

УДК 004.414.3

РАСКАДРОВКА КАК ОДНО ИЗ ПРЕДСТАВЛЕНИЙ СЦЕНАРНОГО ПРОТОТИПА КОМПЬЮТЕРНЫХ ИГР

Г. Ф. Сахибгареева¹, [0000-0003-4673-3253], О. А. Бедрин², [0000-0003-3300-4318],

В. В. Кугуракова³, [0000-0002-1552-4910]

^{1,2,3}*Институт информационных технологий и интеллектуальных систем Казанского (Приволжского) федерального университета*

¹gulnara.sahibgareeva42@gmail.com, ²simplavero@gmail.com,

³vlada.kugurakova@gmail.com

Аннотация

Работа посвящена изучению и усовершенствованию процесса проектирования, разработки и тестирования повествования видеоигр. Изучены существующие практики написания и поддержки в актуальном состоянии сценария интерактивных произведений. Сформулированы определение сценарного прототипа, а также требования к его форме. Выдвинута идея об эффективности автоматизации создания сценарного прототипа в виде инструмента-генератора. Составлено видение такого инструмента. Представлено влияние такого инструмента на порядок разработки. Реализован компонент инструмента и проведен эксперимент, который доказывает эффективность на таком примере, как генерация раскадровки из текста. Сформулированы планы на будущую разработку.

Ключевые слова: компьютерные игры, разработка видеоигр, интерактивное повествование, сценарный прототип, нарративный дизайн, сценаристика, игровая документация, раскадровка, генерация раскадровки, интерактивная раскадровка.

ВВЕДЕНИЕ

Разработка видеоигр – длительный и дорогостоящий процесс, поэтому любые оптимизация и автоматизация могут сыграть ключевую роль в успехе финального продукта.

Сложность разработки видеоигр заключается прежде всего в их составе. Они включают в себя визуализацию, аудиоряд, интерактивное взаимодействие,

правила игры и историю. Каждый компонент тесно связан с остальными, и все вместе они дают результат, уникальный для любого медиа, – опыт интерактивного потребления контента с драматургической подоплекой. Иными словами – игроки сами творят историю.

Индустрия разработки игр – молодое и бурно развивающееся направление. Конкретно повествовательная составляющая долго оставалась скорее бременем, чем мощным инструментом. Только в последние годы в данном направлении начинают появляться своя теория и свои эксперты.

В рамках данной работы проведен анализ существующих практик разработки интерактивного повествования, а также результатов попыток автоматизации этого процесса.

Глобальная цель этой работы – познакомить с процессом разработки инструмента, который призван автоматизировать генерацию прототипа повествования видеоигр, а также продолжить работу по исследованию особенностей процесса разработки видеоигр.

В разделе 1 «Инструменты сценариста видеоигр» приведен анализ ряда форматов документации, практик разработки, а также программного обеспечения, которое применяется в разработке видеоигр.

В разделе 2 «Сценарный прототип» даны определение сценарного прототипа, обоснование этого определения, а также сравнительный анализ этой практики с некоторыми инструментами, приведенными в разделе 1.

В разделе 3 «Генерация сценарного прототипа» дано обоснование необходимости и пользы автоматизации процесса создания сценарного прототипа в форме генератора.

В разделе 4 «Существующие решения визуализации» проанализированы существующие инструменты визуализации, которые доказывают, что генерация визуализации из текста возможна.

В разделе 5 «Видение инструмента генерации сценарного прототипа» представлены схема и видение генератора сценарных прототипов, а также приведен пример внедрения этого инструмента в конвейер разработки повествования.

В разделе 6 «Прогресс в разработке инструмента» представлен прогресс в разработке генератора.

В разделе 7 «Видение процесса генерации событий в пространстве» сформулировано видение процесса генерации визуализации для сценарного прототипа для модуля визуализации.

В разделе 8 «Реализация процесса генерации событий в пространстве» приведена техническая реализация.

В разделе 9 «Результаты работы процесса визуализации» приведены пример генерации раскадровки, а также результаты эксперимента.

В разделе 10 «Дальнейшее развитие» сформулированы планы на будущее.

1. ИНСТРУМЕНТЫ СЦЕНАРИСТА ВИДЕОИГР

Для налаживания коммуникаций с командой разработки и донесения основных идей и важных нюансов сценаристы используют различную документацию. Некоторые технические сведения используются только в процессе разработки; другие содержат текст, который игроки увидят в видеоигре в качестве контентной составляющей. Каждый формат документации призван решить специфические задачи реализации повествовательной составляющей игры. Разработка не может идти с применением единственного формата, т. к. каждый из них имеет свои преимущества и недостатки.

В настоящей раздел представлен анализ таких форматов, как текстовый документ, wiki-разметка, таблица, раскадровка, а также такой практики, как бумажный прототип. Помимо этого, проанализировано программное обеспечение, которое может применяться в работе игровых сценаристов: Twine [1], RenPy [2], Articy Draft [3].

Перечень проанализированных инструментов неисчерпывающий. Помимо них сценарист может использовать в своей работе проприетарные инструменты игровой компании, такие как модули редактирования диалогов и квестов, реализованные в игровых движках Unity [4] или Unreal Engine [5], или другие инструменты и сервисы, которые адаптируются для задач разработки, например, Miro [6], diagrams.net [7] и другие. Анализ проведен нами с точки зрения эффективности в процессе разработки игрового повествования.

1.1. Форматы документации

Текстовый документ

Явление интерактивности скрывает потенциал, который не раскрыт в полной мере. Это значит, что необходимы специализированные форматы и инструменты для работы с формами медиа, которые содержат в себе интерактивность.

Пока индустрии разработки видеоигр не хватает собственной фундаментальной теории и глубокого осознания специфики интерактивного повествования видеоигр, поэтому применяются практики и инструменты других медиа. Например, в случае видеоигр уместны драматургические приемы. Когда речь идет о линейном повествовании или небольшом количестве выбора или о кат-сценах, видеороликах, которые переключают с активного игрового процесса на пассивный просмотр кинематографических отрывков, действуют те же правила погружения, как и для зрителя. Различие состоит в том, что такие вставки прерывают процесс игры, что иногда бывает неуместно.

Другой пример – голливудский стандарт написания документации. Он общепринят во всем мире и применяется для форматирования сценариев видеоигр.

Однако, когда речь заходит о поддержании объемного разветвленного повествования, формата текстового документа оказывается недостаточно. Для создания истории с возможностью выбора необходимы гибкие инструменты, которые позволяют создавать, редактировать и вести учет каждого разветвления в структуре игрового повествования. Редактирование текста – это трудоемкий процесс, который лишен возможности отслеживания комплекса зависимостей событий друг от друга.

Эффективность восприятия текста сравнительно низкая, в отличие от визуальных и интерактивных произведений. Было доказано, что эффективность визуализации [8] и, тем более, погружения [9] значительно выше, чем у передачи информации через текст. Таким образом, текст, сопровождаемый визуализацией или визуализированный в полном объеме, воспринимается лучше, а это значит, что внутри команды разработки увеличивается вероятность взаимного понимания сюжета разрабатываемой игры.

Wiki-разметка

Формат wiki-разметки наследует недостатки текста – это отсутствие гибкости, сложность в поддержании актуальности, а также необходимость в длительном и вдумчивом изучении. Однако он имеет и преимущества: формат wiki-разметки – это возможность создания веб-энциклопедии о внутриигровом мире. В каждой тематической статье возможно размещение ссылок на другие статьи, что создает возможность нелинейного изучения информации.

Данный формат применим в работе сценаристов для специфической цели хранения всеобщего свода правил и истин, касающихся игрового мира. Кроме этого опубликованные wiki-ресурсы позволяют создать некоторую трансмедийность, в которой игроки могут познакомиться с миром игры как в рамках игрового процесса, так и вне игрового пространства. Однако чаще такие ресурсы создаются усилиями как раз фан-сообществ, что еще раз доказывает, что данный формат имеет успех в представлении больших объемов информации в виде, привлекательном для изучения.

Таблица

Таблица позволяет представлять информацию в двумерном и даже в N-мерном пространстве, в котором каждая ячейка определяется правилами и связью с другими ячейками. Рассмотрим преимущества и недостатки табличного представления на примере (см. рис. 1).

№	Название	Персонаж	Реплика персонажа	1 этап	2 этап	3 этап	4 этап	5 этап
1	Именинный торт	Инна	“Сегодня у Оли День рождения. Давай испечем ей торт.”	Купить продукты	Выбрать рецепт	Испечь коржи	Собрать торт	Подарить торт

Рис. 1. Пример квеста в формате таблицы

В примере приведен фрагмент воображаемого мобильного квеста. Для каждого внутриигрового задания выделены следующие параметры: идентификационный номер, название, имя персонажа, который дает игроку задание, его реплика для дачи задания и этапы прохождения задания. Рассмотрим цель каждого параметра:

- идентификационный номер необходим для ускорения коммуникаций в команде разработки;
- название задания также необходимо для коммуникации, но помимо этого оно отображается в самой игре;
- игрок должен каким-то образом получить задание, в данном случае его выдает конкретный персонаж, который с этой целью произносит определенную реплику;
- для выполнения задания игрок должен пройти конечное число этапов.

Табличное решение можно расширять по количеству параметров и строк до объема, необходимого в разработке. Табличное решение эффективнее текстового, т. к. наглядно представляет информацию в структурированном виде. Заметить в таком представлении ошибки и исправить их значительно проще, в особенности, если в таблице настроены формулы и скрипты.

Таблицы подходят не только для квестов. В таком формате можно хранить информацию о необходимости визуализации и аудиального сопровождения, указать специальные требования. Кроме этого возможно прописать формулы для составления текстов из внутриигровых сущностей.

Таблицы – хороший технический инструмент, призванный структурировать информацию и организовать работу команды. Однако он не подходит для презентации истории игры и последующего тестирования и оценки, т. к. таблица не имеет ничего общего с игровым процессом.

1.1. Раскадровка

Раскадровка используется в производстве фильмов, рекламы и других аудиовизуальных произведений. Это последовательность кадров, которая фиксируют в себе основные сцены и кадры из будущего произведения. Степень детализации при этом может быть разной.

Если добавить для каждого кадра возможность выбора, то следующий кадр, который станет доступен для просмотра, будет определяться этим выбором. Получается интерактивная раскадровка. Данный термин не является общепринятым и используется в контексте данной статьи как авторское определение.

Прямой аналогией для интерактивной раскадровки является жанр визуальных новелл. В них есть фоновая картинка, картинки персонажей, окна с именами и репликами персонажей, а также окно выбора реплики или выбора дальнейшего развития событий.

Как можно оценить визуализацию классической раскадровки, так можно протестировать интерактивную раскадровку на различные параметры. Можно оценить наличие персонажей в той или иной сцене, порционность текста в диалогах и записках, уместность аудиального сопровождения и цветового решения.

Раскадровка, как способ визуализации, привлекательнее и эффективнее, чем текст или таблица. Этот упрощенный формат поддается тестированию и сбору оценок от целевой аудитории. Данный формат также примечателен тем, что реализует интерактивную природу игры.

Минусом раскадровки является статичность картинки. Основная доля видеоигр – это произведения, в которых игрок не просто видит набор кадров, он самостоятельно перемещается в пространстве и выбирает способ взаимодействия с доступными внутриигровыми объектами.

1.2. Бумажный прототип

Бумажный прототип – это гибкий инструмент разработки и тестирования игрового повествования. Его отличие в том, что он позволяет вносить изменения в историю на ходу, во время проверки на целевой аудитории.

Для создания бумажного прототипа необходимо подобрать необходимые материалы: а) поле для развития действий, карту, разлинованный ватман или шахматную доску; б) фигуры, которые будут олицетворять персонажей игроков и игровых персонажей; в) кубики или приложение с генерацией чисел для создания случайности в развитии событий и г) специфические элементы, присущие проектируемому повествованию. Если сценарист способен провести игроков через всю историю с помощью подручных средств и достаточно увлечь участников, то есть вероятность, что история будет также увлекательна и в видеоигре.

Однако здесь выступает другая особенность видеоигр – наличие мануального взаимодействия, а также восприятие аудиальной и визуальной составляющих через устройства ввода/вывода компьютера, телефона и других платформ (см., например, [10, 11]). Такая специфика, как целевое устройство и атмосфера, в

которых находится игрок, важна с точки зрения моделирования опыта. Бумажный прототип не дает возможности воссоздания реалистичных условий процесса игры. С помощью бумажного прототипа можно проверить специфические характеристики интерактивного повествования, такие как погружение, темпоритм или эмпатия.

Погружение [12] – состояние, в котором игрок забывает о времени и реальности, проникается историей, осознает ее важность для себя. Погружение – это знакомое состояние, в котором каждый может обнаружить себя, когда читает книги и смотрит фильмы. Это чувство оторванности от происходящего. Тот же эффект уместен для интерактивного повествования.

Темпоритм [13] – частота и плотность подачи истории. Игрок знакомится с миром игры постепенно. Кроме этого должна сохраняться интрига, которая удержит игрока в процессе изучения истории. Темпоритм регулируется исключительно с помощью тестирования, и бумажный прототип позволяет это сделать.

Эмпатия [14] – это важное чувство, которое рождается у игрока по отношению к персонажу, за которого он играет, а также по отношению к другим персонажам. Каждый находит в героях истории что-то своё. Но бывают неудачные персонажи, которые либо сами по себе плохо продуманы, либо не соответствуют ожиданиям целевой аудитории, текущему времени. Данный параметр также проверяется только на практике в ходе тестирования.

Таким образом, бумажный прототип является эффективным и дешевым инструментом тестирования игровой истории на ранних стадиях [15]. Однако еще один минус бумажного прототипа состоит в том, что каждый раз при необходимости провести тестирование необходимо организовать встречу и задействовать сценариста.

1.3. Программное обеспечение

Свойства, важные для инструментария, который применяет игровой сценарист, – это гибкость, возможность поддержки разветвленной системы и ее быстрого редактирования. Для этого сценаристы применяют дополнительное специализированное программное обеспечение или адаптируют существующие сер-

висы под свои задачи. Ниже представлен анализ некоторых инструментов, которые принципиально отличаются друг от друга функционалом и возможностями в разработке: Twine, RenPy, Articy Draft.

Twine

Twine – это open-source движок для создания текстовых игр. Инструмент снижает порог вхождения в игровую индустрию благодаря трем ключевым преимуществам:

- построение историй происходит в виде системы карточек, навыки создания html-разметки при этом необходимы минимально;
- доступны настройки стиля текста, макросов, добавление условий, переменных, картинок и звуков;
- доступен экспорт проекта в виде html-файла; его можно опубликовать в виде веб-страницы.

Результат работы в Twine можно экспортировать в Unity с помощью ассета Crandle [16].

RenPy

RenPy – бесплатный open-source движок для разработки визуальных новелл. Три ключевых преимущества RenPy:

- редактор кода находится в самом движке;
- используется достаточно легкий в изучении язык программирования Python;
- имеется встроенный гайд для быстрого освоения инструмента.

В проект в RenPy можно добавлять стандартные компоненты новелл – текст, фоновые изображения, персонажей, музыку, звуки, анимации. Визуальная новелла эффективнее текстовой игры, т. к. включает в себя визуализацию.

Articy:draft

Articy:draft – это приложение для визуализации и организации игровых компонентов. Разработчики относят его к приложениям для разработки point-and-click адвенчур [17], но оно применимо при разработке игр других жанров.

Articy:draft позволяет систематизировать информацию, которая необходима в течение разработки игры в стандартной форме. Систему хранения сущностей в Articy:draft можно сравнить с диаграммой связей или с системой wiki-разметки.

С точки зрения разработки сценарного прототипа у Articy:draft есть следующие три преимущества:

- экспорт данных доступен в формате json, что делает инструмент универсальным;
- в Articy:draft можно хранить разнообразную информацию: персонажей, диалоги, квесты, инвентарь, достижения и других игровые сущности;
- принцип визуального программирования снижает порог вхождения для начинающих специалистов.

Минус Articy:draft – в его стоимости: система доступна для крупных компаний. В случае с независимыми разработчиками использование Articy:draft может оказаться неоправданным. Функционал Articy:draft позволяет назвать его универсальным инструментом благодаря встроенным шаблонам и готовым элементам пользовательского интерфейса. Articy:draft позволяет разработать быстрые прототипы для системы диалогов и узкого перечня жанров.

2. СЦЕНАРНЫЙ ПРОТОТИП

Сценарный прототип – это интерактивный цифровой прототип повествовательной составляющей видеоигры. Приведем обоснование определения.

а) *Цифровой формат* более точно моделирует игровой процесс будущей игры. Игрок будет проводить время за компьютером или телефоном, поэтому важно соблюсти это физическое взаимодействие на ранних этапах, чтобы не исказить впечатления и получить релевантные отзывы о прототипе.

б) *Повествовательная компонента* неотделима от игрового процесса. В рамках определения допускается, что прототип фокусируется на повествовании и игнорирует интерактивность ровно до тех пор, пока это не принципиально для повествования. В случае механик боя или для создания неожиданности события возможно применение генерации случайного исхода, чтобы смоделировать эмо-

ции от игрового процесса. В случае же повествовательных механик подразумевается, что они будут реализованы в сценарном прототипе. Когда речь идет о диалоге или изучении текстовых носителей, будь то записок или энциклопедий, их также следует включить в прототип. Выбор зависит от конкретного проекта, а также от целей прототипирования.

Иными словами, сценарный прототип – это черновая разработка, соответствующая по жанру разрабатываемой игре; она имеет “заглушки” на геймплей и игровые механики; разработана на целевом движке.

Соответствие сценарного прототипа жанру, а также его разработка на целевом движке – рекомендации, устраняющие возможные трудности с имплементацией сценарного прототипа в игру. К тому же, всегда проще нарастить функционал сценарного прототипа, чем разрабатывать игру с нуля. Иметь готовый сценарный прототип для каких-то игр означает иметь одну из итераций разработки. Таким образом, сценарный прототип сокращает объем работ в течение этапа разработки.

Сценарный прототип – это попытка на время изолировать историю от игры, чтобы протестировать её на такие характерные параметры интерактивного повествования, как погружение, темпоритм и эмпатию. Это решение, позволяющее сэкономить проектные ресурсы, протестировать историю до начала разработки контентной составляющей игры. Он применим как для небольших, так и для крупных проектов, как для разработки с нуля, так и для доработки дополнительных блоков существующей игры.

Разработка сценарного прототипа на основе работы сценариста, представленной в виде технической документации, – работа нарративного дизайнера¹.

В качестве обоснования актуальности данного подхода, приведем сравнительный анализ с другими форматами и практиками разработки (см. рис. 2). Критерии оценки для сравнительного анализа следующие: визуализация, время на изучение, поддержка разветвления, возможность правок, тестирование игрового опыта. Сравниваются инструменты друг относительно друга. Оценка зависит от названия колонки и означает объем возможности или количество необходимых

¹ Нарративный дизайнер – специализация гейм-дизайнера; дизайнер по имплементации истории в игру.

ресурсов: мало, средне, много. Цветом выделена степень эффективности: зеленый – отлично подходит для решения поставленных задач, оранжевый – подход имеет недостатки, красный – формат плохо подходит для работы.

Главная задача сравнения – показать, что сценарный прототип относится к эффективным практикам проектирования, разработки и тестирования интерактивного повествования.

№	Артефакт разработки	Визуализация	Время на изучение	Поддержка разветвления	Возможность правок	Тестирование игрового опыта
1	Текстовый документ	мало	много	мало	мало	мало
2	Wiki-разметка	средне	много	средне	мало	мало
3	Таблица	средне	средне	средне	средне	мало
4	Раскадровка	много	мало	мало	мало	мало
5	Бумажный прототип	средне	мало	много	много	много
6	Сценарный прототип	много	мало	много	много	много

Рис. 2. Сравнение инструментов сценариста

По результатам анализа можно прийти к следующим выводам:

- текстовый документ почти не поддерживает визуализацию, требует много времени на изучение, почти не поддерживает разветвление сюжета, его трудно править, и он совершенно не подходит для тестирования игрового опыта;
- wiki-разметка, как частный случай текстового документа, наследует большую часть проблем этого формата, однако поддерживает нелинейность в изучении текста, а также может содержать в себе больше визуализации;
- таблицы в целом применимы для разработки разветвленного повествования, но не подходят для его тестирования;
- раскадровка хорошо визуализирует историю и сокращает время изучения, но кроме этого больше никакими преимуществами не обладает; кроме

этого раскадровка бессмысленна без сопровождающего текстового документа, комментариев или устного объяснения художника и сценариста;

- бумажный прототип – принципиально иной уровень разработки и тестирования интерактивного повествования, однако визуализация почти отсутствует из-за ограничений данного формата;
- сценарный прототип – инструмент, который во многом подходит для разработки, проектирования и тестирования интерактивного повествования.

3. ГЕНЕРАЦИЯ СЦЕНАРНОГО ПРОТОТИПА

Задача сценариста – создать историю, которая найдет себя в игровом процессе. Задача нарративного дизайнера – внедрить историю в игру или игру в историю так, чтобы они оставались неразрывным целым на протяжении всего игрового процесса. Нарративный дизайнер ближе к непосредственной разработке, в отличие от сценариста, поэтому наличие навыков программирования для них – это значительное преимущество, а для сценариста это скорее лишняя нагрузка.

Создание сценарного прототипа подразумевает работу в игровом движке и иногда включает в себя программирование. По сути это прототипирование компонентов игры, а затем проработка его до финальной версии.

В сфере информационных технологий прототипирование – неотъемлемая часть разработки крупных и небольших проектов. Однако большая часть деятельности в данной сфере посвящена автоматизации. Автоматизация прототипирования – это то, что в перспективе сэкономит время, деньги и другие ресурсы на проекте. Это уместно и для видеоигр.

Генерация сценарного прототипа вместо ручной сборки поможет решить две задачи: 1) разгрузить нарративного дизайнера и 2) снизить порог вхождения нарративных дизайнеров в индустрию. Кроме этого оптимизация нагрузки на специалистов высвободит как финансовые, так и кадровые ресурсы, которые могут быть перераспределены в проекте более эффективным образом.

Первая задача связана с тем, что на предварительном этапе разработки нарративный дизайнер занимается не только тем, что собирает сценарный прототип. Кроме этого ему необходимо создать концепцию внедрения повествования в игру. Если снять с нарративного дизайнера одну задачу, он сможет сконцентрироваться на другой.

Вторая задача также частично связана с разгрузкой других специалистов. Например, если тот же сценарный прототип будет сгенерирован, нарративный дизайнер сможет самостоятельно находить какие-то нестыковки в истории и исправлять их раньше. Тем более при этом не будут задействованы такие специалисты, как художники, программисты и разработчики дорогостоящей трехмерной графики.

Таким образом сформулирована ценность инструмента: генератор сценарных прототипов может сэкономить все возможные ресурсы, которые есть на проекте по разработке игр. Более того, инструмент применим не только в игровой индустрии, но и в индустрии разработки serious games, тренажеров и симуляторов.

Основным сценарием использования генератора сценарного прототипа является следующий план действий: 1) ввести в инструмент текстовые данные, которые содержат в себе как наработки сценариста, так и наработки смежных специалистов, в том числе существующие наработки художника; 2) генератор анализирует и достраивает видение игры и формирует сценарный прототип, а также статистическую информацию и визуализацию структур повествования; 3) пользователь получает результаты генерации, скачивает их и отправляет на тестирование; при необходимости вносит правки.

Более подробно видение инструмента представлено в разделе 5.

4. СУЩЕСТВУЮЩИЕ РЕШЕНИЯ ВИЗУАЛИЗАЦИИ

Сценарный прототип – комплексный продукт, как и видеоигры, которые могут включать в себя текст, визуализацию, звук, интерактивное взаимодействие и возможность выбора.

В рамках данной работы был проанализирован ряд работ, которые так или иначе затрагивают тему эффективной, быстрой и функциональной визуализации как в автоматизированной, так и в ручном форме.

Инструменты разделены на генераторы визуализации из текста и другие инструменты, которые повлияли на видение инструмента генерации сценарного прототипа видеоигр.

4.1. Генерация визуализации из текста

Существует множество решений для генерации визуального контента на основе текста на естественном языке, которые отличаются целями и подходами. Так, например, в работе 2016 года [18] рассматриваются 26 реализованных инструментов. Они делятся на несколько типов: преобразование текста в серию картинок (Story Picturing Engine), преобразование текста в серию ассоциированных картинок (Text-to-Picture Synthesis System), преобразование текста в набор статичных сцен (TEXT-TO-SCENE), преобразование текста в анимацию, в том числе с учетом положения объектов в пространстве (TEXT-TO-ANIMATION).

Точность генерации визуализации во многом зависит от степени развития NLP и нейросетей, а также вычислительных мощностей компьютера. Однако уже сейчас есть приемлемые результаты генерации, с которыми можно работать в индустрии кинематографа [19] и мультипликации (см., например, CRAFT [20], Storyboarder [21]).

В рамках настоящей работы были обозначены инструменты, которые доказывают жизнеспособность идеи генерации визуализации из текста, а также подсказывают пути развития инструментов ScriptViz [22], SceneMaker [23], CRAFT, Storyboarder.

ScriptViz генерирует трехмерную анимацию в режиме реального времени. Пользователь вводит текст, учитывая следующие ограничения: текст должен быть написан на английском, иметь понятные формулировки и хорошую структуру. Инструмент ограничен базой трехмерных моделей, в генерации участвуют готовые объекты и анимации.

SceneMaker – это инструмент, который автоматизирует генерацию ряда артефактов для кинопроизводства. Генерация сцен из сценария планировалась с точностью до мимики виртуальных актеров, освещения и аудиосопровождения. Проект не был реализован.

CRAFT – нейросеть, обученная на 25 184 фрагментах мультсериала Флинстоуны. Нейросеть способна сгенерировать новые серии этого сериала по текстовому описанию. В собранных сценах ещё встречаются ошибки, такие как неправильное наложение объектов или неправильный выбор анимаций, но в целом разработка интересна.

Storyboarder – инструмент для создания раскадровок от компании Wonder Unit. В данном инструменте есть функционал генерации статичных кадров на основе введенных текстовых запросов. Однако стоит отметить, что Storyboarder понимает ограниченный словарь слов и некоторые альтернативные названия.

4.2. Другие инструменты визуализации

Перечисленные выше инструменты подтверждают гипотезу о том, что техническую документацию игровых сценаристов возможно визуализировать программным путем. Однако ключевое отличие прототипирования повествования игр заключается в реализации возможности интерактивного взаимодействия, а также вариативности сюжета, чего не подразумевают эти инструменты, применимые скорее для кинематографа и мультипликации.

Были найдены инструменты, которые возможно адаптировать для разработки игр: StoryFlow [24], Machination Tool [25], Orange3 [26].

StoryFlow позволяет представить структуру в виде Yarn. Она визуализирует вариативность в происходящих событиях, что крайне полезно для красивой и наглядной демонстрации игровых событий, взаимодействия персонажей и определения временных промежутков.

Machination Tool – это приложение для моделирования и балансирования игровых систем. Данный инструмент позволяет визуализировать и симулировать игровые системы в виде динамических диаграмм. Данное решение может быть одним из элементов уже не сценарного, а скорее игрового прототипа.

Orange3 – это программный пакет визуального программирования на основе компонентов для визуализации данных, машинного обучения и интеллектуального анализа данных. В программу включены сотни готовых нод, каждая из которых отвечает за свою часть работы: различные визуализаторы (схемы, графики, таблицы), алгоритмы обработки и препроцессинга текста, готовые к обучению и работе нейросети. Помимо этого, разработчики предоставляют пользователям возможность реализовать собственный функционал в виде программ на Python, для которых есть отдельный нод.

5. ВИДЕНИЕ ИНСТРУМЕНТА ГЕНЕРАЦИИ СЦЕНАРНОГО ПРОТОТИПА

Для представления видения генератора сценарного прототипа разработана схема состава инструмента, а также проанализировано влияние наличия такого инструмента на процесс разработки видеоигр.

5.1. Схема и видение инструмента

На рис. 3 представлена схема инструмента, которая была разработана для последующей реализации.



Рис. 3. Схема инструмента генерации сценарного прототип

На вход инструмент принимает текст, который написан на естественном языке. Допускается, что документы будут удовлетворять некоторым правилам форматирования для лучшего распознавания сущностей. В качестве текста подразумевается техническая документация сценариста, представленная в различных форматах.

Инструмент разделен на подсистемы: UI – пользовательский интерфейс и генератор сценарного прототипа.

Через пользовательский интерфейс пользователь будет иметь доступ к следующему функционалу, который разделен на модули:

- Модуль ввода принимает на вход текстовые документы и таблицы, а также другие форматы, в которых могут храниться данные: html, xml, json.
- Модуль редактирования позволяет вводить и редактировать текстовый документ, редактировать сгенерированный прототип и структуру повествования.
- Модуль отображения демонстрирует результаты визуализации структуры повествования, а также информацию об извлеченных сущностях (персонажах, локациях и т. п.) и статистике по проекту (количество персонажей, локаций и т. п.).
- Модуль запуска позволяет пользователю запускать сценарный прототип, в который можно сыграть. Кроме этого возможен запуск сгенерированной балансной диаграммы.

Генератор сценарного прототипа разделен на модули по типам процессов:

- Модуль анализа текста анализирует входные данные с помощью алгоритмов NLP и машинного обучения. На выходе из него ожидаются перечень сущностей с типами, а также связи между сущностями.
- Модуль анализа данных и сборки структуры переводит информацию в понятный для алгоритмов генерации. На этом этапе обработки подсистема генерации сценарного прототипа включает в себя три модуля, каждый из которых последовательно обрабатывает результат работы предыдущих модулей:
 - а. Модуль анализа текста распознает необходимые сущности и связи между ними с помощью инструментов NLP;
 - б. Модуль анализа данных и построения структуры устанавливает целостную структуру сценария, связывает сущности между собой и строит структуру сценарного прототипа;
 - с. Модуль визуализации включает в себя четыре процесса: визуализация разветвленной структуры, постановка кадра, генерация окружающего пространства и генерация события в пространстве. Модуль получает на вход информацию в формализованном виде. На выходе выдает графические артефакты, ко-

которые можно воспроизвести в пользовательском интерфейсе. Кроме этого планируется, что в этом модуле будет происходить сборка балансных диаграмм на манер Machination.

Был составлен следующий план разработки инструмента генерации сценарного прототипа:

- визуализация структуры игрового сценария, представленного в виде текста на естественном языке (далее называемый TEXT) в виде графов или yarn-представлений;
- автоматическая сборка интерактивной раскадровки (для ограниченного количества жанров в виде визуальной новеллы или point-and-click игры) сценария игры из текста;
- создание алгоритма для генерации интерактивного сценарного прототипа из текста в игровом движке Unity;
- автоматическая генерация диаграммы баланса Machination на основе текста;
- автоматическая сборка игрового прототипа игры из текста.

Весь перечисленный функционал мы видим в рамках целостного приложения, на вход которому поступает серия текстовых документов, а на выходе получается проект, который объединяет в себе сгенерированный сценарный и, в перспективе, игровой прототип игры.

5.2. Влияние инструмента на разработку видеоигр

Процесс работы с повествовательной составляющей игры с применением нашего инструмента видится следующим:

- написание сценария игры в виде обычного текста на естественном языке в форме технической документации (текст, таблицы, wiki-разметка);
- генерация сценарного прототипа на основе документации;
- презентация и обсуждение сценарного прототипа внутри команды, с заказчиком и тестирование на фокус-группе;
- редактирование сценарного прототипа.

6. ПРОГРЕСС В РАЗРАБОТКЕ ИНСТРУМЕНТА

Решению задачи создания генератора сценарных и игровых прототипов посвящена серия наших разработок. В работе [27] были описаны собственный подход к созданию генератора сценарного прототипа и пилотная реализация инструмента для создания сцен на основе извлеченных из текста сущностей. В [28] представлена (см. рис. 4) программа для Orange 3, которая принимает на вход формализованный текст, а на выходе показывает различную визуализацию структуры: геометрическую связанность локаций, в каких локациях появляется персонаж, какие реплики произносятся в этих сценах и т. д.

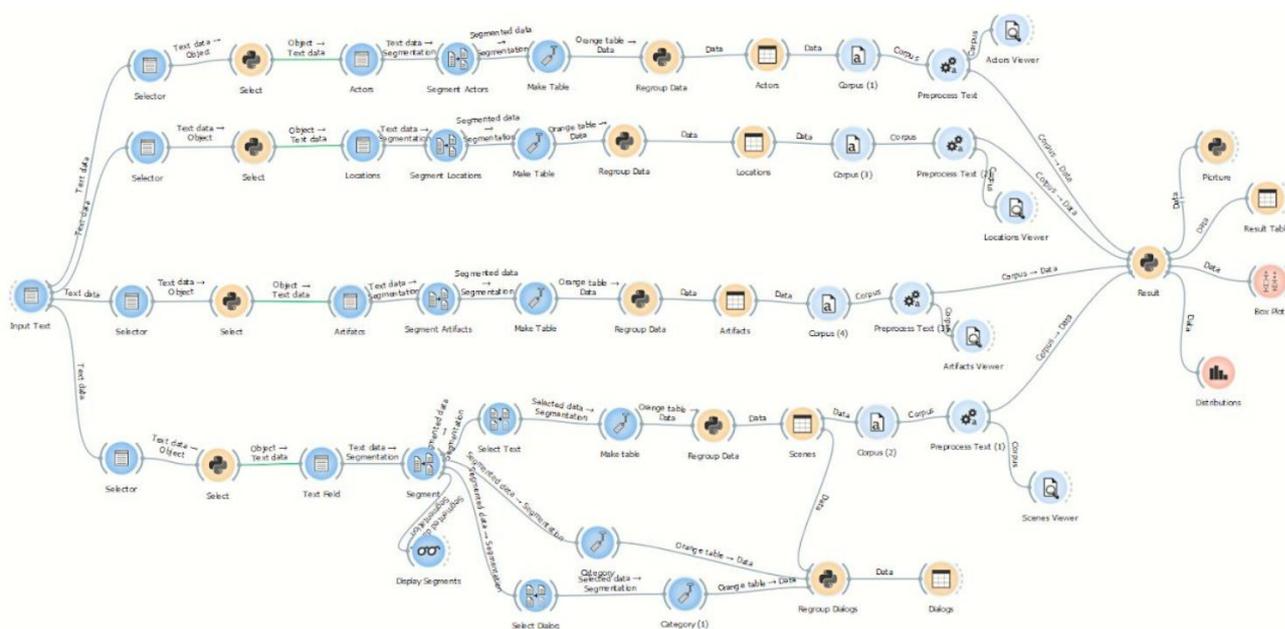


Рис. 4. Нодовая система генерации визуализации сценария

Другая работа (см., например, [29], [30]) связана с решением задачи извлечения информации о постановке кадра из текста. Для этого создана система анализа текста и вычисления необходимых настроек камеры из контекста. Работа находится на раннем этапе: воссоздается функционал текстового распознавания кадра, аналогичный Storyboarder (см. рис. 5).

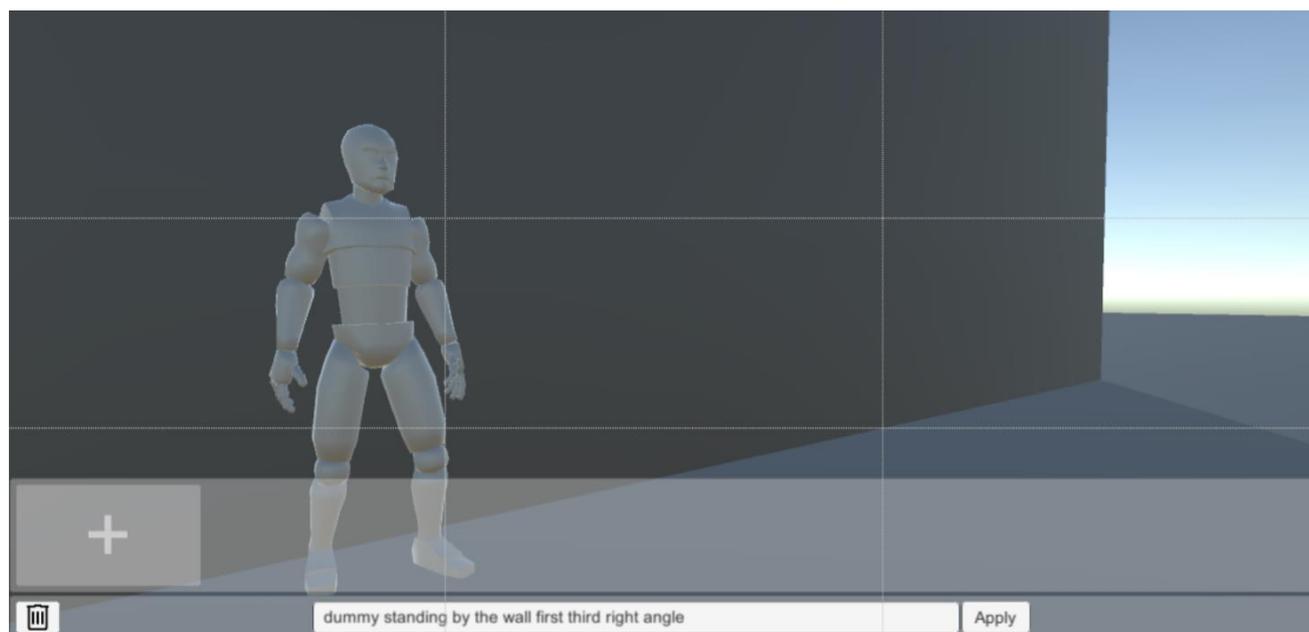


Рис. 5. Извлечение информации о кадре из текста

7. ВИДЕНИЕ ПРОЦЕССА ГЕНЕРАЦИЯ СОБЫТИЙ В ПРОСТРАНСТВЕ

Одна из задач модуля визуализации генератора сценарного прототипа – генерация визуализации событий в пространстве. Для этого необходима информация о персонажах в сцене, их действиях и репликах.

В рамках визуализации событий в пространстве персонажи должны располагаться в позах, соответствующим контексту из текста. На выходе должна получиться серия кадров, которая составляет из себя раскадровку. Данный шаг необходим как один из этапов разработки интерактивной раскадровки, в которой переход к следующему кадру будет зависеть от выбора игрока. В качестве платформы для экспериментов был выбран инструмент Storyboarder, который содержит необходимый функционал генерации визуализации.

Принцип работы состоит в следующем: на вход отправляется текст на английском языке, предложения которого представляют собой описание отдельных сцен. В каждом предложении есть информация о том, кто присутствует в сцене и что он делает. Текст анализируется и из него извлекается информация для визуализации – персонаж и его действие. Результат обрабатывается и отправляется на вход Storyboarder, где генерируются кадры, которые собираются в раскадровку для оценки результата.

Далее представлены подробности технической реализации и результаты эксперимента.

8. РЕАЛИЗАЦИЯ ПРОЦЕССА ГЕНЕРАЦИИ СОБЫТИЙ В ПРОСТРАНСТВЕ

Процесс генерации события в пространстве по текстовому вводу представляет собой автоматический конвейер. В нем происходят прием текста и его обработка. Затем результат отправляется на генерацию раскадровки в Storyboarder.

Возможность генерации элементов раскадровки по заданному вектору слов присутствует только в слепке коммита `abbf5f24` открытого официального репозитория Storyboarder [31]. По этой причине в рамках эксперимента использована именно эта версия, а не последняя актуальная.

Для того чтобы интегрировать Storyboarder в конвейер, был изолирован функционал, который позволяет вызывать процедуру генерации с аргументом в виде текста и инициировать все необходимые механизмы для получения раскадровки. Вход – это текст, находящийся в управлении другого программного компонента, а выход – это сгенерированная раскадровка.

Также был внедрен микро веб-сервер `express`, который принимает на `localhost` интерфейсе запрос с параметром в виде входного текста, передает данный параметр в изолированный функционал и в качестве ответа отправляет тот текст, что получил, чтобы отправитель удостоверился в успешности операции генерации.

Исходный текст преобразуется в корпус [32] – это отобранная и обработанная по определенным правилам совокупность текстов, используемых в качестве базы для исследования языка. Он используется для статистического анализа и проверки статистических гипотез и подтверждения лингвистических правил в данном языке.

Так как машинное обучение в большинстве своем имеет дело с функционалами (любыми функциями, где образ является числом или множеством чисел), для обработки текста его приходится векторизовать. Этот процесс означает преобразование чисел в векторы по определенным правилам, таким, чтобы потеря информации, заложенной в тексте, была минимальной.

Также одним из этапов обработки текста является выделение стоп-слов – таких слов, которые в рамках генеральной совокупности играют малую роль в исследовании свойств языка. Например, в английском языке это будут слова “a”, “am”, “an”, “is”, “are” и т. д. В русском языке таковыми будут являться “же”, “то”, “бы” и т. д.

Для того чтобы Storyboarder корректно сгенерировал раскадровку, входной текст необходимо обработать. Первым этапом будет являться токенизация, а именно, преобразование текста в корпус в виде массива предложений; также каждое предложение необходимо разбить по словам и убрать пунктуацию. Во второй этап входит фильтрация всех слов в корпусе, в результате которого корпус будет очищен от стоп-слов. Далее стоит задача распознавания имён собственных и преобразования их в обезличенную женскую или мужскую сущность. Это необходимо сделать, так как Storyboarder не может различать имен, но может различать пол сущностей, преобразуя его в кадре в модель мужчины или женщины.

После проведения токенизации, во время этапа фильтрации, были использованы стоп-слова, собранные в приложении к работе [33]. Процесс фильтрации происходил итеративно, слова сравнивались посимвольно полно. На следующем этапе алгоритма необходимо построить классификатор, который будет распознавать в тексте имена собственные и определять их пол.

В процессе разработки в качестве модели выбран мультиклассовый наивный байесовский классификатор, так как закономерность выбора имен в обществе отсутствует и имена даются независимо друг от друга. Имена для обучения классификатора взяты из публичных данных Службы социального обеспечения США [34]. Для каждого из имени в обучающей выборке происходят векторизация в символьные биграммы, а также частотный анализ, исходя из которого виден процент использования имени в качестве женского и мужского. Так как обучение наивного байесовского классификатора – это лишь вычисление независимых вероятностей, произведение которых лежит в знаменателе формулы по теореме Байеса [35], трата времени на процесс минимальна.

Обученный байесовский классификатор отвечает нулём, если имя, пришедшее на вход, женское, и единицей, если мужское. Используя операцию классификации, каждый раз, когда в тексте появляется имя собственное, оно заменяется на слово *man* или *woman* в зависимости от пола, который присвоил ему обученный классификатор. Обезличенные слова выбраны как наиболее понятные для парсера предложений Storyboarder.

9. РЕЗУЛЬТАТ РАБОТЫ ПРОЦЕССА ВИЗУАЛИЗАЦИИ

9.1. Пример визуализации

Стоит отметить, что текст на естественном языке написан с оговоркой, что Storyboarder понимает ограниченный список слов. Важно, чтобы в сцене появлялся персонаж с анимацией, возможной в рамках Storyboarder. Настройки камеры задаются вручную последовательностью параметров, которые понимает Storyboarder. Итак, на вход отправляется следующий текст:

Bob is walking.

Alice says hi.

Bob is walking and looking back.

Bob is walking.

Alice hangs one arm.

Alice crossing hands.

Далее, в зависимости от контекста предложения, формулируются необходимые настройки камеры. Они хранятся в отдельном файле:

looking forward, medium long, single person, centered, left angle, eye level, long lens, light, frontrightlit, outside;

looking forward, medium long, single person, centered, right angle, eye level, long lens, light, frontrightlit, outside.

Список процессов преобразования, которые осуществляются над текстом:

1. Токенизация с очищением от пунктуации и стоп-слов.
2. Распознавание и обезличивание имен собственных путем замены на слова *man/woman* в зависимости от пола имени с помощью наивного байесовского классификатора.

3. Полученный после преобразования текст дополняется настройками камер с помощью конкатенации строк.

Таким образом формируется результат, подающийся на вход в Storyboarder:

'man walking, looking forward, medium long, single person, centered, left angle, eye level, long lens, light, frontrightlit, outside';

'woman say hi, looking forward, medium long, single person, centered, right angle, eye level, long lens, light, frontrightlit, outside';

'man walk look back, looking forward, medium long, single person, centered, left angle, eye level, long lens, light, frontrightlit, outside';

'man walking, looking forward, medium long, single person, centered, left angle, eye level, long lens, light, frontrightlit, outside';

'woman hang one arm, looking forward, medium long, single person, centered, left angle, eye level, long lens, light, frontrightlit, outside';

'woman crossing hands, looking forward, medium long, single person, centered, left angle, eye level, long lens, light, frontrightlit, outside'.

Storyboarder генерирует раскадровку на основании полученных данных (см. рис. 6).

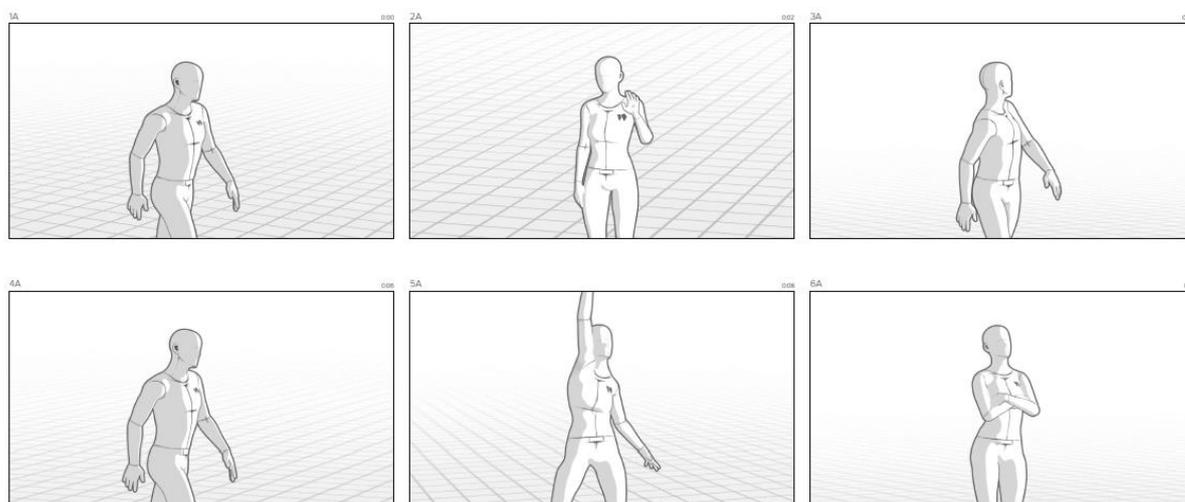


Рис. 6. Пример сгенерированной раскадровки

Возможности для генерации комикса в Storyboarder позволяют использовать его в случаях с подготовленным текстом. Расширение функционала и корпуса

Storyboarder позволит эффективнее работать с визуализацией натурального текста.

В дальнейшем планируется разработка собственного инструмента, а не доработка существующих решений.

9.2. Эксперимент

В качестве эксперимента был взят сценарий существующего проекта [36]. Сценарий был переписан в вид, в котором есть вся необходимая информация для генерации кадра в Storyboarder. Данный текст был предоставлен для ручной сборки раскадровки в Storyboarder. Настройки камеры использовались такие же, как и для генерации раскадровки.

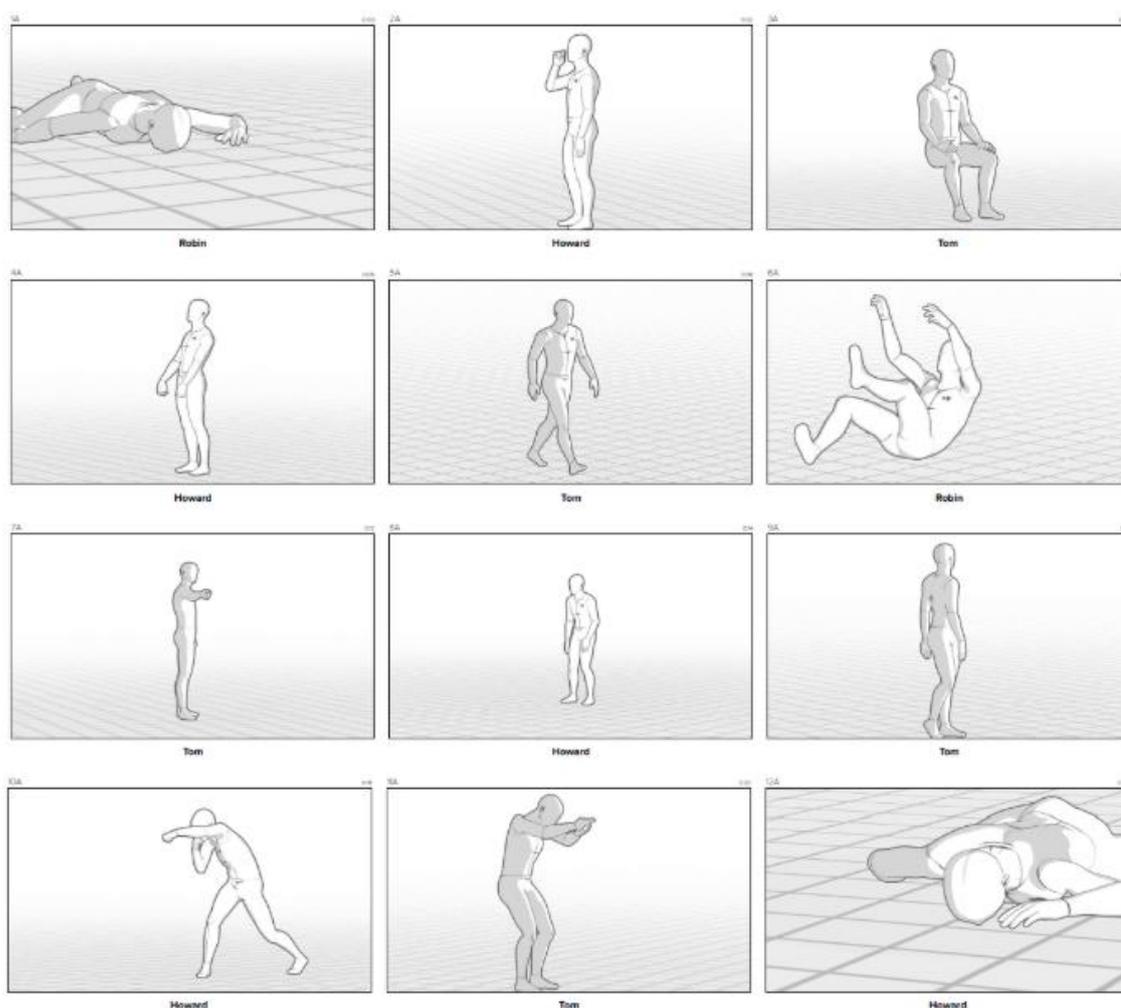


Рис. 7. Результат генерации для эксперимента

Вместе с воспроизведением процесс ручного создания раскадровки занял 236 секунды или 3 минуты и 56 секунд. Далее для сравнения текст был отправлен на автоматическую генерацию кадров. Раскадровка идентична той, что получена вручную (см. рис. 7). Вместе с воспроизведением процесс занят 42 секунды, что в 5,6 раз быстрее ручной сборки. Стоит отметить, что чем больше проект, тем дольше собирать его вручную.

Гипотеза о том, что инструмент для генерации раскадровок является одним из эффективных инструментов для создания визуализации сценарного прототипа, подтверждается.

10. ДАЛЬНЕЙШЕЕ РАЗВИТИЕ ИНСТРУМЕНТА

Сложность дальнейшего развития инструмента заключается в том, что интерактивное повествование по умолчанию разветвленное и содержит в себе возможность выбора. Задача извлечения подобных зависимостей реальна при применении технологий машинного обучения. Для этого обучения нейросетей необходимы корпуса, которых на данный момент не существует.

В дальнейшем разработка должна перейти с прототипных и экспериментальных решений в форму самостоятельного инструмента, разработанного с нуля.

В ходе обсуждения результатов разработки были сформулированы задачи на будущее:

- Извлечение информации о постановке кадра из текста;
- Разметка и форматирование технической документации;
- Создание интерактивной раскадровки;
- Генерация визуализации.

Нереализованной осталась задача постановки кадра в зависимости от контекста повествования. Данный вопрос важен с точки зрения автоматизации генерации визуализации сценария, и уже начата разработка в этом направлении.

Мы не касались темы классической разметки сценарных документов для неинтерактивных произведений и особенностей сценариев для интерактивных произведений. Данному вопросу мы планируем посвятить время в дальнейшем. Решение вопроса разметки и форматирования документов с нелинейным повествованием поможет решить следующую задачу, связанную с интерактивностью раскадровки.

Раскадровки хороши в кинематографе и анимации, а также в играх, например, в создании кат-сцен. Главное изменение, которое мы хотим внести в покадровое представление игрового повествования, – это возможность переключаться с помощью взаимодействия с игровой действительностью. Интерактивная раскадровка отдаленно напоминает нам визуальную новеллу или игры жанра point-and-click. В игре повествование движется за счет действий игрока. Исключая из прототипа влияние геймплея и выбор действий, мы лишаем сценаристов возможности тестирования повествования в условиях реальной разработки. В сценарном прототипе интерактив может быть заменен геймплейными “заглушками”, которые однозначно трактуют одну из веток развития событий. С таким инструментом игровой сценарист и нарративный дизайнер смогут заранее «отыграть» сюжет игры, показать его разработчикам, заказчику и пользователям. Возможность оценки и тестирования сценарного прототипа поможет оценить черты игр, как классические для линейного повествования, так и характерные для интерактивного повествования: погружение, темпоритм, формирование эмпатии. Таким образом, следующей задачей мы ставим приведение раскадровки к интерактивному виду.

Вопрос о генерации визуализации по текстовому описанию связан с тем, что необходимо заранее создать необходимые трехмерные объекты, анимации для них, логику сборки сцены, непротиворечивые правила взаимодействия между объектами. При этом каждая из перечисленных единиц сборки должна иметь свойство, схожее со свойством вектора в базисе линейного пространства. Другими словами, количество созданных единиц должно быть минимальным при условии, что вариативность генерируемых сцен должна быть максимальной. Данный момент необходимо учитывать в последующих разработках.

После усовершенствования прототипа планируется продолжить тестирование, а также сформулировать метрики, которые покажут, насколько генерация сценарного прототипа эффективнее ручного прототипирования.

ЗАКЛЮЧЕНИЕ

В работе введен термин сценарного прототипа и описано авторское представление о нём. Помимо этого, сформулирована актуальность инструмента генерации сценарного прототипа.

В работе приведен анализ инструментов, которые вдохновили авторов. Существование этих инструментов говорит о том, что разработка инструмента генерации сценарного прототипа возможна. Данный факт подтверждают примеры инструментов коллег, которые были разработаны с участием авторов статьи (см. в разделе 6).

В качестве эксперимента был разработан прототип компонента визуализации инструмента генерации сценарного прототипа. Прототип генерирует раскладку на основе текста. Компонент призван сэкономить время на визуализацию игрового повествования.

В результате проделанной работы были детализированы требования к инструменту генерации. Размышления на тему развития инструмента привели к формированию ряда задач для следующей итерации прототипирования.

Данная разработка интересна в рамках использования для разработки не только игр, но и serious games, симуляторов и виртуальных обучающих тренажеров, которым также посвящено наше пристальное внимание (см., например, [36–38]).

СПИСОК ЛИТЕРАТУРЫ

1. Open-source tool for telling interactive, nonlinear stories Twine.
URL: <https://twinery.org/>.
2. Visual novel engine Ren'Py. URL: <https://www.renpy.org/>.
3. The solution for interactive storytelling and game content management articy:draft 3. URL: <https://www.articy.com/en/>.
4. Real-time 3D development platform Unity. URL: <https://unity.com/>.
5. Real-time 3D creation tool Unreal Engine.
URL: <https://www.unrealengine.com/en-US/>.
6. An Online Visual Collaboration Platform for Teamwork Miro.
URL: <https://miro.com/>.

7. Diagram Software and Flowchart Make diagrams.net.
URL: <https://www.diagrams.net/>.
8. *Davies D., Bathurst D., Bathurst R.* The Telling Image: The Changing Balance between Pictures and Words in a Technological Age // *Technology and Culture*. 1992. No. 4. P. 845–846.
9. *Raja D., Bowman D., Lucas J., North C.* Exploring the benefits of immersion in abstract information visualization // *Proc. of the 8th Int'l Immersive Projection Technology Workshop*. 2004.
URL: https://people.cs.vt.edu/bowman/papers/ipt_dheva.pdf.
10. *Montfort N., Bogost I.* *Racing the Beam: The Atari Video Computer System*. Cambridge: MIT Press, 2009. 192 p.
11. *Bogost I.* Videogames are a Mess.
URL: http://bogost.com/writing/videogames_are_a_mess/.
12. *Cairns P., Cox A., Nordin I.* Immersion in Digital Games: Review of Gaming Experience Research // *Handbook of Digital Games*. 2014. P. 337–361.
13. *Newman J.* *Videogames*. Routledge: Psychology Press, 2014. 198 p.
14. *Ianchard L.* *Creating empathy in video games*. The University of Dublin, 2016. 42 p.
15. *Koivisto E., Suomela R.* Using prototypes in early pervasive game development // *Computers in Entertainment*. 2008. No. 17.
16. Open-source plugin for Unity Cradle.
URL: <https://forum.unity.com/threads/released-cradle-play-twine-stories-in-unity.333720/>.
17. Point-and-click. URL: <https://dic.academic.ru/dic.nsf/ruwiki/1443730>.
18. *Hassani K., Lee W.-S.* Visualizing Natural Language Descriptions: A Survey // *ACM Computing Surveys*, 2016. URL: <https://arxiv.org/pdf/1607.00623.pdf>.
19. Artificial intelligence tools RivetAI. URL: <https://www.rivetai.com/>.
20. *Gupta T., Schwenk D., Farhadi A., Hoiem D., Kembhavi A.* *Imagine This! Scripts to Compositions to Videos*. Cornell University, 2018.
URL: https://openaccess.thecvf.com/content_ECCV_2018/papers/Tanmay_Gupta_Imagine_This_Scripts_ECCV_2018_paper.pdf.
21. Storyboarder. URL: <https://wonderunit.com/storyboarder/>.

22. *Liu Z.-Q., Leung K.-M.* Script visualization (ScriptViz): A smart system that makes writing fun. Switzerland: Springer, 2006. P. 34–40.
 23. *Akser M., Bridges B., Campo G., Cheddad A., Curran K., Fitzpatrick L., Hamilton L., Harding J., Leath T., Lunney T., Lyons F., Ma M., Macrae J., Maguire T., McCaughey A., McClory E., McCollum V., Mc Kevitt P., Melvin A., Moore P., Mulholland E., Muñoz K., O’Hanlon G., Roman L.* SceneMaker: Creative technology for digital storytelling. Switzerland: Springer, 2016. P. 29–38.
 24. *Padia K., Bandara K., Healey C.* A system for generating storyline visualizations using hierarchical task network planning // *Computers & Graphics*. 2019. V. 78. P. 64–75.
 25. *Adams E., Joris D.* The Designer's Notebook: Machinations, A New Way to Design Game Mechanics.
URL: https://www.gamasutra.com/view/feature/176033/the_designers_notebook.
 26. Orange 3. URL: <https://orange.biolab.si/>.
 27. *Сахибгареева Г.Ф., Кугуракова В.В.* Концепт инструмента автоматического создания сценарного прототипа компьютерной игры // *Электронные библиотеки*. 2018. Т. 21. № 3-4. С. 235–249.
 28. *Доброквашина А.С., Газизова Э.А.* Автоматизация проектирования игрового прототипа на основании обработки формализованного игрового дизайн-документа // *Ученые записки ИСГЗ*. 2019. Т. 17. № 1. С. 583–589.
 29. *Астафьев А.М.* Разработка инструмента для сборки сцен по тегам // *Казанский (Приволжский) федеральный университет*, 2019. 31 с.
URL: https://kpfu.ru/student_diplom/10.160.178.20_FPEBER9KDIZQVYJAE3VRTIFYWZB_CDDM972OPP2I28S0EEFABT_Astafev.pdf.
 30. *Кугуракова В.В., Сахибгареева Г.Ф., Нгуен А.З., Астафьев А.М.* Пространственная ориентация объектов на основе обработки текстов на естественном языке для генерации раскадровок // *Электронные библиотеки*. 2020. Т. 23. № 6. С. 1214–1238.
 31. GitHub Storyboarder. URL: <https://github.com/wonderunit/storyboarder>.
 32. *Николаев И.С., Митренина О.В., Ландо Т.М.* Прикладная и компьютерная лингвистика. М.: URSS, 2016. 320 с.
-

33. *Nothman J., Qin H., Yurchak R.* Stop Word Lists in Free Open-source Software Packages // Proc. Workshop for NLP Open Source Software. 2018. P. 712.

34. *Baby Names from Social Security Card Applications.*
URL: <https://catalog.data.gov/dataset/baby-names-from-social-security-card-applications-national-level-data>.

35. *McCreery C.* First-year Statistics for Psychology Students Through Worked Examples. 1. Probability and Bayes' Theorem. Oxford Forum, 2018. 29 p.

36. *Антонов И.О., Зезегова К.В., Кугуракова В.В., Лазарев Е.Н., Хафизов М.Р.* Программирование запахов для виртуального осмотра места происшествия // Электронные библиотеки. 2018. Т. 21. № 3-4. С. 301–313.

37. *Abramov V., Kugurakova V., Rizvanov A., Abramskiy M., Manakhov N., Evstafiev M., Ivanov D.* Virtual Biotechnological Lab Development // BioNanoScience. 2017. V. 7. No. 2. P. 363–365.

38. *Kugurakova V., Abramov V., Sultanova R., Tsvil'skiy I., Talanov M.* Virtual Reality-Based Immersive Simulation for Invasive Surgery Training // European Journal of Clinical Investigation. 2018. V. 48. P. 224–225.

STORYBOARD AS ONE OF THE REPRESENTATIONS OF THE SCENARIO PROTOTYPE OF COMPUTER GAMES

G. F. Sahibgareeva^{1, [0000-0003-4673-3253]}, **O. A. Bedrin**^{2, [0000-0003-3300-4318]},
V. V. Kugurakova^{3, [0000-0002-1552-4910]}

^{1, 2, 3}*Institute of ITIS, Kazan Federal University*

¹*gulnara.sahibgareeva42@gmail.com*, ²*simplavero@gmail.com*,

³*vlada.kugurakova@gmail.com*

Abstract

The work is devoted to the study and improvement of the design, development, and testing of video game storytelling. The existing practices of writing and keeping up-to-date scripts for interactive works have been studied. The definition of a scenario

prototype and requirements for its form are formulated. An idea was put forward about the efficiency of automating the creation of a scenario prototype in the form of a generator tool. A vision of such a tool has been drawn up. The impact of such a tool on development order is presented. Implemented a tool component and conducted an experiment that proves its effectiveness with an example such as generating storyboards from the text. Plans for future development have been formulated.

Keywords: *computer games, video game development, interactive storytelling, scenario prototype, narrative design, screenwriting, game documentation, storyboarding, storyboard generation, interactive storyboarding.*

REFERENCES

1. Open-source tool for telling interactive, nonlinear stories Twine.
URL: <https://twinery.org/>.
2. Visual novel engine Ren'Py. URL: <https://www.renpy.org/>.
3. The solution for interactive storytelling and game content management articy:draft 3. URL: <https://www.articy.com/en/>.
4. Real-time 3D development platform Unity. URL: <https://unity.com/>.
5. Real-time 3D creation tool Unreal Engine.
URL: <https://www.unrealengine.com/en-US/>.
6. An Online Visual Collaboration Platform for Teamwork Miro.
URL: <https://miro.com/>.
7. Diagram Software and Flowchart Make diagrams.net.
URL: <https://www.diagrams.net/>.
8. *Davies D., Bathurst D., Bathurst R.* The Telling Image: The Changing Balance between Pictures and Words in a Technological Age // *Technology and Culture*. 1992. No. 4. P. 845–846.
9. *Raja D., Bowman D., Lucas J., North C.* Exploring the benefits of immersion in abstract information visualization // *Proc. of the 8th Int'l Immersive Projection Technology Workshop*. 2004.
URL: https://people.cs.vt.edu/bowman/papers/ipt_dheva.pdf.
10. *Montfort N., Bogost I.* *Racing the Beam: The Atari Video Computer System*. Cambridge: MIT Press, 2009. 192 p.

11. *Bogost I.* Videogames are a Mess.
URL: http://bogost.com/writing/videogames_are_a_mess/.
12. *Cairns P., Cox A., Nordin I.* Immersion in Digital Games: Review of Gaming Experience Research // Handbook of Digital Games. 2014. P. 337–361.
13. *Newman J.* Videogames. Routledge: Psychology Press, 2014. 198 p.
14. *lanchar d L.* Creating empathy in video games. The University of Dublin, 2016. 42 p.
15. *Koivisto E., Suomela R.* Using prototypes in early pervasive game development // Computers in Entertainment. 2008. No. 17.
16. Open-source plugin for Unity Cradle.
URL: <https://forum.unity.com/threads/released-cradle-play-twine-stories-in-unity.333720/>.
17. Point-and-click. URL: <https://dic.academic.ru/dic.nsf/ruwiki/1443730>.
18. *Hassani K., Lee W.-S.* Visualizing Natural Language Descriptions: A Survey // ACM Computing Surveys, 2016. URL: <https://arxiv.org/pdf/1607.00623.pdf>.
19. Artificial intelligence tools RivetAI. URL: <https://www.rivetai.com/>.
20. *Gupta T., Schwenk D., Farhadi A., Hoiem D., Kembhavi A.* Imagine This! Scripts to Compositions to Videos. Cornell University, 2018.
URL: https://openaccess.thecvf.com/content_ECCV_2018/papers/Tanmay_Gupta_Imagine_This_Scripts_ECCV_2018_paper.pdf.
21. Storyboarder. URL: <https://wonderunit.com/storyboarder/>.
22. *Liu Z.-Q., Leung K.-M.* Script visualization (ScriptViz): A smart system that makes writing fun. Switzerland: Springer, 2006. P. 34–40.
23. *Akser M., Bridges B., Campo G., Cheddad A., Curran K., Fitzpatrick L., Hamilton L., Harding J., Leath T., Lunney T., Lyons F., Ma M., Macrae J., Maguire T., McCaughey A., McClory E., McCollum V., Mc Kevitt P., Melvin A., Moore P., Mulholland E., Muñoz K., O’Hanlon G., Roman L.* SceneMaker: Creative technology for digital storytelling. Switzerland: Springer, 2016. P. 29–38.
24. *Padia K., Bandara K., Healey C.* A system for generating storyline visualizations using hierarchical task network planning // Computers & Graphics. 2019. V. 78. P. 64–75.

25. *Adams E., Joris D.* The Designer's Notebook: Machinations, A New Way to Design Game Mechanics.

URL: https://www.gamasutra.com/view/feature/176033/the_designers_notebook.

26. Orange 3. URL: <https://orange.biolab.si/>.

27. *Sahibgareeva G.F., Kugurakova V.V.* Koncept instrumenta avtomaticheskogo sozdaniya scenarnogo prototipa komp'yuternoj igry // *Jelektronnye biblioteki*. 2018. T. 21. № 3-4. S. 235–249.

28. *Dobrokvashina A.S., Gazizova Je.A.* Avtomatizacija proektirovanija igrovogo prototipa na osnovanii obrabotki formalizovannogo igrovogo dizajn-dokumenta // *Uchenye zapiski ISGZ*. 2019. T. 17. № 1. S. 583–589.

29. *Astaf'ev A.M.* Razrabotka instrumenta dlja sborki scen po tegam. Kazanskij (Privolzhskij) federal'nyj universitet, 2019. 31 s.

URL: https://kpfu.ru/student_diplom/10.160.178.20_FPEBER9KDIZQVYJAE3VRTI-FYWZB_CDDM972OPP2I28S0EEFABT_Astafev.pdf.

30. *Kugurakova V.V., Sahibgareeva G.F., Nguen A.Z., Astaf'ev A.M.* Prostranstvennaja orientacija ob#ektov na osnove obrabotki tekstov na estestvennom jazyke dlja generacii raskadrovok // *Jelektronnye biblioteki*. 2020. T. 23. № 6. S. 1214–1238.

31. GitHub Storyboarder. URL: <https://github.com/wonderunit/storyboarder>.

32. *Nikolaev I.S., Mitrenina O.V., Lando T.M.* Prikladnaja i komp'yuternaja lingvistika. M.: URSS, 2016. 320 s.

33. *Nothman J., Qin H., Yurchak R.* Stop Word Lists in Free Open-source Software Packages // *Proc. Workshop for NLP Open Source Software*. 2018. P. 712.

34. Baby Names from Social Security Card Applications.
URL: <https://catalog.data.gov/dataset/baby-names-from-social-security-card-applications-national-level-data>.

35. *McCreery C.* First-year Statistics for Psychology Students Through Worked Examples. 1. Probability and Bayes' Theorem. Oxford Forum, 2018. 29 p.

36. *Antonov I.O., Zezegova K.V., Kugurakova V.V., Lazarev E.N., Hafizov M.R.* Programirovanie zapahov dlja virtual'nogo osmotra mesta proisshestvija // *Jelektronnye biblioteki*. 2018. T. 21. № 3-4. S. 301–313.

37. Abramov V., Kugurakova V., Rizvanov A., Abramskiy M., Manakhov N., Evstafiev M., Ivanov D. Virtual Biotechnological Lab Development // BioNanoScience. 2017. V. 7, No. 2. P. 363–365.

38. Kugurakova V., Abramov V., Sultanova R., Tsvil'skiy I., Talanov M. Virtual Reality-Based Immersive Simulation for Invasive Surgery Training // European Journal of Clinical Investigation. 2018. V. 48. P. 224–225.

СВЕДЕНИЯ ОБ АВТОРАХ



КУГУРАКОВА Влада Владимировна – к. т. н., доцент кафедры программной инженерии Института ИТИС КФУ, руководитель НИЛ разработки AR/VR приложений и компьютерных игр. Сфера научных интересов – иммерсивность виртуальных сред, проблемы генерации реалистичной визуализации, различные аспекты проектирования игр, AR/VR, подходы к интерпретации UX.

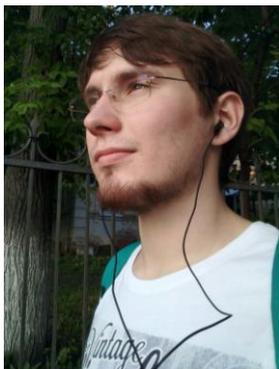
Vlada Vladimirovna KUGURAKOVA, PhD., Docent of the Institute ITIS KFU, Head of Laboratory «AR/VR applications and Gamedev». Research interests include immersiveness of virtual environments, problems of generating realistic visualization, various aspects of game design, AR/VR, approaches to UX interpretation.

vlada.kugurakova@gmail.com



САХИБГАРЕЕВА Гульнара Фаритовна – ассистент кафедры программной инженерии Института ИТИС КФУ. Сфера научных интересов – игровая сценаристика, нарративный дизайн, изучение вопроса эффективности создания сценарного прототипа и возможности автоматизации данного процесса.

Gulnara Faritovna SAHIBGAREEVA – assistant of the Department of Software Engineering of the Institute ITIS KFU. Research interests - game scripting, narrative design, studying the issue of the effectiveness of creating a scenario prototype and the possibility of automating this process. gulnara.sahibgareeva42@gmail.com



БЕДРИН Олег Александрович – бакалавр Института ИТИС КФУ.
Сфера научных интересов – машинное обучение, разработка видеоигр.

Oleg Aleksandrovich BEDRIN – bachelor of the Institute ITIS KFU. Re-
search interests – machine learning, development.
simplavero@gmail.com

Материал поступил в редакцию 17 ноября 2020 года