга в необходимом формате. Непосредственно при экспорте необходимо выбрать формат файла 3D-модели, который напрямую отвечает за чтение другими системами и редакторами, обрабатывающими полученный объект. Наиболее популярный формат, которым обладают модели – это fbx, т.к. он позволяет беспрепятственно переносить 3D-объекты, 2D-объекты с текстурной картой высоты, имеющимися источниками света, настроенными камерами и

уникальными материалами между всеми популярными системами и графическими редакторами. Данный формат обладает всеми необходимыми видами аспектов модели: собственной геометрией, набором материалов, текстурами, группой анимационных клипов, ограничителями по атрибутам и цепочками инверсивной кинематики, а также настроенным освещением. Fbx-формат постоянно оснащается и модернизируется со стороны компании Autodesk, которая оптимизирует функционал, тем самым реализует уменьшение объема итогового файла. В открытом доступе имеется бесплатный набор средств разработки (SDK). Недостаток формата заключается в закрытом исходном коде, который позволяет компании-владельцу обладать некой монополией и тем самым не позволять сторонним разработчикам редактировать функционал собственноручно.

Наиболее популярные ошибки при экспорте моделей: наличие у 3D-модели лишних полигонов и ребер, сгенерированных отдельно от основного объекта, но которые приводят к критическим ошибкам при математических расчетах со стороны ПО; наличие некорректной UV-развертки и полное отсутствие шаблонного аналога; отличие алгоритмов созданной UV-развертки от используемой в конечной программе; некорректное разделение анимационного фильма на клипы, что приводит к неправильному отображению анимации или к конвертированию в отдельные объекты; использование специфического формата итоговой 3D-модели, который не обрабатывается со стороны конечного ПО; использование автоматических математических и физических операций, которые увеличивают темп работы, но обладают риском создания некорректной сетки 3D-объектов.

Таким образом, процесс экспортирования играет значительную роль с точки зрения рабочего процесса, а также напрямую зависит от используемого ПО и знаний правил со стороны разработчика. Корректное выполнение данного этапа позволяет совершить перенос 3D-модели из одного графического редактора в другой для дальнейшей работы или предоставления конечному пользователю.

Обсуждение результатов в научном и прикладном аспектах. В настоящей статье определен и подробно описан поэтапный, современный способ создания цифровых аналогов объектов реального мира, который в IT-сфере имеет название AAA-пайплайн. Выбранный процесс позволяет реализовать любой дизайнерский проект по всем правилам и требованиям нынешнего выполнения 3D-моделей.

Разработаны необходимые методы по оптимизации объектов для графических редакторов, снижающие нагрузку на ПО, а также описаны правила по корректному экспортированию, позволяющие избежать фатальных ошибок на всех этапах работы.

Выбран набор необходимых методов для выполнения концептов нового формата аудиторий и коридора любых сложно архитектурных зданий.

КОНФЛИКТ ИНТЕРЕСОВ

Авторы заявляют, что у них нет конфликта интересов.

СПИСОК ЛИТЕРАТУРЫ

1. Алиева Э.Ф., Алексеева А.С., Ванданова Э.Л., Карташова Е.В., Резапкина Г.В. Цифровая переподготовка: обучение руководителей образовательных организаций // Образовательная политика. 2020. № 1 (81). С. 54.
2. Антонова Д.А., Оспенникова Е.В., Спирин Е.В. Цифровая трансформация системы образования. Проектирование ресурсов для современной цифровой учебной среды как одно из ее основных направлений // Вестник Пермского государственного гуманитарно-педагогического университета. Серия: Информационные компьютерные технологии в образовании. 2018. № 14. С. 5.

3. *Бороненко Т.А., Кайсина А.В., Федотова В.С.* Развитие цифровой грамотности школьников в условиях создания цифровой образовательной среды // *Перспективы науки и образования*. 2019. № 2 (38). С. 167.
4. *Буцык С.В.* “Цифровое” поколение в образовательной системе российского региона: проблемы и пути решения // *Открытое образование*. 2019. № 1. С. 27.
5. *Гэйбл Э.* Цифровая трансформация школьного образования. Международный опыт, тренды, глобальные рекомендации / Пер. с англ. Под науч. ред. П.А. Сергоманова. Национальный исследовательский университет “Высшая школа экономики”, Институт образования. М.: НИУ ВШЭ, 2019. 108 с. (Современная аналитика образования. № 2 (23)).
6. *Биленко П.Н., Блинов В.И., Дулинов М.В., Есенина Е.Ю., Кондаков А.М., Сергеев И.С.* Дидактическая концепция цифрового профессионального образования и обучения / Под науч. ред. В.И. Блинова. М.: “Перо”, 2019. 98 с.
7. *Маниковская М.А.* Цифровизация образования: вызовы традиционным нормам и принципам морали // *Власть и управление на Востоке России*. 2019. № 2 (87). С. 100.
8. *Морозов А.В., Самборская Л.Н.* Профессионализм учителя как важнейший ресурс и детерминанта качества педагогической деятельности в условиях цифровой образовательной среды // *Казанский педагогический журнал*. 2018. № 6 (131). С. 43.
9. *Мухин О.И.* Формирование таланта в эпоху цифровизации. Модель обучения одаренных и талантливых учащихся // *Вестник Пермского государственного гуманитарно-педагогического университета*. Серия: Информационные компьютерные технологии в образовании. 2017. № 13. С. 19.
10. *Уваров А.Ю., Ван С., Кан Ц. и др.* Проблемы и перспективы цифровой трансформации образования в России и Китае // II Российско-китайская конференция исследователей образования “Цифровая трансформация образования и искусственный интеллект”, 26–27 сентября 2019 г. Москва, Россия / Отв. ред. И.В. Дворецкая. Пер. с кит. Н.С. Кучмы. М.: Изд. дом Высшей школы экономики, 2019. 155 с.
11. *Уваров А.Ю., Гейбл Э., Дворецкая И.В. и др.* Трудности и перспективы цифровой трансформации образования / Под ред. А.Ю. Уварова, И.Д. Фрумина; М.: Изд. дом Высшей школы экономики, 2019. 343 с.
12. *Уваров А.Ю.* На пути к цифровой трансформации школы. М.: Образование и информатика, 2018. 120 с.
13. *Уваров А.Ю.* Модель цифровой школы и цифровая трансформация образования // *Исследователь*. 2019. № 1–2. С. 22.
14. Формирование цифровой грамотности обучающихся: Методические рекомендации для работников образования в рамках реализации Федерального проекта “Цифровая образовательная среда” / Авт.-сост. М. В. Кузьмина и др. Киров: ИРО Кировской области, 2019. 47 с.
15. *Якубенко А.А.* Алгоритмы построения трехмерных моделей объектов с регулярной структурой по фотографиям при взаимодействии с пользователем для виртуальных сред: Дис. ... канд. физ.-мат. наук. М.: Моск. гос. ун-т им. М.В. Ломоносова, 2012.