

*К ВОПРОСУ ОБ ИСТОРИИ РАЗВИТИЯ КОМПЬЮТЕРНЫХ ИГР***В. А. МУРЛИНА, А.Е. ШЕЛЯКИН**

*Кубанский государственный технологический университет,
350072, Российская Федерация, г. Краснодар, ул. Московская, 2,
электронная почта: ShelyakinAleksandr@yandex.ru*

Статья раскрывает историю развития компьютерных игр. Коротко рассмотрены и описаны игры прошлого века. Показан большой скачок в проектировании и разработке игровых приложений. Так же игровые приложения представлены как сложные программные продукты со стороны процесса разработки, реализации и продажи конечного результата. Выделены и проанализированы основные моменты, к которым на первых стадиях проектирования уделяется наибольшее и наименьшее внимание, а какие, в некоторых случаях, не берутся в расчет вообще. Затронуты моменты оптимизации и переноса на другие платформы, как мобильные, так и стационарные. Рассмотрены модели жизненного цикла и предложен наиболее эффективный жизненный цикл на взгляд автора. В настоящее время получили наибольшую популярность разработки для мобильных устройств, что не исключает дальнейшие разработки для стационарных машин.

Ключевые слова: игровое приложение, история, основные аспекты, тестирование.

Люди любят играть. Люди обожают, как кэрроловская Алиса, «представить себя королями и королевами» и забыть, что завтра надо идти в школу или на работу. И компьютерные игры, пожалуй, ничего нового в эту ситуацию не привнесли. Какая разница, в сущности, во что именно играть?..

Впрочем, несмотря на отсутствие качественных изменений, количественных накопилось – хоть отбавляй! За прошедшие четыре десятилетия индустрия компьютерных игр успела не просто родиться из ничего и завоевать популярность, но и добраться по добытым деньгам до заслуженной и куда более «раскрученной» индустрии кино. А начиналось все очень и очень скромно...

В далеком шестьдесят первом году двадцатого века в Массачусетском технологическом (знаменитый MIT) на одном из мэйнфреймов впервые родилась компьютерная игра под названием «SpaceWar». Это событие осталось практически никому не известным, поскольку компьютеры тех времен были дороговаты для того, чтобы на них играть. Но, само собой, это ничуть не

мешало настоящим программистам, работавшим на них, делать для себя игры и играть в них.

Знаменитая «SpaceWar» была весьма любопытна с технологической точки зрения: в те времена и на том компьютере еще не было привычных растровых графических дисплеев, только текстовые терминалы и, как вершина графических возможностей, векторные графические дисплеи, где электронный луч на каждом кадре не бегает по строкам, а вырисовывает заданные программой контуры объектов.

Суть игры-прародительницы была весьма проста: два небольших кораблика летали по экрану и стреляли друг в друга снарядами. Так что первый в мире жанр – это, безусловно, аркада [1].

Позже, в 1952 году, появилась программа «OXO», имитирующая игру «крестики-нолики», созданная А.С. Дугласом как часть его докторской диссертации в Кембриджском университете. Игра работала на большом университетском компьютере, известном как «EDSAC» (Electronic Delay Storage Automatic Calculator).

В 1958 году Уильям Хигинботам, помогавший строить первую ядерную бомбу в Брукхейвенской национальной лаборатории (Аптон, Нью-Йорк), для развлечения посетителей создал «Tennis for Two» («Теннис для двоих»).

В 1962 году Стив Рассел написал «Космическая война и Большое Приключение Джона» (англ. Spacewar and John's Great Adventure). Игра работала на миникомпьютере PDP-1 и быстро распространилась по всем университетам страны.

В 1968 году Ральф Баер, который позже стал известен как «Король Видеоигр», запросил патент на раннюю версию игровой консоли «Television Gaming and Training Apparatus». В 1967 году Баер создал пинг-понг игру, похожую на «Теннис для двоих». Вместе с Magnavox он работал над созданием первой консоли, названной «Magnavox Odyssey» в 1972 году. Разработка игровых автоматов в 1970-х годах привела к так называемому «Золотому веку аркад». Одна из самых известных игр того времени – «Pong» [2].

Существуют попытки выделить компьютерные игры как отдельную область искусства, наряду с театром, кино и т. п. Разработка игр может оказаться не только увлекательным, но и прибыльным делом, примеров этому предостаточно в истории.

Сейчас игровые приложения – это сложные программы с множеством алгоритмов и очень красивой графикой. Игровые приложения могут быть разные, от простых игр для обучения детей любого возраста, до сложных, для проверки профессиональных качеств людей разных профессий.

Ролевая игра – игра обучающего или развлекательного назначения, вид драматического действия, участники которого действуют в рамках выбранных ими ролей, руководствуясь характером своей роли и внутренней логикой среды действия; вместе создают или следуют уже созданному сюжету. Действия участников игры считаются успешными или нет в соответствии с принятыми правилами. Игроки могут свободно импровизировать в рамках выбранных правил, определяя направление и исход игры.

Игростроение – трудоемкий процесс, требующий от разработчиков полной отдачи и внимания, даже к самым незначительным мелочам. Неважно будь то мелкие детали персонажей или крупные элементы окружающей обстановки. Звук занимает одну из первых строчек в разработке игр.

Хорошо подобранный звук совершенно меняет восприятие происходящего, будь то мелкий шорох или какой-то грохот. Игрок может моментально отреагировать на тот или иной звук определенными действиями. Опытный игрок не обращает внимания на определенные действия сопровождаемые звуком. Ему достаточно лишь услышать и он сразу определит, какое игровое событие произошло. Другой аспект звукового сопровождения в игре это озвучивание персонажей. Грамотно поставленная речь человека озвучивающего данного персонажа может заинтересовать игрока самыми разными аспектами, например акцент, эмоции, манера речи, дикция, внятность. Кому будет приятно слушать мямлю или непонятно говорящего персонажа, если это конечно не режиссерский ход. Иногда не самый лучший текст,

интересно слушать за счет актера который вложил душу в озвучивание своего персонажа.

Другой вопрос нужно ли это? Звуковые файлы в хорошем качестве значительно увеличивают размер игры. Сегодня для компьютерных игр размер в 30 Гб на жестком диске в порядке вещей, но для мобильных платформ это непосильный размер. Изменение формата или снижение качества звукового файла может негативно отразиться на игре в целом. Как бы хорошо актер не озвучил своего персонажа, плохое качество воспроизведения все испортит. Но в игре обязательно должен быть звук, и здесь нужно подумать, что стоит на первом плане, что на втором, а что можно убрать. Если делать небольшую игру, то можно обойтись обычными текстовыми сообщениями. Просто подобрать размер шрифта, цвет и скорость его появления и удаления с экрана, если это какое-то игровое событие. Следующая задача, которая сразу ставится перед разработчиками, это хорошо написанный и легко читаемый текст. Здесь как с книгой, если в обычной книге, предназначенной для обычного обывателя, будет множество трудночитаемых и незнакомых терминов то она «отобьёт охоту» прочтения у любого человека, так же в тексте персонажей не должно быть лишних слов. Сценарист, кем бы он ни был, должен написать текст соответствующий игровой тематике, проработать диалоги персонажей, что бы они не были слишком длинные, но и не слишком краткие, и уложить в этот текст суть происходящего на экране.

Следующий аспект – это графика, визуальная составляющая всего проекта, то, что игрок непосредственно видит перед собой. И первое что он видит перед собой в игре это главное меню. Меню в игре, как и в другой любой программе, должно быть понятным, а его стиль обычно соответствует самой игре. В порядке вещей, если игра, затрагивающая средневековую тематику, будет использовать на заднем плане изображения замков или рыцарей, а сами кнопки и курсор могут представлять собой щиты и меч, а если она затрагивает современные военные действия, меню можно представить в виде тактической карты. Со временем как то само собой утвердились и основные кнопки меню,

практически во всех играх их число составляет от пяти до семи. Меньше, много подменю, больше, слишком много информации, да и пользователи привыкли, что игровые настройки видео и аудио находятся в подменю кнопки под названием «параметры» или «настройки». Персонажи тоже должны быть сделаны под стать игре, если игра в двух мерном пространстве будут трех мерные модели, это может оказаться не совсем к месту. Проще будет реализовать это как несколько картинок сменяющих друг друга или как это называется анимировать 2D персонажей. Анимация 2D персонажа происходит в два этапа, первое эта анимация покоя, второе это анимация действия, количество действий может быть много, и для каждого действия может быть своя анимация. В разработке анимации, нужно грамотно подобрать определенное количество изображений и скорость их смены друг с другом. Это влияет на размер и работоспособность самой игры. Если в анимации персонажей будет слишком много изображений то игре понадобится много времени для загрузки, а если слишком мало, то анимация будет прерывистой и некрасивой. Здесь нужно искать золотую середину между количеством и скоростью их смены. Когда разработчик это «нащупает», анимация будет выглядеть плавно и красиво.

Следующая часть это внутренняя начинка, а именно программный код, насколько он громоздкий, хорошо ли оптимизирован. В эти два термина вписывается очень много понятий, которые программист должен учесть все до единого. Скорость работы, если загрузка каждого уровня будет происходить в течение пары десятков минут, у игроков не хватит терпения. Оптимизация, любой программист старается сделать свой код понятным хотя бы для себя, убрать пару лишних переменных, усовершенствовать некоторые алгоритмы под поставленную задачу, потому что любой выпуск программного продукта, в любой сфере, подразумевает сопровождение. Будут выпускаться обновления и дополнения, которые затрагивают оригинальный код. Это работа может начаться не сразу, и если код не будет понятен, то на само осмысление что делает та или другая сточка кода, может уйти не один день. Вследствие чего,

сопровождение даже небольшого продукта может вылиться в огромные затраты, которые могут превысить стоимость первоначальной разработки. Тестирование это еще один вид, как разработки, так и сопровождения. В объектно-ориентированном программировании тестируются классы и методы, это на много упрощает процесс. На некоторых движках разработки классы называются скриптами. Скрипты прикрепляются к объектам. К объекту может быть прикреплено много скриптов и на оборот, несколько объектов могут использовать один скрипт. Как было сказано выше скрипты это классы, которые содержат методы, вследствие чего процесс тестирования упрощается. Тестированию отведен целый большой пункт в разработке любого ПО, разработаны, международные стандарты, ГОСТы, методы и алгоритмы тестирования. Стандарты определяют две связанные модели качества программного обеспечения (ISO 9126–01) – внутреннее качество, внешнее качество, а также набор соответствующих работ по оценке качества программного обеспечения (ISO14598–98). Качество программного обеспечения, управление качеством и качество программной инженерии имеют непосредственное отношение к качеству создаваемого программного продукта [3]. Существует два важнейших стандарта в области качества программного обеспечения. Общая система менеджмента качества ISO 9001:2000 в приложении к программным проектам, и представленная в виде специальных рекомендаций ISO 90003:2004. Другие важные стандарты – CMMI и ISO 15504 предоставляет рекомендации по совершенствованию ПС (программных средств).

Из отечественных есть «ГОСТ Р ИСО/МЭК 9126-93». Этот ГОСТ предлагает шесть характеристик, которые с минимальным дублированием описывают качество программного обеспечения.

Функциональные возможности (Functionality) Набор атрибутов, относящихся к сути набора функций и их конкретным свойствам. Функциями являются те, которые реализуют установленные или предполагаемые потребности.

Надежность (Reliability), набор атрибутов, относящихся к способности программного обеспечения сохранять свой уровень качества функционирования при установленных условиях за установленный период времени.

Практичность (Usability), набор атрибутов, относящихся к объему работ, требуемых для использования и индивидуальной оценки такого использования определенным или предполагаемым кругом пользователей.

Эффективность (Efficiencies) набор атрибутов, относящихся к соотношению между уровнем качества функционирования программного обеспечения и объемом используемых ресурсов при установленных условиях.

Сопровождаемость (Maintainability) Набор атрибутов, относящихся к объему работ, требуемых для проведения конкретных изменений (модификаций).

Мобильность (Portability) Набор атрибутов, относящихся к способности программного обеспечения быть перенесенным из одного окружения в другое.

В некоторых случаях в игровых приложениях на первый план выносятся мобильность и надежность. Мобильные платформы отличаются самыми разными параметрами. В расчет берется их мощность, операционная система (ОС) и размеры. В зависимости от ОС, выбирается язык программирования, на котором будет написано приложение. Иногда для переноса на другую ОС нужно переписать код на другой язык, при этом нужно учитывать быстроедействие переписанного кода. Поскольку разные языки используют разные методы и методологии оптимизации одних и тех же алгоритмов. Размер тоже важен, если игровое приложение пишется на ПК, размеры монитора всегда стандартны, но на мобильных платформах они сильно изменяются, и здесь нужно продумать, как подогнать графику под определённые моменты. Мощность была описана выше.

К любому программному продукту (далее – ПП) применимы жизненные циклы (далее – ЖЦ). Игры – не исключение.

К настоящему времени наибольшее распространение получили следующие две основные модели ЖЦ:

- каскадная модель;
- спиральная модель.

Из выше предложенных ЖЦ, лучше всего подходит спиральная модель. Она позволяет показать пользователям самую раннюю версию, и узнать мнение игроков на самой ранней стадии разработки. Разработчики в зависимости от собранной информации могут корректировать тех. задание, подстраиваясь под желания пользователей и менять приоритеты поставленных задач.

За очень короткое время игровые приложения совершили скачек вперед. В частности это связано с развитием мобильных платформ. Разработчики больше не ограничены стационарными машинами. Это развитие расширило возможности, но и создало новые проблемы, которые описаны выше.

Вывод: О влиянии компьютерных игр на людей, можно спорить бесконечно. В этой статье игры были рассмотрены изнутри, их оболочка и начинка. Игровое приложение – это сложный программный продукт, местами даже сложнее некоторых программ, как в разработке, так и в продаже. Они требуют от разработчиков полной самоотдачи и терпения.

ЛИТЕРАТУРА

1. Павел Гродек – История компьютерных игр – Виртуальный компьютерный музей – URL: [<http://www.computer-museum.ru/games/genesis.htm>] – Дата обращения: 23.01.2016.

2. История развития компьютерных игр – компьютерных игры как искусство – URL: [http://gamesisart.ru/istoriya_komputernyh_igr.html] – Дата обращения: 23.01.2016.

3. ГОСТ Р ИСО/МЭК 9126-93. Информационная технология. Оценка программной продукции. Характеристики качества и руководства по их применению. Введен в действие: 01.07.1994.

REFERENCES

1. Pavel Grodek – Istoriya kompyuternykh igr – Virtualnyy kompyuternyy muzey – URL: [<http://www.computer-museum.ru/games/genesis.htm>] – Data obrashcheniya: 23.01.2016.
2. Istoriya razvitiya kompyuternykh igr – kompyuternykh igry kak iskusstvo – URL: [http://gamesisart.ru/istoriya_komputernyh_igr.html] – Data obrashcheniya: 23.01.2016.
3. GOST R ISO/MEK 9126-93. Informatsionnaya tekhnologiya. Otsenka programmnoy produktsii. Kharakteristiki kachestva i rukovodstva po ikh primeneniyu. Vveden v deystvie: 01.07.1994.

*TO A QUESTION ABOUT THE HISTORY OF COMPUTER GAMES***V. A. MURLINA, A. E. SHELYAKIN**

*Kuban State Technological University,
2, Moskovskaya st., Krasnodar, Russian Federation, 350072,
e-mail: ShelyakinAleksandr@yandex.ru*

This article touched upon the history of computer games. Briefly discussed and described in that it represents the game of the last century that it was possible to see how big the race they did in such a short time. Gaming applications as well as complex represented by the software development process, and realization of the sale of the final result. Isolated and analyzed highlights to which the first design stages for the highest and lowest paid attention, and which in some cases are not taken into account at all. Touched upon aspects of optimization and migration to other platforms, both mobile and stationary. We consider the life cycle model and propose the most effective life cycle of the author's opinion.

Key words: gaming applications, the history, the main aspects of testing.