

И. И. Коляскин, М. И. Жадан

РАЗРАБОТКА КРОССПЛАТФОРМЕННЫХ ИГР НА ОСНОВЕ MICROSOFT XNA FRAMEWORK

Статья рассказывает об основных этапах и принципах создания игровых приложений с использованием технологии Microsoft XNA на языке программирования C# в среде разработки Microsoft Visual Studio на конкретном примере поиска созвездий. Используя возможности перечисленных технологий и продуктов, создано несложное игровое приложение, которое впоследствии легко может быть собрано под мобильную платформу Windows Phone с помощью Windows Phone SDK. Используя технологию Xamarin игра компилируется под такие популярные платформы как Android и iOS.

Microsoft XNA – набор инструментов с управляемой средой времени выполнения (.NET), созданный Microsoft, облегчающий разработку и управление компьютерными играми. XNA стремится освободить разработку игр от написания повторяющегося шаблонного кода и объединить различные аспекты разработки игр в одной системе.

Приложения, разработанные на XNA, поддерживаются платформами Windows, Microsoft Xbox 360 и Microsoft Zune. Теоретически писать можно на любом .Net языке в любой IDE, но официально поддерживаются только C# и XNA Game Studio Express и все версии Visual Studio 2005 и выше [1]. XNA также дает возможность портировать игры на поддерживаемые платформы с минимальными изменениями. XNA Framework скрывает низкоуровневые технологические детали, связанные с разработкой игры.

Основой игры является класс Microsoft.Xna.Framework.Game, который и представляет собой логику игры. В классе Game определены следующие методы:

- void Initialize() – вызывается единожды, для инициализации ресурсов до начала игры;
- void LoadContent() – вызывается единожды, используется для загрузки контента (спрайты и т.д.);
- void UnloadContent() – вызывается единожды, используется для выгрузки контента;
- void Update(GameTime gameTime) – в этом методе реализуется собственно логика игры, обработка коллизий, обработка событий клавиатуры или джойстика, проигрывание аудио и т. д.;
- void Draw(GameTime gameTime) – вызывается для прорисовки игрового поля.

Для создания приложения с помощью Microsoft XNA Framework необходимо установить его, после чего в Microsoft Visual Studio станет доступен новый вид создаваемого проекта – Windows Game. Такой проект сразу же будет содержать ссылки на компоненты XNA Framework и класс-наследник класса Game, который и будет описывать игру [2].

Для ознакомления с возможностями XNA Framework ниже будет описан процесс создания игрового приложения на этой технологии [3]. Игра представляет собой головоломку, где игрок должен находить созвездия, задаваемые в каждом уровне. Игровой экран представляет собой небо со звездами, а игроку дается возможность соединять эти звезды, образуя созвездия. Как только все заданные созвездия на текущем уровне найдены, происходит переход к следующему уровню.

Все уровни приложения разбиты на четыре локации. Локация представляет собой участок неба, содержащий созвездия всех уровней данной локации. Переход на следующий уровень в пределах локации представляет собой перемещение камеры к месту нахождения созвездий следующего уровня (рисунок 1).

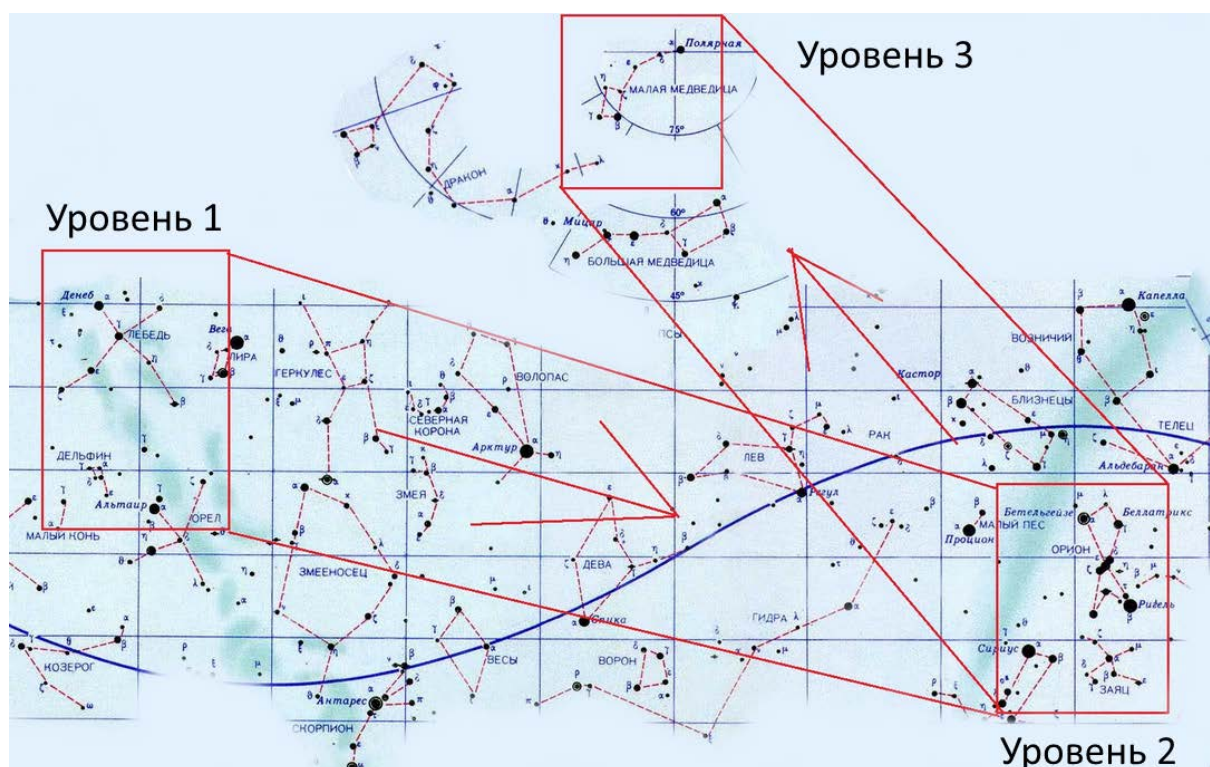


Рисунок 1 – Переходы между уровнями

Локации различаются по сложности, а также по расположению созвездий на реальной карте звездного неба. В первых локациях созвездия содержат мало звезд, камера сильно приближена к тому месту, где нужно искать созвездие, в уровнях мало «посторонних» звезд, т. е. тех, которые не относятся к искомому созвездию. С переходом на следующие локации все эти характеристики изменяются, что затрудняет поиск.

Для описания игровых объектов создаются классы игровых сущностей. Эти классы содержат лишь свойства и конструкторы и предназначены для хранения данных об этих самых сущностях. Сущности, необходимые в игре: Location (локация), LevelInfo (уровень), Constellation (созвездие), Star (звезда), Relation (отношение или связь между двумя звездами).

Для хранения данных о локациях используются XML-файлы, имеющие структуру описанных выше сущностей. Один файл содержит описание одной локации. Поскольку объем информации, необходимой для описания локаций, слишком велик, чтобы создавать её вручную, был создан редактор уровней, который позволяет быстро создавать уровни, добавлять звезды, связывать их, образуя созвездия и устанавливать прочие необходимые параметры. Редактор уровней был создан на технологии Windows Forms. Для создания созвездий использовались реальные карты северного и южного полушария небесной сферы. Редактор уровней сохраняет введенные данные в файлах, которые впоследствии считываются игрой (рисунок 2).

Создание самой игры начинается с переопределения класса Game. В этом классе должны быть переопределены основные методы, такие как Initialize, LoadContent, UnloadContent, если требуется, и, самое главное, методы Update и Draw.

В методе Initialize задаются главные настройки, касающиеся всей игры. Например, разрешение экрана, инициализация игровой камеры, менеджера ресурсов, который будет предоставлять текстуры и прочие ресурсы в дальнейшем и т. д. В методе LoadContent в навигатор по страницам загружаются игровые страницы, такие как главное меню, меню выбора локации и игровой экран. В Update вызываются методы Update

очередь добавляет к ней игровую логику и прорисовку. В методах Update и Draw вызываются соответствующие методы для связей, звезд и иллюстраций к созвездиям, появляющимся после открытия заданного созвездия.

Аналогично классу Level, классы звезд Star и связей Connection содержат внутри себя сущность, считываемую из XML-файла с данными, и дополняют это прорисовкой.

Класс Illustration создан в декоративных целях. Иллюстрация появляется, когда отгадано созвездие. Этот класс управляет анимацией иллюстрации – она плавно изменяет форму и цвет согласно логике описанной в методах Update и Draw данного класса.

К игровым объектам можно отнести класс Background, который управляет фоновыми объектами, такими как туманности и фоновые звезды. Эти классы созданы исключительно для декорирования игрового поля, но они делают игру гораздо более приятной на вид, а значит и более интересной. Класс Background обновляет и прорисовывает все фоновые объекты, которые в свою очередь определяют логику этой прорисовки и собственную анимацию, если таковая имеется. Игровой экран с разными объектами показан на рисунке 3.

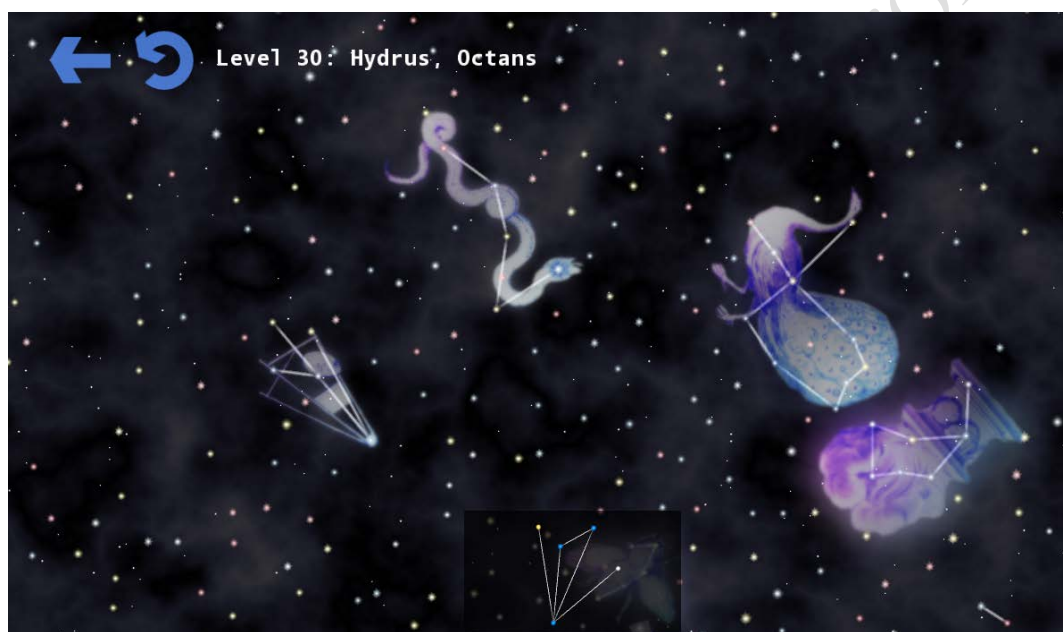


Рисунок 3 – Игровые объекты

XNA позволяет создать копию имеющегося проекта, который запускается на Windows, для платформы Windows Phone. Для этого необходимо установить Windows Phone SDK, после чего приложение может быть собрано под мобильную платформу Windows Phone с минимальными изменениями, в данном случае касающимися работы с файлами. Логика игры остается нетронутой.

Приложение создано для Android и iOS. Для этих целей использована технология Xamarin.

Литература

- 1 Шилдт, Г., C# 4.0. Полное руководство / Г. Шилдт. – Издательство: Вильямс, 2010. – 1056 с.
- 2 Thurrott, P., Windows Phone 7 Secrets / P. Thurrott – Издательство: Wiley Publishing, Inc., 2011. – 433 с.
- 3 Petzold, Ch., Microsoft XNA Framework Edition: Programming Windows Phone 7 / Ch. Petzold. – Издательство: Microsoft Press, 2010. – 389 с.