

# О НЕКОТОРЫХ ПОДХОДАХ К РАЗРАБОТКЕ ИНТЕРНЕТ-СРЕДЫ ДЛЯ СОЗДАНИЯ 3D-ПРИЛОЖЕНИЙ

Ломакин Г. А.

Кафедра программного обеспечения интеллектуальных и компьютерных систем Учреждение образования  
«Гродненский государственный университет имени Янки Купалы»  
Гродно, Республика Беларусь  
E-mail: spellbound.fpmi@gmail.com

*В статье приводятся общие подходы к разработке Интернет-среды для создания 3D-приложений. Описаны графическое ядро и набор инструментов, а также набор классов и интерфейсов для интеграции ядра в конкретные разработки на платформе .NET-приложений.*

## ВВЕДЕНИЕ

Визуализация реальных моделей является сложной задачей. Для экранной визуализации синтезированной 3D-модели используются различные алгоритмы, выбор, адаптация и модификация которых зависит от рассматриваемой задачи предметной области. Более того, чрезмерная трудоемкость создания объектов реального мира приводит к необходимости разработки специальных моделей и новых методов экранизации для таких объектов.

При современном развитии графических систем и технологий, актуальным является подход, связанный с переводом визуализации 3D-объектов на новый уровень. Разработка 3D-приложений характеризуется определенным набором ограничений, что, естественно, сказывается на результатах визуализации.

Наиболее часто разработчики 3D-приложений обращаются к двум основным библиотекам 3D-графики: DirectX и OpenGL. Однако работа с ними вызывает трудности, связанные, прежде всего, с написанием многих строк кода, использованием API и указателей. Часть проблем по реализации 3D-объектов можно решить с использованием XNA framework, используя соответствующие классы. Однако XNA не позволяет полностью устранить проблемы, возникающие при разработке 3D-приложений, т.к. указанный framework является .NET реализацией DirectX.

Таким образом, разработка среды, которая предоставит пользователю возможность быстрого создания 3D-объектов, используя современные Интернет- и мобильные технологии, является востребованным направлением в развитии компьютерной графики.

## I. КРАТКАЯ ХАРАКТЕРИСТИКА ИНТЕРНЕТ-СРЕДЫ

Предлагаемая интерактивная среда состоит из нескольких компонентов: облака для хранения контента в различных категориях; графического ядра, реализованного с использованием

OpenGL и DirectX; набора утилит для персонального компьютера; приложения для мобильной операционной системы Android; фреймворка, обеспечивающего доступ ко всем возможностям ядра.

Главными особенностями предлагаемой Интернет-среды являются: открытое ядро на OpenGL и DirectX; набор классов и интерфейсов для рендеринга примитивов, а так же набор базовых шейдеров для реализации различных эффектов на графическом конвейере; утилиты для построения визуализаций на платформах PC и Android; синхронизация контента клиентского приложения с сервером.

Рассмотрим основные аспекты, связанные с архитектурной и программной реализацией, а также – с основными конструктивными особенностями предлагаемой интегрированной среды разработки 3D-приложений.

## II. ОБЩИЕ ПОДХОДЫ К РЕАЛИЗАЦИИ ИНТЕРНЕТ-СРЕДЫ

Предлагаемая интегрированная среда абстрагирует пользователя от промежуточных уровней создания графики и дает возможность напрямую создавать графические сцены посредством GUI, в том числе, обеспечивая пользователя шаблонным контентом [1]. Были реализованы следующие модули [2]: ядро для построения сложных графических сцен предоставляющее открытое API; графическое ядро на OpenGL для ОС Windows 7 и OpenGL ES 2.0 для ОС Android; набор утилит, использующих ядро для манипулирования графическими процессами с помощью GUI; сервер с контентом для предоставления пользователю возможности использования шаблонного наполнения, а так же загрузки пользователем произвольного контента.

Для визуализации сложных трехмерных объектов реального и виртуального мира используется разработанное графическое ядро CoreX, базирующееся на XNA framework, которое включает Content Pipeline. Это упрощает работу с контентом и, при необходимости, позволяет расширить список поддерживаемых типов.

Отметим, что графические объекты представляются в качестве классов. Класс каждого графического объекта описывает его основные свойства, а также методы для отображения объекта в пространстве и обновления его состояния. Каждый класс связывается с определенным контентом, в нем же определяется, как данный контент будет использоваться.

На базе ядра CoreX реализован также и графический интерфейс пользователя (GUI), позволяющий производить интеграцию различных элементов управления в сцене и манипулировать происходящим на ней. Кроме графической составляющей в CoreX присутствует звук. Графическое ядро CoreX поддерживает DirectX 10, а также шейдерную модель версии 4.0. Реализованы такие эффекты как per-Pixel lighting, bump mapping, parallax mapping, shadow mapping, shadow volumes, multisampling.

Реализованное ядро предоставляет API для построения приложений. Разработанное API позволяет строить и изменять сцену, добавлять новые объекты для рендеринга и изменять их параметры (положение в пространстве, коэффициенты масштабирования, углы наклона относительно осей, текстуры, шейдер и др.). Кроме этого есть возможность применения глобальных эффектов ко всей сцене (таких как глобальное освещение/затенение, motion blur, HDR и др.).

Пользователь взаимодействует с ядром при помощи универсальной утилиты. Утилита состоит из двух основных частей. Первая часть отвечает за контент и объекты, использующие этот контент, а также за свойства и различные характеристики этих объектов. Вторая часть отвечает за настройку ядра CoreX: детализацию, качество текстур, используемые технологии и др. После конфигурации, все данные передаются на ядро, после чего запускается визуализация.

Для предоставления пользователю шаблонных ресурсов разработано дистанционное хранилище контента. Пользователю доступна возмож-

ность загрузки на сервер ресурсов для общего использования. Доступ к серверу осуществляется как из утилит, так и с помощью веб-интерфейса, включая мобильные платформы.

Для хранения 3D-объектов разработана модель базы данных, предназначенная для их создания и хранения. Каждый объект, помещаемый в базу, характеризуется типом и методом его реализации. Пользователи, которые добавляют или же используют соответствующий объект, могут добавить необходимые комментарии по поводу использования, методов реализации, характеристик объекта. В зависимости от типа, пользователям назначаются соответствующие привилегии по работе с 3D-объектами. Отметим также, что в модели данных поддерживается декларативным ссылочная целостность.

### III. ЗАКЛЮЧЕНИЕ

Предлагаемая Интернет-среда предоставляет пользователям возможности по формированию сцены, приданию различным объектам требуемых физических свойств, а также заданию необходимых взаимодействий объектов. Все взаимодействия в среде осуществляются посредством графического интерфейса.

Дальнейшая разработка системы предполагает добавление нового контента на сервер, расширение графических возможностей, перенос системы на другие платформы (например, Unix-системы). Несомненно, разработка такого рода является востребованной и найдет широкое применение, как в научных, так в игровых приложениях.

1. Мэрдок Келли Л. Autodesk 3ds Max 2009. 3D Studio max. Библия пользователя = Autodesk 3ds Max 2009 Bible. 3D Studio max / Келли Л. Мэрдок. – М.: «Диалектика», 2009. – С. 1312.
2. Ломакин, Г. А. Об одном подходе к построению графических приложений / Г. А. Ломакин // Наука-2011 : сб науч. ст. В 2ч. Ч. 2 / БГУ ; рекол.: В. И. Малугин (отв. ред.) [и др.]. – Минск : БГУ, 2011. – С. 89–91.