

Роберт Зубек
**Элементы гейм-дизайна. Как создавать игры, от которых
невозможно оторваться**

Мировой компьютерный бестселлер. Гейм-дизайн –

РОБЕРТ ЗУБЕК

ЭЛЕН ГЕЙМ ДИЗАЙН

КАК СОЗДАВАТЬ ИГРЫ,
ОТ КОТОРЫХ НЕВОЗМОЖНО
ОТОРВАТЬСЯ

МЕХАНИКИ

The background features a complex network of thin, light-colored lines connecting various points, creating a web-like or circuit-like structure. A prominent feature is a blue dot from which several lines radiate outwards, connecting to other points. The overall aesthetic is clean, technical, and modern, suggesting a focus on mechanics and design.

«Элементы гейм-дизайна. Как создавать игры, от которых невозможно оторваться»:
ISBN 978-5-04-170678-4

Аннотация

Хотите разобраться, в чем секрет успешных игр? Это вводное пособие по гейм-дизайну поможет вам освоить не только в теоретические основы, но и практические техники дизайна игр. Роберт Зубек, гейм-дизайнер и сооснователь независимой игровой студии в Чикаго, на конкретных примерах объясняет, что стоит за словом «гейм-дизайн» и как создать игру, которая понравится всем. Его подход базируется на трех столпах разработки игр, от которых зависит успех вашей игры. В этой книге вы найдете множество иллюстраций и схем, а также сможете выполнить индивидуальные и групповые упражнения, которые помогут вам стать настоящим профессионалом в разработке игр.

В формате PDF A4 сохранен издательский макет книги.

Роберт Зубек Элементы гейм-дизайна. Как создавать игры, от которых невозможно оторваться

<...>

Мэтту

За все

© Перфильев О.И., перевод на русский язык, 2022
© Оформление. ООО «Издательство «Эксмо», 2022

Благодарности

Идея этой книги возникла в ходе преподавания курса по разработке игр в Северо-Западном университете, поэтому первым делом хотелось бы поблагодарить моих студентов, для которых данный материал и был собран (изначально в виде заметок к курсу, прежде чем они превратились в книгу). Они стали «бета-тестировщиками» этого текста. Особые благодарности Иэну Хосвиллу и Северо-Западному университету за поддержку, а также помощнику Итэну Робинсону за его предложения относительно курса.

Также я благодарю коллег, которые любезно согласились просмотреть мое произведение, когда оно еще было в очень сыром виде, особенно Ноа Уордрипа-Фруина за обширные замечания и за «игровое тестирование» текста на его семинаре по дизайну, а также Крису Соларски и Рэнди О’Коннору за подробную критику. Дополнительные благодарности Эбиану Альваресу-Буюлле, Питеру Франклину, Тайгу Келли и Робу Локхарту за их мысли и отзывы, а также Гедиминасу Эйникису за иллюстрации.

Множество благодарностей всем коллегам, которые обсуждали со мной вопросы гейм-дизайна. Спасибо тем играм, что мы создавали вместе и по отдельности – я многое узнал о них (как на работе, так и в аспирантуре). Вас слишком много, чтобы перечислить всех, но вы знаете, о ком я говорю. Спасибо.

Наконец, я благодарю своего партнера Мэтта за то, что он поддержал меня в смелом решении написать эту книгу и за то, что подбадривал меня по вечерам и выходным, как только стало ясно, какую глубокую яму я вырыл сам себе. Но, как утверждает, однажды Леонард Бернстайн сказал: «Чтобы создать нечто грандиозное, необходимы два условия: план и недостаток времени». Возможно, нам, разработчикам, это послужит утешением, так

как именно последнего постоянно недостает.

Введение

Этот текст охватывает основы гейм-дизайна как процесса. Он предназначен для новичков в этой области, особенно для студентов вводных курсов.

Цель его практическая. Я хотел познакомить читателя с процессом создания новых игр и анализа существующих. В ходе знакомства мы будем рассуждать об играх на трех разных уровнях (чистой механики, динамики поведения и геймплея, конечного опыта игрока) и постараемся понять, как пользоваться этими уровнями для упрощения проблемы дизайна посредством разделения его на стадии.

Также в данной книге мне хотелось бы описать инструменты и приемы, которыми пользуются гейм-дизайнеры. Будучи сам разработчиком и гейм-дизайнером, я понимаю, насколько важно поделиться со студентами основами тех знаний, которые мы все сообща накопили в своей профессиональной деятельности. Наша индустрия молода, ей всего лишь несколько десятилетий, она быстро развивается – но при этом уже начинают формироваться кое-какие ее фундаментальные основы. Поэтому, пожалуй, сейчас наиболее подходящее время для обзора всего, чему мы научились, и для размышлений о том, что эти основы позволят создавать в будущем.

Гейм-дизайн и предмет данной книги

Игры – это результат синтеза многих дисциплин, и для того чтобы игрок получал отличный игровой опыт, все части должны хорошо работать вместе. Интересный и увлекательный геймплей, визуальный и звуковой дизайн, графика и музыка, дизайн игрового опыта в целях плавного взаимодействия, технический дизайн со средствами его реализации и многое другое.

Собственно, «гейм-дизайном» в игровой индустрии обычно называют дизайн геймплейного аспекта. Дизайн игры – это нечто вроде формулировки того, как игра должна *работать*. Как она будет разворачиваться, чем будет заниматься игрок, какие в игре персонажи, как участник взаимодействует с ними и тому подобное. Его дополняет понятие визуального дизайна – того, как игра будет выглядеть, или технического дизайна – как будут реализованы все эти аспекты, но гейм-дизайн при этом воспринимают как отдельный вид практики. Дизайн геймплея взаимодействует с другими элементами и сводит их вместе.

Такую широкую тему, как гейм-дизайн, можно освещать множеством разных способов, но в этой книге основное внимание фокусируется на нескольких специфических мотивах.

1. *Фокус на практических моделях и терминологии индустрии*. Мы будем пользоваться приемами и терминами, принятыми среди практикующих дизайнеров, работающих над коммерческими продуктами. В сообществе гейм-дизайнеров выработались многочисленные абстрактные и ментальные модели работы в данной сфере, как и своя собственная терминология, а эта книга должна познакомить нас с тем, как они используются на практике. Вышли также многочисленные теоретические и исследовательские работы, предлагающие дополнительные модели и термины; но мы по большей части оставим их в стороне – за исключением тех, что уже используются в практической деятельности.

2. *Фокус на играх как на динамических системах геймплея*. Мы будем рассматривать прежде всего геймплей и ориентированные на него элементы. Как было отмечено выше, современные игры не сводятся исключительно к геймплею: в них есть сюжет, персонажи, визуальный и звуковой дизайн и тому подобное. Но вместо того, чтобы разбирать все эти элементы, мы сконцентрируемся на самых существенных, без которых немислимы игры. Рассмотрим динамический процесс, разворачивающийся, когда игроки запускают игру и начинают взаимодействовать с ней, и как такого рода динамика геймплея порождает

задуманный игровой опыт. Что касается других элементов разработки игр, то им посвящены многие другие источники, повествующие об изобразительном искусстве, сюжете или техническом дизайне в играх.

3. *Фокус на процессе дизайна, а не на профессии дизайнера*. В соответствии с вышесказанным мы будем уделять основное внимание собственно дизайну геймплея, оставляя в стороне реалии работы гейм-дизайнера в целом. Работа в отдельном коллективе в рамках коммерческого предприятия сопряжена с дополнительными сложными задачами: коммуникацией в группе, лидерством в творческой сфере, порождением и отсеиванием дизайнерских идей, оценки их коммерческой целесообразности и тому подобными. Они, конечно, влияют на гейм-дизайн, но здесь мы по большей части будем их игнорировать, потому что описание всех этих задач коммерческого производства сделает рассуждение слишком сложным для наших целей.

4. *Фокус на гейм-дизайне, а не на дизайне игры как продукции*. Гейм-дизайн – это процесс определения того, как игра будет работать с точки зрения геймплея, а в дизайне продукции главное – насколько она подходит к рынку, как будет организован ее маркетинг, как игра будет представлена аудитории, как ее будут продавать, какие будут использованы модели продаж или монетизации, насколько программа окажется успешной на очень насыщенном рынке. Гейм-дизайн и дизайн продукции взаимозависимы, и опытные гейм-дизайнеры часто очень хорошо понимают, как требования к продукции влияют на гейм-дизайн и как инновации в нем могут открывать новые возможности для игры как для продукта. Но дизайн продукции – тоже достаточно сложная тема для данного вводного пособия.

Использование в образовательных целях

Эта книга рассматривается как вводное пособие для знакомства с элементарными понятиями гейм-дизайна. Она основана на моих записях, которые я вел во время чтения курса по разработке игр в Северо-Западном университете. При этом предполагается, что книга может вписываться в различные контексты – от более теоретических курсов гейм-дизайна до лабораторных практикумов по разработке игр, а также для самостоятельного обучения.

Начинается она с описания центральной модели – разного восприятия гейм-дизайна с точки зрения игрока и с точки зрения гейм-дизайнера, а также с перечисления трех уровней размышления об играх. В последующих главах отдельные части этой концепции исследуются более подробно. Впрочем, порядок глав допускает большую гибкость и его можно менять. Указанная последовательность отражает порядок моего вводного курса по разработке игр.

- Глава 1 знакомит с моделью.
- В главе 2 обсуждается опыт игрока, определяются цели дизайна.
- В главах 3 и 4 описываются способы анализа низкоуровневых деталей механики.
- В главах 5 и 6 описываются способы анализа высокоуровневых деталей геймплея, структуры и зависящего от них игрового опыта.
- В главе 7 описывается, как применять эти элементы при создании игрового прототипа.

При этом в других курсах и семинарах порядок глав может быть изменен. Например, в курсе, который больше сосредоточен на теории дизайна, последовательность может быть примерно следующей.

- Глава 1 знакомит с моделью.
- В главах 5 и 6 описываются способы анализа высокоуровневых деталей геймплея,

структуры и зависящего от них игрового опыта.

- В главах 3 и 4 описываются способы анализа низкоуровневых деталей механики.
- Главы 2 и 7 опускаются.

Таким образом, предполагается, что порядок глав зависит от потребностей каждого курса. Кроме того, книга настолько сжата, что преподаватели при необходимости без труда могут сочетать ее с дополнительными материалами о гейм-дизайне.

Замечание об упражнениях

Каждая глава сопровождается упражнениями, предназначенными для индивидуального выполнения, и заданиями, которые лучше выполнять в группе. Индивидуальные упражнения обычно предполагают анализ или рассуждения – например, предлагают читателям проанализировать известную им игру в контексте только что усвоенного материала данной главы или нескольких предыдущих. Как правило, эти упражнения носят открытый характер – с расчетом на то, что они будут использоваться в образовательном контексте, включая домашние задания или дискуссию во время семинара, а также с учетом различной подготовки студентов.

Пособие не содержит упражнений, в которых студентам на практике нужно разрабатывать целые игры или игровые прототипы, поскольку такие задания очень зависят от образовательного контекста. Они были бы очень разными для различных курсов – например, для ориентированных на цифровые игры в противоположность ориентированным на физические или для предполагающих знание программирования в противоположность тем, что не требуют этого и т.д. Читателям, интересующимся упражнениями по физическому дизайну, я искренне рекомендую работу Братуэйта и Шрайбера (Brathwaite and Schreiber, 2009).

Замечание о стиле

В данной книге я использую местоимение первого лица единственного числа «я» для обозначения себя как писателя, говорящего от своего имени, а местоимение первого лица множественного числа «мы» для обозначения себя и читателей, как если бы мы совместно исследовали какую-либо проблему или рассматривали игровые примеры. Такие фразы, как: «Итак, мы видим, что...» используются как риторические конструкции для обозначения траектории, которой следуем мы – писатель и читатель – в своем исследовании темы.

Но мне хотелось бы подчеркнуть, что «мы» однозначно не предполагает того, что я говорю от лица других гейм-дизайнеров или разработчиков игр. Помимо того что я не посмел бы выступать от их имени, это было бы просто невозможно в силу крайней разнородности постоянно развивающегося гейм-дизайнерского сообщества. Прошу рассматривать все высказанные в этой книге взгляды и мнения исключительно как мои собственные.

Замечание об определениях

Книги о гейм-дизайне принято начинать с определения слова «игра», но я постараюсь не поддаваться такому искушению.

За долгое время было выдвинуто немало определений. Среди самых распространенных – предложение рассматривать игры как «конфликты, в которых игроки непосредственно взаимодействуют между собой таким образом, чтобы помешать осуществлению целей других игроков» (Crawford, 2003, 8), или как «систему, в которой игроки вовлечены в искусственный конфликт, определяемый правилами с количественно определяемым исходом» (Salen and Zimmerman, 2004). Помимо этих, существует множество других

определений, замечательные описания которых приведены в обзорах Селлерса (Sellers, 2017, 90–96) и Сэйлен и Циммермана (Salen and Zimmerman, 2004, 73–81).

Такого рода краткие определения, как правило, бывают неточными, поскольку им обычно не удается передать огромное разнообразие *вещей и явлений*, которые люди называют играми. Слово «игра» используется для описания невероятно обширной категории занятий и артефактов, включая аркадные игры («стрелялки» вроде *Arkanoid* или *Gauntlet*), «экшен-игры» (*Super Mario Bros.*, *Grand Theft Auto*), логические игры или головоломки (*Go*, *судоку*), сюжетные игры (*The Curse of Monkey Island*, *Gone Home*), симуляторы-песочницы (*SimCity*, *Minecraft*, *Dwarf Fortress*), реалистичные симуляторы (*Microsoft Flight Simulator*, *Forza Motorsport*), спортивные видеоигры (*Madden*, *NBA Live*), реальные спортивные игры (футбол, бейсбол), карточные игры (*покер*, *джин-рамми*), коллекционные карточные игры (*Hearthstone*, *Magic the Gathering*), салонные игры («Слова», «Изысканный труп»), азартные игры (*рулетка*, различные слот-машины), стратегические настольные игры (*шахматы*, «Дипломатия», «Писк»), стратегические компьютерные игры (*Civilization*, *Master of Orion*), настольные игры для детей («Змеи и лестницы», *Candy Land*), уличные игры (варианты *прятки*, «казаков-разбойников»), игры с дополненной реальностью (*Majestic*, *Pokémon Go*), ролевые игры живого действия (*LARPs*), ролевые игры на основе правил (*Dungeons & Dragons*, *Ultima*), игры с открытым миром (*The Witcher*, *The Elder Scrolls V: Skyrim*), интерактивную литературу (квесты, текстовые квесты *Inform*), гипертекстовые повествования (игры Twine, серия «Выбери себе приключение», *Choose Your Own Adventure*) и прочие, которые я не буду перечислять, потому что суть и так ясна.

Проблема простых определений заключается в том, что им, как правило, не удается передать все это разнообразие и в итоге, как ни парадоксально, многие известные игры оказываются исключенными из понятия «игра». Например, согласно строгим определениям, которые дают Кроуфорд и Сэйлен с Циммерманом, чрезвычайно популярные игры из серии *SimCity* или *Minecraft* собственно «играми» не оказываются. Такой вывод противоречит живому использованию термина «игра», и, следовательно, эти определения не точны и не годятся для использования.

Во-вторых, что представляет еще большую проблему, неточные определения могут становиться нормативными и тем самым отказывать в онтологическом статусе «игра» новым типам игр. В конце 1990-х велись оживленные дискуссии по поводу того, что игры вроде *SimCity* или *The Sims* – это всего лишь «игрушки» (*toys*), потому что у них нет четко обозначенных целей и условий победы. Точно так же в начале 2000-х годов маргинальными объявлялись казуальные игры, поскольку они очень легкие и им не хватает более навыкоориентированного физического взаимодействия или стратегической сложности. В конце 2000-х ситуация повторилась с играми для социальных сетей (такими как *FarmVille*) и для мобильных устройств (такими как *Candy Crush*) – их обвиняли в том, что они чрезвычайно просты, но при этом сильно затягивают, а также позволяют игрокам пропускать нежелательные задания. В 2010-х годах предметом дискуссий стали «симуляторы-бродилки» вроде *Gone Home* и гипертекстовые игры вроде *Depression Quest*, поскольку их дизайнеры отказывались от традиционных механик и систем в пользу повествования и эмоциональных переживаний. Похоже, что феномен маргинализации различных типов игр и отрицание важности их вклада в силу неточности определения понятия «игры» повторяется с досадной регулярностью.

Похоже, что точное определение – это нечто зыбкое и ускользающее. Поэтому в данной книге мы придерживаемся не прескриптивного (предписывающего), а дескриптивного (описательного) подхода: игры – это то, во что играют люди, которые играют в игры, говорят об играх, делают игры и наслаждаются играми как таковыми. У этих «игр» могут быть некоторые общие черты. Обычно они не имеют последствий для реальной жизни, могут подразумевать элемент воображения, быть соревновательными, иметь цели или наблюдаемый исход (результат) и тому подобное – но все эти черты не обязательны. Такие «игроподобные» черты могут присутствовать, но не всегда и не все, и если рассматриваемое

явление *достаточно похоже* на другие явления, называемые «играми», то можно благополучно называть его «игрой». Таким образом, мы придерживаемся языкового подхода Витгенштейна (1959, параграфы 66–71) и воспринимаем термин «игра» как описание семейства напоминающих друг друга феноменов. Не представляется возможным составить их описание с набором необходимых и достаточных условий, но можно выделить феномен, разделяющий многие свои характеристики с другими объектами и явлениями, которые мы называем играми, и включить его в семейство.

Теперь, покончив с преамбулой, можно наконец-то перейти к главному вопросу, которому, собственно, и посвящен данный текст – как же делать такие вещи?

1 Элементы

«На какой-то промежуток времени мы покидаем себя, а когда возвращаемся, а нечто расширяется и укрепляется, ощущаем себя изменившимися как интеллектуально, так и эмоционально. А иногда... Иногда мы испытываем то, чего никогда не позволила бы нам испытать жизнь. Это неоценимый дар».

Марианна Вулф, «Что погружение в книгу делает с вашим мозгом?»

Всем нам знакомо чувство глубокого и приятного погружения в книгу, кинофильм, произведение искусства. Мы сознательно стремимся к тому, чтобы поддаться их очарованию, посмотреть на мир с иной точки зрения. Различные виды искусства достигают этого различными средствами, но все они разделяют эту загадочную силу.

Как хорошо известно игрокам, такая сила есть и у игр. Погружаясь в них, мы испытываем знакомое чувство трансформации – на какой-то промежуток времени покидаем себя и становимся кем-то другим, переживаем иную жизнь, переносимся в новый мир, становимся участниками какой-то истории.

Игры обладают и дополнительной силой. В них мы получаем возможность *действовать*. Можем исполнять различные роли. Становимся искателями приключений, первооткрывателями, полководцами – не только благодаря переживаниям, но и с помощью своих *активных* действий. Мы ощущаем, каково это – быть кем-то иным, действовать в другом мире. Непосредственно узнаем о последствиях различных поступков, учась на своем опыте тому, как устроен мир. Такова уникальная сила игр – они позволяют нам не только наблюдать за миром, но и жить в нем, действовать и даже изменять его.

Наша задача как гейм-дизайнеров (а заодно и тема данной книги) звучит так: как создавать такие миры, которые игроки будут населять и с которыми будут взаимодействовать?

Процесс дизайна

Создавая новую игру, мы уже можем иметь представление о некоторых ее основных идеях, о том, какого типа эта игра, что в ней будет делать игрок или как она будет выглядеть. Но нужно превратить эти идеи в конкретные детали новой игры, разработав дизайн с самого начала. Подобно тому как писатели готовятся к написанию книги, делая наброски сюжета или персонажей, так и гейм-дизайнеры пользуются различными методами для подготовки к созданию игры: планируют механики и циклы обратной связи, анализируют действия игрока в различных обстоятельствах, сопоставляют мотивацию игрока с восприятием игры и тому подобное. Но главное – мы создаем прототипы и экспериментируем.

В данном тексте речь пойдет о разнообразных инструментах и процессах, к которым прибегают гейм-дизайнеры. Ни одна игра не появляется на свет полностью сформировавшейся – они требуют определенной творческой искры, но помимо этого

необходимы технические средства и умение воплощать задуманное. В этой книге мы сосредоточимся на последней части, а именно на практических методах, которыми пользуются гейм-дизайнеры – таких как ментальные модели, описания процессов и способы мышления, оказывающиеся особенно полезными в процессе создания новых игр.

Игры как механизмы

Игры можно рассматривать с разных сторон: с точки зрения сюжета или сценария, с точки зрения художественного оформления и визуального дизайна, с точки зрения культурного анализа и т. д. Но поскольку в настоящий момент нас больше всего интересует создание геймплея и взаимодействие игрока с игрой, то мы будем придерживаться иного подхода.

Сосредоточимся на играх как на системах, с которыми взаимодействует игрок – как на *механизмах, с которыми играют*. В данном случае слово «механизм» – это сокращенное обозначение динамической модели, искусственной системы правил для взаимодействия, а не какой-то физической конструкции в буквальном смысле. Необходимо подчеркнуть интерактивную и динамическую природу игры: это механизм со своими собственными правилами; проверяя различные возможности, игрок взаимодействует с этим механизмом, а механизм реагирует на воздействие и, в свою очередь, вынуждает игрока действовать дальше. Такое чередование воздействия и реакции в рамках правил игры и образует то, что игрок воспринимает как «геймплей».

Чтобы понять, как эти механизмы работают, мы разберем их на части, а в процессе неизбежно начнем различать некоторые повторяющиеся схемы и шаблоны. Среди широкого разнообразия игр наблюдаются некие общие черты, распространенные структурные элементы и схожие дизайнерские решения. В каждой оригинальной игре они могут проявляться и сочетаться по-своему. Программа составлена из стандартных строительных блоков, и хотя со временем появляются новые элементы, многие из них существуют достаточно давно и хорошо известны. Будучи собранной, такая структура позволяет игрокам взаимодействовать с собой специфическим образом и порождает особый вид геймплея.

Такой способ анализа и понимания игр не единственный, но он весьма полезен. Практикующие гейм-дизайнеры уже выявили и определили многие регулярно повторяющиеся в практике гейм-дизайна общие структуры и строительные блоки.

Ориентированный на игрока гейм-дизайн

Рассуждая о своей профессии, гейм-дизайнеры придерживаются негласного предположения о том, что игры создают для того, *чтобы в них играли игроки*. Звучит как банальность, но не все рассматривают игры именно в таком ключе. Математики, например, могут рассматривать игры как задачи по поиску оптимального решения, игнорируя игроков.

Но мы будем придерживаться ключевого принципа, согласно которому игры существуют, чтобы в них играли. И поскольку их создают для игроков, то мы с самого начала должны учитывать игроков и их восприятие. Игрок должен быть отправной и центральной точкой в процессе дизайна.

Ориентированность на игрока – фундаментальный принцип гейм-дизайна. Игра разрабатывается как источник впечатлений (опыта), как механизм для взаимодействия, предоставляющий игроку агентивность (возможность активной деятельности) и автономию. Следуя этой идее, мы в процессе дизайна будем рассматривать три элемента: задаваемые дизайнером цели опыта; игровой артефакт, который будет реализовывать эти цели; и, что самое важное, игрока, переживающего этот опыт. Все они необходимы для диалога между дизайнером и игроком с помощью геймплея с задуманным дизайном.

Так с чего же нам начать?

Мотивирующий пример: покер

Воспользуемся популярной игрой в *покер* как практическим примером для изучения – рассмотрим, из каких элементов она состоит. Представим, что мы сидим в компании друзей и разыгрываем партию. На какие части делится этот процесс?

Во-первых, выделим основную структуру. Имеется некоторое количество игровых элементов, с которыми взаимодействуют участники: карты, фишки, игровой стол и другие составляющие, представляющие собой физический аспект игры. Можем считать их *объектами*, или *существительными*, игры. Карты и фишки – это конкретные объекты, но имеются и более абстрактные элементы: раздача карт («рука»), ход игрока и т. д. Они не обязательно должны быть физическими. Онлайн-покер состоит из тех же объектов (таких как стол), только виртуальных. Кроме того, у нас есть правила, что и в какие моменты делать с данными элементами. Правила определяют *действия*, или *глаголы*, сопоставляемые с указанными существительными: когда тасовать карты, как раздавать их, как делать начальную ставку, как повышать или уравнивать ее и т. д. Игра начинается с некоего начального *игрового состояния*, а затем благодаря действиям игроков переходит из одного состояния в другое. В случае с *покером* существуют правила, определяющие победу определенного расклада, имеющегося на руках, поэтому игроки могут попытаться достичь победного состояния с помощью выданных им карт.

Начиная игру, участники приводят все элементы в движение. Дилер тасует колоду, участники делают начальные ставки, им раздают карты. Затем игроки по очереди совершают свой ход, анализируя различные варианты: они могут сравнить ставку предыдущего игрока, поменять себе некоторые карты, повысить ставку, спасовать и т. д. В какой-то момент комбинации карт на руках игроков сравниваются между собой, после чего определяется победитель. Затем процесс начинается снова. Таков геймплей игры – анализ, решения и взаимодействия игроков. Но он не ограничивается просто игровыми объектами. Игроки рассчитывают шансы и реагируют на действия других участников. Они блефуют и пытаются заставить противников сбросить карты. Могут делать вид, что у них на руках совсем иные карты, чем в действительности. Другие игроки – очень важный элемент для многих игр, а покер особенно славится тем, что представляет собой смесь элементов стратегического мышления и планирования с психологической тактикой и попытками предсказать действия противников.

Вдобавок ко всему, можно поговорить о том, что делают и что *переживают* игроки во время игры. Разумеется, участники состязаются друг с другом, но также они сотрудничают – например, создавая и разрушая временные альянсы. Игроки рискуют своими деньгами и наслаждаются острыми ощущениями от осознания того, что сделали ставку на неопределенное будущее. Они рассчитывают свои дальнейшие шаги и получают удовольствие от планирования своих действий. Но в то же время участники беседуют и шутят, борются за статус победителя и приятно проводят время в компании. Это и есть *опыт игрока* для игры в *покер*. Приятное времяпрепровождение, социализация, азартные ставки, продумывание стратегий, попытки обмануть друг друга – достаточно притягательное для многих сочетание сильных ощущений, благодаря которому люди получают большое удовольствие от данного процесса.

Описание модели

В своем анализе игр мы будем пользоваться следующими тремя уровнями.

- Игроки взаимодействуют с различными игровыми *объектами*.
- Взаимодействие с игрой и с другими игроками в течение времени образует *геймплей*.
- Геймплей порождает у каждого игрока определенные *переживания* (*опыт*) и *ощущения*.

В случае с *покером* элементы – это карты, фишки и т. д., а игроки могут разговаривать друг с другом, что считается одним из приемлемых игровых действий. На основе правил участники рассаживаются и начинают раздавать карты, делать ставки, уравнивать или перебивать их, блефовать с целью сбить с толку соперников, а также иным образом взаимодействовать друг с другом и с меняющимся состоянием игры. Это, в свою очередь, позволяет игрокам получать удовольствие от состязания друг с другом, социализации, подсчета фишек на столе, разработки стратегий для выигрыша или просто от беседы и приятного времяпрепровождения с друзьями.

На иллюстрации 1.1 показано взаимодействие между основными объектами и действиями, производимый ими геймплей и возникающий в результате опыт игрока.

Трем данным уровням можно дать более общие названия (Иллюстрация 1.2).

- *Механики* – это игровые объекты и действия, которые совершают с ними игроки. Их можно собирать в *системы* с определенными свойствами.
- *Геймплей* – это процесс взаимодействия игроков с игровыми механиками.
- *Опыт игрока* – это субъективные ощущения участника игры от геймплея.

Такова наша базовая модель взаимодействия: игроки взаимодействуют с механикой и системами, порождающими геймплей, воспринимаемый игроками в виде субъективных ощущений. В следующих разделах мы более подробно опишем, как на основе этой модели можно понять роли игроков и дизайнеров в формировании такого взаимодействия.

Данная трехчастная модель была разработана на основе модели MDA (Hunicke, LeBlanc and Zubeck, 2004), в которой три уровня. Они называются «механика», «динамика» и «эстетика» (*mechanics, dynamics and aesthetics*), хотя соответствия между уровнями двух этих моделей не точные. Сходство и различия между ними объясняются в разделе «Дополнительная литература» в конце главы.



Иллюстрация 1.1. Пример покера

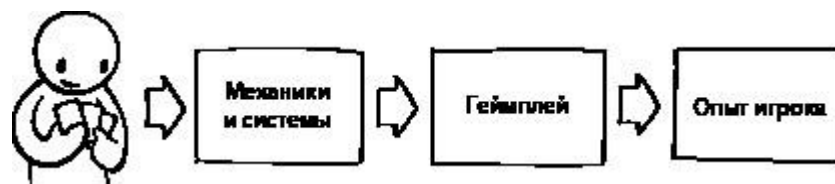


Иллюстрация 1.2. Базовая модель: механика и системы, геймплей, опыт игрока

Роль дизайнера

Представим теперь себя дизайнером, желающим создать новую игру. Обычно у специалиста уже есть какое-то представление о желаниях игрока – например, он хочет, чтобы тот решал какие-нибудь стратегические задачи, получал удовольствие от повествования с новым сюжетом или реализовывал свои фантазии о том, чтобы стать иным человеком из другой эпохи. Возможности почти безграничны, но как воплотить такой опыт на практике?

Трудность состоит в том, что мы не в состоянии внушить игроку эти желания напрямую. Можем только манипулировать конкретными базовыми элементами (правилами игры, объектами, персонажами и т. д.), из которых состоит игра. Нам необходимо создать

игровой артефакт , который при использовании поможет участнику испытать задуманный нами опыт.

Это трудная проблема второго порядка, вдвое увеличивающая для нас расстояние до конечной цели. Мы создаем не просто статический объект, а динамический механизм, который будет вести себя определенным образом, поэтому нам остается надеяться, что игрок получит удовольствие от такого поведения.

Процесс дизайна

Если рассмотреть, как разрабатывается дизайн чего-то другого – например, не игр, а программного обеспечения или физических объектов, то можно выделить два направления процесса.

- *Дизайн сверху вниз* : мы начинаем с общих концепции и целей, затем делим их на более мелкие части, подробно описываем, как они работают, затем делим части на еще более мелкие фрагменты и т. д.

- *Дизайн снизу вверх* : мы начинаем с мельчайших, насколько это возможно, элементов, и движемся по направлению к нашей общей концепции. Тестируем созданное, удостоверяемся в том, что оно соответствует нашим целям, а затем из мелких фрагментов постепенно выстраиваем всю систему.

В гейм-дизайне эти два подхода выглядят следующим образом.

- При движении сверху вниз дизайнер начинает с задуманного опыта игрока и решает, как разделить его на несколько частей. Мы продумываем, какого рода геймплей может породить такой опыт, а затем размышляем, какие из известных нам механик помогут создать данный геймплей.

- При движении снизу вверх сначала разрабатываются механики и системы, которые испытываются на реальных игроках; регулярное тестирование геймплея помогает понять, какой игровой опыт он создает.

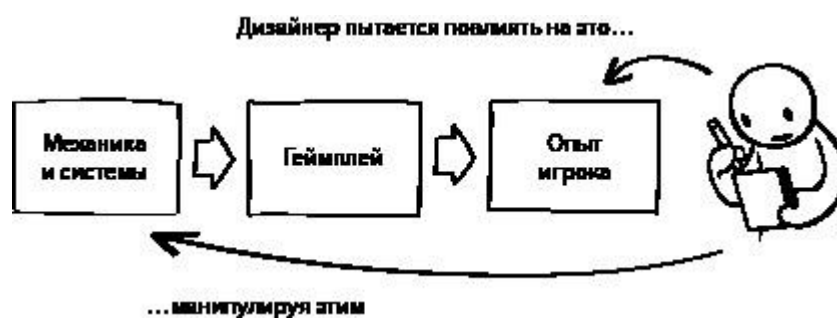


Иллюстрация 1.3. Роль дизайнера: разработка механики и систем для создания задуманного опыта

На практике эти два подхода не используются отдельно друг от друга. Чистый дизайн «сверху вниз» реализовать тяжело, потому что при попытке создания дизайна динамических систем трудно предсказать, как они будут работать, когда их предложат игрокам. Чистый же дизайн «снизу вверх» носит исключительно экспериментальный характер, но если при этом не руководствоваться какой-либо концепцией и представлением об опыте игрока, то все эти эксперименты никогда не приведут к созданию чего-либо цельного.



Иллюстрация 1.4. Два подхода к дизайну: сверху вниз и снизу вверх

Решение заключается в том, чтобы прибегнуть к гибриднему, итеративному подходу, работать с двух концов, разрабатывать планы сверху вниз, экспериментировать снизу вверх и, главное, создавать прототипы с самых ранних этапов, регулярно тестируя дизайнерские идеи и стараясь создать на их основе нечто цельное и связное. Такой гибридный, итеративный процесс делает гейм-дизайн похожим на некоторые другие его виды (Brooks, 2010).

Опыт игрока

Игрок рассматривает игру с иной перспективой, нежели дизайнер. Он не знает, какой опыт мы задумали для него, поэтому начинает с конкретных элементов игры – механик, игровых объектов и правил. Игрок запускает игру и начинает играть.

Получаемый в результате опыт бывает самым разным. Игроки могут восхититься увлекательными стратегическими заданиями или приятно провести время в исследовании виртуального мира, а могут испытать и неприятные эмоции, разочарование или скуку. Но все эти ощущения порождаются в результате *прохождения игры* – взаимодействия с различными игровыми объектами, правилами, возможно, другими участниками – по усмотрению игрока.

Как уже стало ясно к этому моменту, опыт игрока – это не то, что дизайнер задумал, а то, что он *реализовал*. Игроку известно только то, что ему предоставлено – объекты, правила, враги или задания. Намерения дизайнера нематериальны, за исключением тех случаев, когда они выразились в реализации.

Также наблюдается некоторое противостояние между намерениями дизайнера и действиями игрока. С одной стороны, игрок играет в мире, созданном дизайнером, взаимодействуя с механиками и системами, предоставленными ему специалистом и задуманными определенным образом. С другой стороны, игрок обладает свободой действий, у него свои собственные представления о том, что он хочет делать, не обязательно соответствующие замыслу дизайнера. В итоге только игроку решать, как поступать, в рамках правил игры он обладает полной свободой действовать так, как ему покажется правильным.

Геймплей является итогом взаимодействия между игроком и дизайнером. В него вносят свой вклад как специалист, так и игрок. Дизайнер задает рамки опыта, а игрок действует так, как ему захочется – согласно ожиданиям специалиста или вопреки им. Иногда такое взаимодействие оказывается успешным, и дизайнеру удастся реализовать задуманный опыт. В других же случаях игроки находят свои способы расшатать конструкцию дизайнера и превратить ее в нечто другое, соответствующее их пожеланиям. А порой тщательно продуманная конструкция просто рушится, а весь опыт разваливается на куски. Такое взаимодействие – диалог, требующий определенной степени сотрудничества с обеих сторон, равно как и понимания того, что не существует единственного «правильного» способа играть в игру.

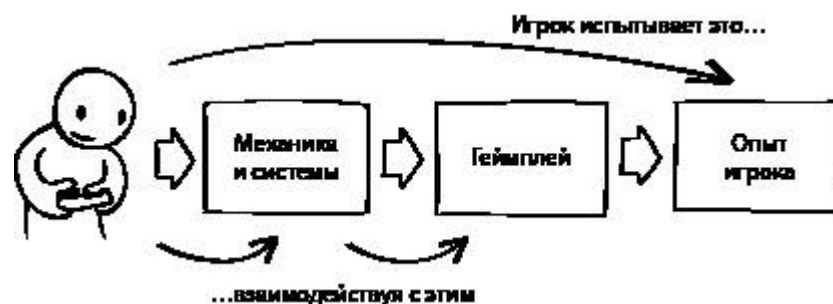


Иллюстрация 1.5. Игровой опыт складывается из взаимодействия игрока с игрой

Элементы игры вне этой модели

До сих пор мы описывали такие элементы, как механика, системы, геймплей и опыт игрока, лежащие в основе гейм-дизайна. В последующих главах мы будем освещать их более подробно.

Но, как знает любой геймер, удовольствие от игры обеспечивают не только эти элементы. Разумеется, они крайне важны, но существуют и другие вещи, также воздействующие на опыт игрока.

Огромное влияние на восприятие игры оказывает *визуальный дизайн*. Под ним подразумевается целый спектр различных элементов – от дизайна окружающей среды (как мир игры преподносится игроку), внешнего вида персонажей, окружения и типичных объектов до дизайна таких мелких особенностей как шрифт или детализация трехмерных моделей.

Также напрямую влияют на восприятие игрока качество *пользовательского интерфейса* и его общая подача. В компьютерных играх обычно говорят о «проектировании пользовательского опыта», хотя в физических играх к пользовательскому интерфейсу можно причислить и тактильное ощущение от игровых объектов или восприятие игрового поля.

С визуальным восприятием тесно связан выбор «*сеттинга*», или контекста игры, – мира, в котором разворачивается процесс, роли игрока и т. д. То, как игрок воспринимает свою роль в игровом мире, естественно, влияет на его ощущения от всего процесса.

В играх, основанных на повествовании, очень важную роль играет *сюжет*. На восприятие игрока влияют такие сценарные элементы, как персонажи и их мотивация, ситуации, в которых они оказываются, развитие сюжета, а также более базовые аспекты – например, качество диалогов.

Огромную роль в видеоиграх играет *музыкальный и звуковой дизайн*, музыкальное оформление или саундтреки. В некоторых жанрах музыка и звуковой дизайн занимают, несомненно, центральное положение.

Элементы *технического дизайна* обычно скрыты от игрока, но они оказывают очень большое воздействие на качество игрового опыта – например, уровень бросающего вызов игроку ИИ (искусственного интеллекта) или типы доступных для платформы многопользовательских матчей.

Попытка составления карты разнообразия игровых элементов, в своей совокупности влияющих на впечатления и опыт игрока, отражена на иллюстрации 1.6.

В данном тексте мы сосредоточимся только на дизайне механик и геймплея и на том, как изменения в геймплее воздействуют на игрока при том допущении, что «остальное тоже в равной степени важно». Другими словами, мы в данном случае сосредоточимся на *аспектах игрового опыта, порождаемых геймплеем*. Но общее впечатление от игры и *общий опыт* игрока от любой конкретной игры – явление многогранное, для которого геймплей не единственный фактор, часто даже не главный.



Иллюстрация 1.6. Примеры элементов вне геймплея, влияющих на опыт игрока

Практика гейм-дизайна

До сих пор мы использовали термин «гейм-дизайнер» в широком смысле – для обозначения человека, создающего игровые элементы, механику и системы, порождающие геймплей и влияющие на опыт игрока. Рассмотрим же теперь более точные определения термина «гейм-дизайнер» в практике гейм-индустрии.

Гейм-дизайн, системный дизайн, дизайн контента

На практике термин «гейм-дизайнер», как правило, описывает человека, чья непосредственная обязанность – разрабатывать правила игры, а затем реализовывать их.

Такое понимание намеренно отделяет эту профессию от других, вроде художника-оформителя, сосредотачивающегося на визуальном дизайне, или программиста, заведующего архитектурой программного кода и его внедрением. И хотя опыт игрока зависит от вклада каждого члена команды, именно дизайнеры несут ответственность за геймплей, то есть за то, как ведет себя игра и как игрок с ней взаимодействует.

Далее профессия гейм-дизайнера разделяется на категории.

Системные дизайнеры сосредотачиваются на общих механиках и системах, например на следующих.

Правила игры – на том, что представляют собой «существительные» и «глаголы».

Дизайн боев – как происходят сражения в игре, какие в них участвуют юниты и какое используется оружие.

Дизайн экономики – правила о том, как производится обмен объектами и валютой.

Дизайнеры контента концентрируются на отдельных игровых объектах и действиях.

Дизайн уровней – конкретное окружение, в котором разворачивается игра.

Дизайн персонажей – что из себя представляют персонажи и что они делают.

Дизайн мира – обстановка, в которой прокладывает свой путь игрок, почему это интересно и т. д.

Различие между системным дизайном и дизайном контента порой бывает весьма размытым. Как правило, и тем и другим обычно занимаются одни и те же люди. В целом «гейм-дизайн» – это определение того, как будет разворачиваться игра, продумывание правил взаимодействий, но «системный дизайн» – это подтип, уделяющий больше внимания общим правилам, а «дизайн контента» сосредотачивается на более конкретных местах, персонажах или объектах, а также на правилах, связанных с этими конкретными элементами. Как мы увидим в главе 4 «Системы», при настройке игры и при проверке сбалансированности общих правил и отдельных элементов обязательно следует уделять внимание этим типам.

Что касается других типов дизайна, таких как дизайн пользовательского интерфейса

или визуальный, то в индустрии они не считаются «гейм-дизайном». Обычно занимающиеся ими люди называются «художниками», потому что их задача заключается в художественном оформлении, а не в разработке интерактивного взаимодействия (хотя дизайн пользовательского и графического интерфейсов – это более интерактивное и междисциплинарное направление). Точно так же и дизайном миров могут заниматься *писатели*, разрабатывающие общий сюжет игры, придумывающие истории и мотивации персонажей и т. д., но при этом они не разрабатывают правила игры, касающиеся поведения мира.

Разумеется, все эти роли может одновременно исполнять и один человек, поэтому в небольших командах ведущий дизайнер обычно занимается самыми разными делами, пересекающимися с другими направлениями. Дизайнер контента часто работает вместе с визуальным дизайнером и сочинителем сюжета, а системные дизайнеры тесно сотрудничают с программистами.

В этом тексте мы будем сосредотачиваться преимущественно на гейм-дизайне. Дизайн контента – очень широкая дисциплина, требующая специфических знаний с большим упором на стили и жанры. Более подробно с ним можно познакомиться в книгах, перечисленных в разделе «Дополнительная литература» в конце главы.

Междисциплинарное взаимодействие

Гейм-дизайн – это один из столпов разработки игр, другие, не менее важные – производство графики (визуальный дизайн) и программирование (включая дизайн архитектуры и разработку программного обеспечения). Компании разработчиков часто привлекают специалистов других направлений, таких как писатели или звуковые дизайнеры, а также профессионалов из других отраслей, менее связанных с гейм-дизайном, таких как управление бизнесом или маркетинг.

Разделение на арт, гейм-дизайн и разработку кода всегда отражается на структуре студии. Организационное построение проектов часто формируется вертикально в соответствии с этими дисциплинами. Гейм-дизайнеров имеет смысл объединять в одну группу под руководством гейм-директора, потому что так они смогут оценивать работу друг друга и делать полезные замечания.

В то же время эти роли часто возлагаются на других специалистов. Бывает так, что на практике, помимо вертикального распределения в рабочих коллективах, наблюдается горизонтальное разделение по небольшим междисциплинарным группам. Такие рабочие группы объединяют представителей всех трех дисциплин и занимаются разработкой какой-то определенной особенности игры от начала до конца. В данном тексте мы будем рассуждать преимущественно об элементах дизайна, но для гейм-дизайнера очень важно научиться сотрудничать с представителями других дисциплин – художниками, программистами, маркетологами, специалистами по бизнес-планированию и прочими.

Краткий итог

В этой главе мы представили базовую модель, основу для обсуждения в дальнейшем. Ключевые ее положения таковы.

- Далее будут рассматриваться игры как «механизмы» и как системы правил и принципов взаимодействия, которыми пользуются игроки и которые определяют их действия. Мы сосредоточимся на том, как работают эти механизмы, и на том, как их можно анализировать.

- В данном тексте рассматриваются игры на трех уровнях.
- *Механики* – отдельные элементы, из которых состоит игра.
- *Геймплей* – динамический процесс взаимодействия с игрой со стороны игрока.
- *Опыт игрока* – субъективные впечатления игрока от игры.

- Механики – наиболее доступный для анализа элемент. Игроки взаимодействуют с игрой посредством механик, порождающих геймплей и определяющих конкретное восприятие игрока. Гейм-дизайнеры могут пожелать, чтобы у игрока возникали какие-то конкретные впечатления, но не могут непосредственно вызывать их. Вместо этого приходится разрабатывать механики, определяющие специфический геймплей и влияющие на впечатления игрока, взаимодействующего с ними. Но как механики, так и игроки могут действовать непредсказуемым образом. Дизайнеры по мере возможностей должны учитывать это.

- На практике гейм-дизайн подразделяется на *системный дизайн* общих правил, механик и систем, и *дизайн контента*, то есть дизайн отдельных конкретных элементов, с которыми сталкивается игрок. Мы будем сосредотачиваться на первом типе.

- Наконец, данный текст сосредотачивается на геймплее, но следует помнить о том, что другие элементы разработки игр также оказывают огромное влияние на игрока, на его опыт и получаемое от игры удовольствие. Будем рассуждать об опыте участника с точки зрения того, как данный процесс вытекает из геймплея.

На основе трехуровневой модели гейм-дизайна главы этой книги рассматривают данные уровни и распределяются следующим образом.

- В главе 2 начинаем с верхнего уровня и рассматриваем разные виды игрового опыта, а также различные способы его анализа.

- Далее, в главах 3 и 4, переключаемся на нижний уровень и подробно рассматриваем механики, что они собой представляют, как работают и как сообща образуют системы.

- Затем, в главе 5 и 6, рассмотрим средний уровень – как механики и системы приводят к геймплею, как можно его анализировать, и как геймплей обуславливает опыт игрока. Эти главы описывают своего рода синтез задуманного дизайнером для игрока опыта и механик, которые реализуют его на практике.

- Наконец, в главе 7, переключаемся на весь процесс – как дизайнеры пользуются описанными аналитическими инструментами при разработке игр и при создании прототипов, благодаря которым изначальная идея превращается в рабочий дизайн.

Дополнительная литература

Формальные инструменты

Гейм-дизайнеры стали делиться своим опытом начиная с самых первых дней разработки коммерческих настольных и компьютерных игр – описывая свои методы, разбирая уже существующие игры или давая советы о том, что сработало, а что нет. Но по мере роста индустрии в 1990-х годах росла и потребность не просто в обмене опытом в связи с конкретными продуктами, а в обобщении и документировании всего накопленного гейм-дизайнерами опыта. Возможно, самая известная попытка такого обобщения была предпринята в статье «Формальные абстрактные инструменты дизайна» Дага Черча (*Formal Abstract Design Tools*, Doug Church, 1999), за которым последовал ответ Грега Костикиана в виде статьи «У меня нет слов, и я должен заниматься дизайном» (*I Have No Words and I Must Design*, Greg Costikyan, 2002). Обе статьи доступны онлайн и стоят того, чтобы прочитать их целиком.

MDA

С годами были разработаны различные общие модели декомпозиции и анализа игр. Одна из популярных – это фреймворк MDA (Hunicke, LeBlanc and Zubek, 2004), который и

лег в основу данной работы. Автор этого текста – один из соавторов работы, посвященной модели MDA.

Читатели, знакомые с концепцией MDA, вероятно заметили ее сходство с описываемой здесь моделью, ведь она так же описывает три аспекта дизайна. В модели MDA они называются «механика», «динамика» и «эстетика», благодаря чему модель и получила свое название (*mechanics, dynamics, aesthetics*). В данном тексте сохраняется деление на три части, как и упор на дизайнерскую проблему высшего порядка, когда игрокам и дизайнерам приходится взаимодействовать с игровым артефактом только посредством его механик.

Но в этой работе модель MDA не излагается по ряду причин. Главная – это разительное отличие терминологии современной практики дизайна от терминологии концепции MDA. Термин «динамика» на практике почти никогда не употребляется для описания геймплея, это затруднило бы читателям общение с настоящими дизайнерами. Точно так же и термин «эстетика» в концепции MDA означает «эстетическое впечатление от взаимодействия с игровыми системами», но в индустрии разработки игр он используется почти исключительно в рассуждениях о визуальной эстетике, а не о геймплейном опыте игрока. Стремление настоять на терминологии MDA стало бы источником путаницы.

Во-вторых, в модели MDA понятие «динамики» рассматривается слишком широко и включает в себя как анализ динамических систем, состоящих из механик, так и анализ взаимодействия игрока с игрой. И хотя оба эти аспекта имеют общие корни (они описывают поведение), в современной дизайнерской практике считается более удобным рассуждать об игровых системах как об отдельном классе феноменов, отличающихся от геймплейных циклов и от игрового опыта при взаимодействии с программой.

Наконец, описываемое в модели MDA расположение дизайнера и игрока на противоположных концах цепочки идеализировано и сбивает с толку, поскольку в реальном итеративном процессе дизайна к игре приближаются одновременно с обеих сторон. Надеюсь, что данная улучшенная модель поможет устранить некоторые из этих недостатков.

Практика дизайна

Что касается дизайна в целом, а не только игр, то крайне полезная книга, представляющая собой введение в дизайн и повествующая о том, как его выбор определяется психологией, – это «Дизайн повседневных вещей» Нормана (*The Design of Everyday Things*, Norman, 1988). «Дизайн Дизайна» Брукса (*The Design of Design*, Brooks, 2010) – также вполне доступное введение в общую теорию дизайна.

Среди литературы, посвященной гейм-дизайну, особенно дизайну контента и повседневному труду специалистов этой отрасли, включающей многочисленные интервью с профессионалами, выделяются две относительно недавно вышедшие работы: «Мастерская гейм-дизайна» Фуллертон (*Game Design Workshop*, Fullerton, 2008) и «Испытания для гейм-дизайнеров» Брэттуэйт и Шрайбера (*Challenges for Game Designers*, Brathwaite and Schreiber, 2009).

Помимо дизайна геймплея, гейм-дизайнеры должны быть знакомы с основами визуального дизайна, поскольку он обычно сильно влияет на впечатления от игры и ее прохождения. Прекрасное введение в визуальный дизайн игр и его влияние на геймплей – «Интерактивные истории и искусство видеоигр» Соларски (*Interactive Stories and Video Game Art*, Solarski, 2017a) с многочисленными практическими советами и подсказками.

Мое пособие посвящено практике дизайна игр как артефактов и не уделяет большого внимания внешним факторам игровой индустрии, играм как средству коммуникации, как повествованиям или их пересечению с более обширными культурными и социальными практиками. Среди литературы, посвященной данным аспектам, достойны упоминания «Правила игры» Сейлен и Циммермана (*Rules of Play*, Salen and Zimmerman, 2004), особенно третья и четвертая части, а также сборники эссе «Первое лицо» (*First Person*, Wardrip-Fruin and Harrigan, 2004), «Второе лицо» (*Second Person*, Wardrip-Fruin and Harrigan, 2007)

и «Третье лицо» (*Third Person*, Wardrip-Fruin and Harrigan, 2009).

Упражнения для индивидуального выполнения

1.1. Существительные и глаголы

Выберите известную вам многопользовательскую игру и разберите ее подобно тому, как в этой главе был разобран *покер*. Лучше выбрать не слишком сложную карточную или настольную игру уровня *покера*. Ответьте на следующие вопросы.

а. Каковы существительные и глаголы данной игры? Что в ней делают игроки и с помощью чего они совершают действия?

б. Что представляет собой ее геймплей? Что происходит, когда игроки начинают игру? В какие виды деятельности они оказываются вовлечены?

в. Как можно описать опыт игрока в данной игре? Доставляет ли он удовольствие? Как на впечатление от игры влияет геймплей?

1.2. Элементы вне этой модели

Снова проанализируйте игру, которую вы рассматривали в упражнении 1.1. На этот раз выделите элементы помимо механик и геймплея, влияющие на впечатление от нее. Составьте их список. Что это за элементы? Для каждого элемента в одном предложении опишите, как он, по вашему мнению, способствует получению удовольствия от игры.

2 Опыт игрока

У всех нас есть свои любимые игры. Каждый помнит те, которым удалось захватить наше внимание, в которые мы погружались и которые очаровывали нас; игры, в которые мы какое-то время играли «запоем» или к которым регулярно возвращаемся до сих пор. Работа гейм-дизайнера как раз и заключается в том, чтобы у игрока возникали такие ощущения и воспоминания. Мы хотим, чтобы участники заинтересовались игрой, чтобы она их увлекла, чтобы у игроков возникло вдохновение или желание решить какие-то задачи либо чтобы они просто получали удовольствие и другие ощущения от геймплея.

Но почему игрокам захочется поиграть в нашу игру? Что такого захватывающего в ней будет? Этот вопрос – ключевой в понимании роли геймдизайнера. И на него очень трудно ответить.

В предыдущей главе мы описали общую модель анализа гейм-дизайна и поговорили о том, что дизайнеры стараются сделать, чтобы игроки переживали задуманный ими опыт. С этого и начнем наш анализ. В данной главе исследуем опыт игрока, обсудим разные способы размышлений об игроках и проанализируем виды мотивации игроков во время игры.

Относительность опыта

Что такого притягательного мы находим в играх? Я могу ответить на данный вопрос от своего имени: почему мне нравится то, что нравится? Подумав о своих любимых играх, легко могу привести ряд причин. Например, мне нравятся игры из серии *Civilization*, но почему? Наверное, из-за стратегического вызова – ведь в них можно создать свою могущественную империю, которую придется защищать от внешней угрозы и от внутреннего экономического краха. Или взять для примера игру из серии *SimCity*, где мне особенно нравится процесс постройки раскинувшегося на обширном пространстве города, которым по мере его роста становится все труднее управлять, а чтобы построить что-то

новое, приходится разрушать старое.

Каждый может привести в пример свои любимые игры и перечислить множество причин, по которым они нравятся. В то же время если я начну спрашивать других людей, нравятся ли им игры *Civilization* или *SimCity*, то некоторые ответят, что нет. Или нравятся, но по другим причинам. Возможно, кто-то обожает сражения и вообще военный аспект *Civilization*, но, в отличие от меня, недолюбливает строительство и управление городами.

Выражаясь словами старой поговорки, «на вкус и цвет товарищей нет».

Несмотря на избитость и банальность, мы интуитивно ощущаем ее правоту. Ожидаем, что у каждого игрока есть свои предпочтения, свои симпатии и антипатии. Эта предпосылка приводит к выводам, которые послужат нам отправной точкой.

1. *Игроки играют в игры, потому что получают удовольствие от них*. Такое положение кажется прописной истиной. Участники находят в затягивающей их игре нечто интересное, соответствующее психологическому настрою, получают мощную мотивацию для игры, иначе это будет нудная обязанность, а не увлекательный процесс. Но каким образом она возникает? Можно сказать, что игроки получают «удовольствие» от игры, но воспринимать это слово в широком смысле довольно опасно. В повседневном понимании «удовольствие» – это ощущение от чего-либо приятного, забавного, веселого, но игры необязательно должны быть такими, чтобы доставлять игроку «удовольствие». Игрокам часто приходится решать трудные задачи или продолжать упорно двигаться вперед, несмотря на постоянные неудачи и разочарования. Им может нравиться игра, повествующая о таких серьезных и болезненных темах, как смерть или катастрофа. Мы не можем назвать в качестве причины «удовольствие» как таковое, но упомянем, что люди «получают удовольствие от игры», используя это выражение как сокращенное обозначение того, что игроки испытывают мотивацию для игры совершенно по различным причинам.

2. *Разные игроки получают удовольствие от различных аспектов игр*. Кому-то нравятся стратегические игры. Они пробуждают интеллектуальное любопытство и дарят мотивацию. Но стратегии нравятся далеко не всем – некоторые их откровенно недолюбливают. То же самое можно сказать о «стрелялках», «гонках» и т. д. – у каждого из нас свои представления об играх, которые ему нравятся или не нравятся. Говоря вкратце, нет такого жанра или стиля, который бы всем безоговорочно нравился и доставлял удовольствие. Точно так же мы никогда не найдем человека, которому нравились бы игры всех жанров и стилей без исключения.

3. *Игроки могут получать удовольствие от одних и тех же игр по разным причинам*. Мне в *Civilization* могут нравиться исторический аспект и экономический менеджмент, а другой игрок может предпочитать сражения и военный аспект. Не бывает никакого «правильного» способа получать удовольствие от игры. Но это не значит, что программа в равной степени ориентирована на все возможные направления. Если я решу поиграть в *Civilization*, ожидая глубокого сюжета и проработанных персонажей, то буду разочарован, потому что игра в очень большой степени ориентирована на стратегию и менеджмент. И хотя некоторые крупномасштабные игры стараются поддерживать различные аспекты геймплея, чтобы разные типы игроков находили в них что-то свое, вряд ли эти аспекты будут развиты в них в равной степени.

4. *Доставляет игра удовольствие или нет – это понятие относительное*. Непосредственно из предыдущего пункта следует вывод, что если восприятие у людей разное, то не имеет смысла спрашивать объективно, доставляет ли она «удовольствие» сама по себе. Одна и та же игра по разным причинам может нравиться одним игрокам и не нравиться другим, поэтому имеет смысл говорить о том, что она *доставляет удовольствие некоторым определенным игрокам*. Позже мы попытаемся обобщить этот вывод и задать вопрос, может ли игра доставлять удовольствие какому-то определенному типу игроков. Такое обобщение будет весьма полезным, но при этом оно всегда должно сопровождаться упоминанием о конкретной аудитории.

Эти пункты не должны показаться неожиданными и новыми для тех, кто знаком с практикой современного искусства, ведь не существует самой по себе «эстетики» как таковой без упоминания участников, их предпочтений и способов взаимодействия с произведениями искусства. Но в данной работе мы не станем затрагивать философию эстетики, а упомянем лишь, что она просто имеет некоторое отношение к рассматриваемому вопросу.

Перечисленные интуитивные выводы оказывают непосредственное влияние на гейм-дизайн. Мы не можем создать игру, которая объективно «понравилась» бы всем, доставляя объективное «удовольствие». Нельзя даже надеяться на то, чтобы создать программу, которая будет объективно «лучше» другой, без упоминания аудитории или каких-то критериев оценки. Нельзя ожидать, что игра понравится или не понравится всем, но при этом она будет нравиться по различным причинам, а разные игроки будут находить в ней разную мотивацию. То, что мы действительно можем сделать – это создать игру, которая будет каким-то образом нравиться каким-либо определенным игрокам. Мы можем и должны рассуждать о том, как эти определенные игроки будут воспринимать нашу игру и как мы можем направлять их к задуманному нами восприятию.

Как следствие, в процессе дизайна с самого начала нужно задумываться, какова наша аудитория и почему она может получать удовольствие от данной игры.

Что вам нравится?

Итак, мы понимаем, что удовольствие от игры – это понятие в высшей степени субъективное, зависящее как от игрока, так и от самой игры. Сделаем следующий шаг. Осознав, какая игра нравится, можно спросить себя о том, что же *конкретно* нравится в ней.

Взяв для примера игру *Civilization V*, представляющую собой довольно сложную стратегию, я могу перечислить целый ряд личных причин.

- Мне нравится, что нужно начинать с небольшого поселения и превращать его в сверхдержаву.
- Мне нравится строить большие города и управлять ими, следя за тем, чтобы жители в них были довольны.
- Мне не нравятся сражения, но нравится обладать большими армиями и использовать их для переговоров.
- Мне нравится быстро принимать решения в неожиданных ситуациях и рассчитывать следующие шаги.
- Мне очень нравится заниматься научными исследованиями и развивать новые технологии, позволяющие создавать более мощные юниты.
- Мне нравится визуальный стиль и то, что игра немного похожа на настольную.
- Мне нравится исторический контекст и нравится играть против реально существовавших или существующих стран.
- Мне кажется забавным играть за реального исторического деятеля вроде Наполеона или Александра Македонского.

И так далее...



Иллюстрация 2.1. Скриншот из игры *Civilization V*

Это мой персональный мотивационный список – в нем перечислено то, что мне нравится и почему. Каждый из нас может составить похожие списки того, что ему нравится в одних играх и не нравится в других.

Но польза от таких длинных и подробных списков невелика. Будет лучше, если мы попытаемся обобщить наши выводы, чтобы применить их для рассуждений о других играх и других игроках.

Построение простой и практической таксономии

Начну с простой *категоризации* указанных выше пунктов. Выявить глубинную структуру предпочтений нам поможет выделение таксономических групп.

Я распределю свои причины для игры в *Civilization* по категориям, а затем попытаюсь понять, что такого особо привлекательного наблюдается в каждой из них. Наивная (то есть простая и не претендующая на особую систематичность) группировка может выглядеть примерно таким образом.

- *Элементы геймплея* : то, что я делаю в игре, с какими существительными и глаголами мне приходится иметь дело, основанная на них система, предлагаемые ими задачи, особенно касающиеся стратегии и экономического менеджмента. Если поменять оформление игры, но сохранить геймплейные элементы, от них можно будет получать то же самое удовольствие.

- *Фантазийные элементы* : исторический сеттинг, персонажи игры, уровни, юниты, оружие и прочие элементы. Наша цивилизация и ее история – сами по себе интересные темы и без правил и механик игры.

- *Элементы оформления* : визуальный дизайн, внешний вид персонажей и мест, их анимация и стили для разных регионов и исторических эпох.

- *Повествовательные элементы* : персонажи, их истории, действия, поступки и мотивация могут показаться интересными подобно тому, как нас увлекают фильмы и книги, затягивая зрителей и читателей в вымышленные миры.

На основе такой простой и практичной таксономии можно достаточно хорошо описать, насколько *Civilization* удовлетворяет моим предпочтениям.

- Мне очень нравятся *элементы геймплея* этой игры, их много, они разнообразны.
- Мне нравится *фантазировать* – воображать себя историческим правителем, а игра очень хорошо это поддерживает.

- Я нейтрально отношусь к *оформлению* , хотя игра оформлена очень хорошо.
- Мне нравятся игры с *повествовательными элементами* , но в данной игре их не много.

Вот таким образом мы создали нашу первую теорию *мотивации игрока*, проанализировав мотивацию отдельного представителя и перегруппировав его мотивы так, чтобы сделать какие-то теоретические обобщения. На основе этой таксономии можно предсказывать и тестировать их. Например, если я встречу другие игры с сильными элементами геймплея и фантазийными элементами, то на основании данной теории могу предсказать, что они мне тоже понравятся.

Но не будем заблуждаться – эта наивная таксономия *ужасна* . Она страдает от многочисленных дефектов, и худший из них – то, что все основано на моих интуитивных предположениях и на моем собственном опыте взаимодействия всего лишь с одной игрой (без учета других игр или опыта других игроков). Можно ли исходя из данной теории предсказать, насколько *Civilization* понравится другим? Вряд ли. Поэтому она не будет полезной в рассуждениях о том, как делать игры для других игроков.

Тем не менее это неплохая отправная точка для рассмотрения иных, более удачных теорий опыта игрока.

Теории об игроках

Итак, мы увидели, что индивидуального опыта недостаточно. Мой собственный мотивационный профиль объясняет мое личное восприятие, но не объясняет восприятие других. Предпочтительной будет мотивационная теория, помогающая понять, нравятся ли другим людям какие-то специфические игры, почему они им нравятся и какого рода удовольствие игроки получают от этих игр.

Было предпринято немало попыток построения моделей, подразделяющих игроков на различные типы, дающие возможность понять связи между типами и мотивами. Мы будем рассуждать о трех разных подходах: дизайнерских теориях, моделях «психологических портретов игроков» и об эмпирических исследованиях. В данном разделе разберем первые два, а третий подход рассмотрим далее.

Дизайнерские теории

Из наблюдений за игроками становится очевидным, что разные игроки получают различного рода удовольствие от игры или отдают предпочтение одним типам игр над другими. Гейм-дизайнеры постоянно тестируют свои программы с помощью разных игроков и в различных контекстах, а некоторые делятся своими наблюдениями относительно поведения игроков.

В этой главе будем называть эти наблюдения *дизайнерскими теориями* о поведении игроков – созданными на основе наблюдений дизайнеров и их интуитивных выводов о различных типах мотивации игроков. Ниже рассмотрим две, пожалуй, самые известные теории: модель типов игроков Бартла и модель мотивации индивидуального игрока Костера.

Модель Бартла

Вероятно, самая известная дизайнерская таксономия – это модель Бартла (1996). Наблюдая за многопользовательскими текстовыми онлайн-играми (так называемыми MUD), Бартл заметил, что при взаимодействии между собой участники таких игр, как правило, придерживаются определенных шаблонов поведения. На основании этого он заявил, что игроки MUD в зависимости от своей мотивации делятся на четыре общие категории.

- «Карьеристы». В данном случае основная мотивация – повышать уровни и достигать целей.
- «Исследователи». У них мотивация – исследовать игровой мир и узнавать, как он работает.
- «Социальщики». Здесь мотивация – повествование, общение с другими игроками, ролевая игра.
- «Киллеры». Их мотивация – состязание, победа, доминирование и даже «троллинг».

Модель Бартла предполагает, что эти разные типы игроков получают удовольствие от игры в неравной степени из разных источников. Если бы я был «карьеристом», то мне больше нравились бы аспекты игры, посвященные рангам и целям, тогда как «социальщику» это нравится в меньшей степени, он получает больше удовольствия от компании других игроков.



Иллюстрация 2.2. Диаграмма Бартла четырех типов игроков (1996)

Данная модель также предполагает различное поведение групп, что потенциально вовлекает их в конфликт. «Карьеристам», например, бывает неприятно объединяться с «социальщиками», потому что те «не играют ради победы». Поэтому в контексте многопользовательских игр очень важно создавать такие игровые миры, в которых каждый найдет что-то свое и где будет соблюдаться баланс.

Несмотря на то что изначально таксономия Бартла была создана для описания игроков в MUD, она сохраняет свою популярность, поскольку служит простым и практичным средством определения мотивации разных типов игроков и способов улучшения игры для них. Тем не менее, поскольку эти категории основаны на интуитивных догадках и неформальных наблюдениях, каждому гейм-дизайнеру следует заменять их наблюдениями за *своей собственной базой игроков*, взаимодействующих с его собственной игрой, потому что у игроков какой-то конкретной игры могут быть иные мотивы, отличающиеся от четырех описанных выше.

Модель Костера

В книге Рэфа Костера «Теория развлечений» (*A Theory of Fun*, 2004) приводится модель мотивации игроков, исходящей из фундаментального стремления получать удовольствие от обучения и тем самым становиться лучше (улучшать свои навыки). Пока мы сталкиваемся с новыми задачами, преодолеваем трудности и учимся чему-то новому, игра нас увлекает и доставляет удовольствие. Но для этого задачи должны быть как действительно интересными для нас (к счастью, люди – это любознательные существа, много что находящие интересным), так и «правильной степени сложности» (быть не слишком легкими, но и не очень тяжелыми).

Удовольствие от обучения связано с психологической *теорией потока* и персональной мотивации, описывающей многие области человеческой деятельности, а не только игры (Csikszentmihalyi, Abuhamdeh and Nakamura, 2005). Она утверждает, что все люди могут испытывать приятное состояние глубокого погружения (так называемое

состояние потока, по которому и названа теория), когда занимаются чем-то интересным с подходящей степенью сложности. Поскольку такого рода удовольствие связано с интересами игрока, его мастерством и сложностью игры, теория довольно удачно объясняет, почему одна и та же программа воспринимается различными игроками очень по-разному. Например, такая сложная стратегия как *Civilization* может казаться очень привлекательной для опытных игроков, но неинтересной для неопытных, не очень знакомых с этим жанром и его условностями, даже если обе группы в равной степени интересуются играми, требующими быстрых рефлексов или стратегического мышления и планирования и т. д. (Более подробно теорию потока мы затронем в главе 5 «Геймплей».)

Такое понимание «удовольствия» связано с понятиями «пространства действий» и «пространства состояний», о которых мы поговорим в главе 3 «Механика». Оно напрямую зависит от того, насколько хорошо игрок понял, каким образом его действия влияют на состояние игры. Процесс не доставляет наслаждения, если он слишком предсказуем или наоборот, а наибольшее удовольствие игрок получает именно тогда, когда благодаря своим действиям активно обучается влиять на состояние игры.

Существуют и другие дизайнерские таксономии, которые мы затронем в разделе «Дополнительная литература» в конце главы.

Психологический портрет игрока

Если дизайнерские теории стараются объяснить мотивацию игроков в абстрактном плане, то другой подход предлагает более персонализированные возможные объяснения мотивации каких-либо отдельных игроков в контексте их образа жизни и предпочтений.

Такой подход используется в методе создания «психологических портретов игроков» (далее «портрет игрока»). Это популярное средство дизайна и маркетинга всех типов товаров и услуг; обычно в данном случае составляются «портреты» представителей разных типов людей, которые могли бы заинтересоваться продуктом. Такие портреты формируются либо на основе исследований и опросов среди игроков существующих продуктов и выявления их общих черт, либо просто исходя из сделанных на основе опыта *предположений* о том, кто мог бы стать наиболее типичным потребителем данного продукта.

В «портрете игрока» мы постараемся обобщить черты различных людей, представить их в образе конкретного человека, ведущего определенный образ жизни, и предположить, как и почему этот человек мог бы захотеть сыграть в данную игру. При работе над программой вроде *The Sims* можно представить себе несколько разных типов игроков в образах трех-четырёх стереотипных личностей, например таких.

Элис – девочка-подросток, играющая в *The Sims* несколько часов после школы. Она любит создавать дома и придумывать истории разных живущих по соседству семей. Также ей очень нравятся различные тематические расширения.

Боб – молодой профессионал двадцати с лишним лет, регулярно играющий в пошаговые и экономические стратегии, переключаясь между ними. В играх его привлекает элемент конкуренции, он находит новые способы улучшить свои предыдущие результаты и достижения.

Кэрол – профессионал и мать двоих детей школьного возраста, которой нравятся игры, но она может уделять им время лишь изредка, возможно, около часа в день. Ей нравится придумывать интерьеры, она часто переделывает свои дома и создает новые игровые объекты.

Такие описания могут показаться слишком неуклюжими и ужасно стереотипными, и они действительно часто бывают таковыми. Почему, например, мужчина обязательно любит состязание, а женщина, причем мать двоих детей, создавать интерьер? Отражение ли это реальных типов игроков или всего лишь частный взгляд на мир составителя описаний?

Но к шаблонности прибегают намеренно. Суть в том, чтобы, исходя из сопоставления между собой различных стереотипных образов (возможно, неверных, но интересных),

представить себе мотивацию и поведение игроков. Портрет игрока – это «насос интуиции», история, помогающая дизайнеру понять воображаемого игрока и ответить на вопросы вроде: «Для кого мы делаем этот продукт?» и «Почему данный человек захочет приобрести нашу игру?» Портрет игрока олицетворяет собой наши интуитивные догадки и расхожие представления о людях, помогая понять их мотивацию.

Этот метод очень полезен тем, что помогает дизайнерам определить общие задачи и направление игры. Например, если мы приходим к выводу, что типичные наши игроки – взрослые люди, играющие в игры после работы, то постараемся сделать ее полегче и разбить на короткие сессии, с необязательной многопользовательской составляющей, продумаем онлайн-соревнования с учетом графика рабочих дней и т. д.

Метод портретов игроков отличается от теоретических моделей тем, что не претендует на универсальность. Портреты описываются с нуля для каждого конкретного продукта и для каждой конкретной аудитории, но, помимо этого, не используются для построения каких-то общих выводов.

Эмпирические модели

Дизайнерские теории, вроде упомянутых выше, как правило, основаны на интуиции и опыте их авторов, имевших дело с большим количеством игроков, это «выжимки» их непосредственных наблюдений. Но можно подойти к вопросу и с иной точки зрения, удалив из уравнения дизайнера. Например, можно попросить самих игроков рассказать о том, что они чувствуют в различных ситуациях и при разных видах деятельности (в играх или за их пределами), а потом сопоставить между собой ответы и выделить на этой основе группы.

Опишем здесь два интересных и относящихся к нашей теме эмпирических исследования, оказавших непосредственное влияние на разработку игр: во-первых, это популярная модель описания черт личности «Большая пятерка» (независимо от игр); во-вторых, некоторые недавние исследования типов игроков именно компьютерных игр. После этого постараемся описать, как эти исследования соотносятся между собой.

Модель личности «Большая пятерка»

Все мы отличаемся друг от друга и каждый обладает своими личными особенностями и чертами характера, побуждающими нас в разных ситуациях действовать по-своему. Это следует из простого здравого смысла. Но представив себя в роли дизайнера, можно задуматься – а существует ли способ как-то классифицировать эти персональные отличия, понять, почему разные люди действуют так или иначе, или предсказать, как они будут действовать в той или иной ситуации?

В истории было выдвинуто много моделей личности, начиная с античных теорий о четырех основных жидкостях организма до относительно недавних спекуляций в духе типологии Майерс – Бриггс, но всем им присущ один общий дефект: они пытаются втиснуть людей в рамки заранее определенной таксономии. Современная же психология, напротив, предпочитает такие модели личности, в которых категории определяются эмпирически, а не задаются априори.

Возможно, самая известная и хорошо протестированная среди этих ориентированных на данные моделей – «Большая пятерка». Она основана на анализе того, как люди описывают самих себя и других, а также на том, какими словами они пользуются при описании распространенных реакций и эмоциональных состояний. В модели «Большая пятерка» выделяются следующие пять основных факторов или измерений личности (John and Srivastava, 1999, 121).

- *Экстраверсия* – ориентированность на социальное окружение, общительность, а также настойчивость, напористость.

- *Доброжелательность* – ориентированность на интересы общества, альтруизм, доверие, скромность.
- *Сознательность* – ориентированность на цель, умение контролировать импульсивное поведение, склонность следовать установленным нормам, организованность.
- *Нейротизм* – отрицательная эмоциональность, беспокойство, нервозность, напряженность.
- *Открытость* – стремление к расширению опыта, оригинальность, сложность.

У каждого из этих измерений есть своя обратная сторона. Например, противоположность экстраверсии – интроверсия (необщительность, неприязнь к социальному окружению). Черты личности каждого человека можно описать как занимающие некое положение где-то между положительным и отрицательным полюсами каждого измерения. Они обычно сохраняются на протяжении взрослой жизни, несмотря на изменяющиеся окружающие обстоятельства.

Предполагается, что наблюдается некая связь между чертами личности, поведением и достижениями в жизни, хотя эта связь не ярко выражена и в большой степени зависит от контекста. Опросы среди сотрудников предприятий показали, что сознательность обычно соотносится с профессиональной работоспособностью. Но в более специфических профессиональных контекстах конкретные достижения больше зависят от других черт – например, экстраверсия соотносится с успехом в области продаж и более высокопоставленными должностями. Изучалось также влияние черт личности в других разрезах – например, в связи со школьной успеваемостью или здоровым образом жизни (John and Srivastava, 1999, 121).

При всем этом модель «Большой пятерки» не часто используется в гейм-дизайне. Она может помочь при размышлениях о том, чем мотивируются различные типы игроков, но сама по себе слишком широка, ее трудно применить к специфическим задачам гейм-дизайна. В следующем разделе мы опишем некоторые новые теоретические находки, связывающие типы личности непосредственно с предпочтениями в играх и открывающие многообещающие возможности для дальнейших исследований.

Профили игровой мотивации Йи

Помимо исследований в области личностных черт, недавно была проведена еще одна научная работа, попытавшаяся дать ответ на вопрос о мотивации игроков.

Ник Йи из Quantum Foundries провел долгосрочное исследование среди геймеров, предлагая им оценивать свои предпочтения для разных аспектов геймплея. Вопросы работы основывались на двенадцати типах внутриигровой мотивации, позаимствованных из различных моделей, включая ранние модели психологических исследований мотивации, а также более традиционные дизайнерские теории (Yee, 2016, 12).

Экшен-аспекты

Разрушение – удовольствие от хаоса, беспорядка, стрельбы, взрывов.

Возбуждение – удовольствие от напряженных и быстрых игр.

Социальные аспекты

Состязание – удовольствие от соревнования с другими игроками.

Общение – удовольствие от взаимодействия и сотрудничества с другими игроками.

Аспекты мастерства

Вызов – удовольствие от преодоления трудностей, предпочтение играм, в которых нужно проявлять мастерство.

Стратегия – удовольствие от игры, требующей принятия обдуманных решений и

планирования.

Аспекты достижения

Завершение – желание выполнить каждое задание, получить каждый коллекционный предмет и спрятанный объект.

Влияние – удовольствие от власти и влияния в контексте игрового мира.

Аспекты погружения

Фантазия – желание стать кем-то другим.

Сюжет – важность разработанной сюжетной линии и интересных персонажей.

Творческие аспекты

Дизайн – стремление к самовыражению и глубокой кастомизации.

Открытие – желание исследовать, создавать и экспериментировать с игровым миром.

Попросив игроков оценить себя по этим двенадцати типам мотивации, Йи обнаружил последовательные корреляции с сообщениями игроков о том, какие аспекты игр им нравятся. Игроки, сообщавшие о своих высоких предпочтениях относительно какого-то одного вида мотивации, также выказывали предпочтение одному или двум другим схожим видам. Например, игроки, сообщавшие о том, что им нравятся стратегии, чаще утверждали, что им нравится вызов, но для таких измерений, как общение или фантазия, их ответы были менее предсказуемыми.

В ходе исследования были составлены многочисленные диаграммы и двухмерные графики, в которых степень корреляции выражается расстоянием между точками. На иллюстрации 2.4 показан один из таких графиков – диаграмма для игроков из Западной Европы. Но, что любопытно, примерно такая же корреляция наблюдается и в других странах и культурах (Yee, 2016, 14–16).



Экшен «Бум!»	Социализация «Сыграем вместе!»	Мастерство «Нужно подумать»	Достижения «Хочу еще!»	Погружение «Жили-были...»	Творчество «Что, если?»
Разрушение Оружие. Взрывчатка. Хаос. Беспорядок	Состязание Дуэли. Поединки. Место в рейтинге	Вызов Практика. Высокая сложность. Трудные задания	Завершение Сбор коллекционных предметов. Прохождение всех заданий	Фантазия Отыгрыш роли	Дизайн Самовыражение. Кастомизация
Возбуждение Быстрота. Экшен. Скорпыва. Азарт	Общение Командная игра. Чаты. Взаимодействие	Стратегия Предварительный расчет. Принятие решений	Влияние Сильные персонажи. Мощное снаряжение	Сюжет Продуманная история. Интересные персонажи	Открытие Исследование. Создание. Эксперименты

Иллюстрация 2.3. Диаграмма типов мотивации игроков (Yee, 2016, 12). Печатается с разрешения. Copyright 2016 Quantic Foundry

На основе этих корреляций авторы предложили разбить типы мотивации на три *группы (кластера)*, внутри которых показатели часто встречаются вместе.

1. *Экшен-социальный кластер*: возбуждение, разрушение, общение, состязание. Соответствует энергичной, напряженной и совместной игре с другими людьми.

2. *Кластер мастерства-достижения*. Достижение, стратегия, вызов. Соответствует различным способам прохождения игры, достижения мастерства, влияния внутри игры.

3. *Кластер погружения-творчества*. Сюжет, дизайн, фантазия. Охватывает различные повествовательные аспекты игры и дизайн мира.

Кластеры показывают, что в мотивации игроков могут наблюдаться определенные закономерности. Например, близость стратегии и вызова (или сюжета и дизайна) предполагает, что игроков, которым нравятся оба этих аспекта, будет больше, чем тех, которым, нравятся одновременно вызов и дизайн. А открытие и влияние находятся примерно посередине между кластерами и служат «связующими звеньями», привлекательными для широкой аудитории.



Иллюстрация 2.4. Диаграмма расстояний между типами мотивации игроков (Yee, 2016, 14). Печатается с разрешения. Copyright 2016 Quantic Foundry

Некоторые выводы, сделанные на основе анализа кластеров, довольно неожиданны. Например, почему удовольствие от состязания и удовольствие от общения попадают в одну группу, хотя на первый взгляд противоречат друг другу? Авторы утверждают, что эта группировка имеет смысл, поскольку отражает такую черту личности игрока, как экстраверсия, при этом один игрок может как сотрудничать с другими игроками, так и состязаться с ними.

Хотя данные свидетельствуют о том, что аналогичные соответствия наблюдаются среди различных групп игроков из разных стран и говорящих на разных языках, в данный момент мы еще не знаем, каковы причины такого распределения, а также связано ли оно со знакомством с существующими играми (стилями, жанрами и т. д.) или же отражает какие-то черты личности, существующие на глубинном уровне. Но, несмотря на то что модель еще находится на ранней стадии разработки, в целом такой подход полезен, так как исследователи начинают собирать информацию о предпочтениях игроков и пытаются сгруппировать ее на основе эмпирически наблюдаемых данных. Эти типы мотивации и кластеры помогут нам, гейм-дизайнерам, точнее предсказывать, какого рода впечатлений ожидают от игр игроки.

Мотивация игроков и «Большая пятерка»

В последующем исследовании Йи предлагал игрокам пройти опрос о мотивации и чертах личности, а затем постарался обнаружить корреляции между их мотивацией и личностным профилем.

Полученные в результате исследования результаты выражены не слишком четко, но при этом наблюдаются некоторые любопытные соответствия (Yee, 2016). Они говорят о потенциальных корреляциях между тремя из пяти черт личности и некоторыми типами мотивации игроков, тогда как оставшиеся два типа личности не имеют соответствий.

- Большая открытость коррелирует с несколько большим предпочтением, отдаваемым фантазии, сюжету, дизайну и открытиям.
- Большая экстраверсия коррелирует с несколько большим предпочтением, отдаваемым возбуждению, состязанию и общению.
- Большая сознательность слабо коррелирует с несколько большим предпочтением, отдаваемым стратегии.
- Доброжелательность и нейротизм, не демонстрируют особой корреляции с типами мотивации игроков, но причины этого не ясны.

Как говорилось выше, эти корреляции между чертами личности и типами мотивации игроков высветили довольно неожиданные результаты кластеризации. Например, состязание и общение – это очень разные типы мотивации, но оба они соотносятся с экстраверсией и желанием взаимодействовать с другими людьми (как посредством кооперации, так и благодаря доминированию).

Любопытно, что результаты исследований идут вразрез с общераспространенным мнением о том, что игры – это эскапистские фантазии, в которых люди воображают себя тем, кем они на самом деле не являются. Йи утверждает, что данные говорят об обратном – игры, в которые мы играем, отражают наши черты личности и сообщают о том, что нам действительно кажется привлекательным.

Другими словами, *игры позволяют нам сильнее выразить себя самих такими, каковыми мы уже являемся.*

Дизайн опыта

Так что же делает игру привлекательной?

Мы описали несколько способов взглянуть на данный вопрос и пришли к выводу, что «привлекательность» – это не внутреннее свойство программы, а скорее функция как игры, так и конкретного игрока, играющего в нее.

Можно попытаться развить ряд идей о том, почему разным людям нравятся различные аспекты игр, и проанализировать, что же именно привлекает конкретных игроков в ряде моделей игрового опыта. Использовать дизайнерские теории обычно легко, к тому же они служат великолепным источником идей. Точно так же «насосом интуиции» могут послужить списки наших собственных предпочтений – представив опыт игроков в перспективе от первого лица, мы лучше поймем игроков и их впечатления. Можно также воспользоваться данными проведенных эмпирических исследований и сгруппировать игроков по их ответам о своих предпочтениях – так мы поймем, что доставляет им удовольствие, а также как предпочтения игроков группируются в кластеры, поддерживая (или не поддерживая) друг друга.

Такие концептуальные модели помогут нам заранее представить опыт участника. Даже если мы не можем пережить его непосредственно, сведения о типах игроков и о типах мотивации помогут предсказать реакцию игроков на различные виды геймплея и механики, над которыми надо работать.

Вопросы, которыми можно руководствоваться при дизайне опыта

Предположим, мы задумали создать какую-то определенную игру и хотим проанализировать опыт игрока. Для этого следует переходить от общих вопросов вроде: «В чем заключается привлекательность игр?» – к вполне конкретным: «Что сделает привлекательной *эту игру* ?» или «Что сделает привлекательной *мою игру* для *моих игроков* ?»

Итак, при разработке новой игры следует задавать вопросы о ее идее, исходя из контекста игроков.

• *Для кого я делаю эту игру?* Предназначена ли она для специфического типа игроков или должна казаться привлекательной для многих, самых разных типов людей? Как мне узнать, кто будет ее игроками? Эти вопросы можно рассматривать как в свете мотивационных моделей, так и попробовав составить стереотипные «портреты игроков». Но в любом случае этот вопрос будет первым и от него будут зависеть последующие.

• *Какого рода удовольствие игроки будут получать от данной игры?* Этот вопрос непосредственно связан с нашей аудиторией. Если исходить из предположений о том, какими будут наши игроки, то что именно в игре покажется им привлекательным? Гейм-дизайнеры обычно отвечают на данный вопрос на основе богатого собственного опыта. Мы также можем опираться на модели мотиваций, чтобы представить, какие аспекты игры понравятся определенному типу игроков. С узко жанровыми и нишевыми играми это решается относительно легко. В случае с крупными, ориентированными на широкий рынок играми, по-прежнему нужно понять, почему игроки будут получать от них удовольствие, но их при этом нужно будет поделить на множественные подгруппы, связанные с пристрастием к различным аспектам игры. В данном случае портреты игроков становятся особенно полезными.

• *Что еще из того, чего нет в игре, может доставить игрокам удовольствие? Что им не понравится?* Например, если игра посвящена военным сражениям с упором на состязание и стратегию, неплохо будет добавить в нее возможности для общения и взаимодействия – такие как гильдии или чаты. Но если вместо них мы решим добавить сюжет, то сначала нужно подумать, насколько он окажется уместным для данной целевой аудитории. Другими словами, можно ли предусмотреть какие-то дополнительные источники удовольствия, имея представление об аудитории игры? И какие, не связанные с основным, источники удовольствия можно добавить, чтобы игроки не теряли сосредоточенности на основной цели, а игра не казалась слишком размытой, ориентированной на абсолютно разные группы игроков?

Эти вопросы – лишь отправные точки для исследования, но они помогают сосредоточиться на главных пунктах – какова наша аудитория и как ее можно заставить получать задуманные нами впечатления от игры.

Архетипы и жанры игрового опыта

Мы уже не раз говорили о существовании жанров. При описании игр, помимо прочих, используются такие термины, как «*военная игра*», «*стратегическая игра*» или «*симулятор*». Это популярные наименования жанров, а их бывает огромное количество: «*стратегии в реальном времени*», «*шутеры от первого лица*», «*шутеры-арены*», «*ролевые игры*», «*песочницы с открытым миром*» и тому подобные. Их определения могут быть достаточно размытыми, но обычно как игроки, так и разработчики понимают, что имеется в виду.

Можно утверждать, что жанры – это краткие описания ожидаемого игрового опыта.

Очень часто они подразумевают определенные механики, системы и виды геймплея. Например, если я слышу, что какая-то игра относится к жанру «*тайкун*», то само это название порождает определенные ожидания – что в игре будет много менеджмента и очень большое внимание будет уделено тщательной экономической симуляции с элементами стратегии и соревновательных элементов; возможно, это будет однопользовательская игра без социальной составляющей. Точно так же я получаю представление об игре, услышав, что она относится к «*ролевым играм с открытым миром*», к «*тактическим командным шутерам от первого лица*» и т. д.

Жанры удобны в процессе *создания* игры, поскольку помогают быстро принять большое количество дизайнерских решений, тогда как в противном случае много времени пришлось бы потратить на тесты и разработку прототипов. Выбор жанра позволяет ограничить выбор геймплея и свести возможные игровые системы к небольшому, удобному

для исследования набора. Например, если мы разрабатываем «песочницу с открытым миром», то уже многое знаем об игроках такой игры – чем они будут в ней заниматься, о существительных и глаголах, которые потребуется включить в игру, и т. д. Раз фундамент уже определен, можно сразу приступать к разработке чего-то нового и интересного.

С другой стороны, кросс-жанровые, или «ломающие жанр», игры бывают не менее интересными, потому что в них необычно, по-новому смешиваются различные элементы и не оправдываются типичные ожидания. Работать над кросс-жанровыми играми бывает довольно трудно, потому что приходится совмещать механики разных жанров и следить за тем, чтобы они были хорошо подогнаны друг к другу. Но когда такое сочетание срабатывает, то на свет появляется новая игра, находящая отклик среди разных групп игроков с весьма разными видами мотивации.

Краткий итог

В данной главе мы перечислили некоторые методы анализа игрового опыта, с помощью которых можно получить представление о мотивации игроков и о том, что же они находят привлекательным в играх. Ключевые ее положения таковы.

- Опыт – понятие относительное. То, что одни игроки находят привлекательным, совсем не интересует других, а разные игроки могут получать удовольствие от одной и той же игры по разным причинам. Как следствие, нельзя ожидать какой-то «объективности» в оценке игры и получаемого от нее удовольствия. Такого рода впечатления всегда соотносятся с *конкретными игроками* и их мотивацией.

- Чтобы понять, что делает игры привлекательными, нужно для начала исследовать игроков и их мотивы. Можно попытаться построить *теории игроков*, которые помогут нам предсказывать реакцию определенных типов участников на определенные игровые ситуации.

- Существует ряд *дизайнерских теорий*, объясняющих мотивацию игроков на основе опыта разрабатывающих игры дизайнеров и их наблюдений за участниками. Также разработчики пользуются методом «портрет игрока» для описания стереотипного поведения и типов мотивации на основе исследований широкого круга игроков.

- Недавние исследования с широкомасштабными опросами, предпринятые Ником Йи, позволили создать эмпирическую модель предпочтений игроков, разделенных на двенадцать мотивационных типов. Эта модель позволила прийти к следующим предварительным заключениям.

- Судя по опросам, наблюдается деление мотиваций на три большие категории (*экишен, мастерство* и *творчество*), каждая из которых содержит несколько других, поддерживающих друг друга. Но при этом мотивы каждого игрока индивидуальны, и у любого могут наблюдаться различные их сочетания.

- Возможно, существует слабая корреляция между мотивационными кластерами игроков (выделенных на основе их собственных описаний) и их типом личности, согласно модели «Большая пятерка», но это предположение требует дополнительной проверки.

- Знание мотивации игрока помогает понять, как он будет реагировать на разрабатываемый нами игровой опыт, благодаря чему можно постараться предвосхитить его ожидания.

После экскурса в мотивацию игроков переключимся на механики и обсудим более приземленные подробности их воплощения. Следующая глава – «Механики» – посвящена низкоуровневым элементам игры. В главах 4–6 мы будем исследовать сочетание элементов, которое поможет сделать так, чтобы игроки получали те самые удовольствие и опыт, которые мы с вами обсудили ранее.

Дополнительная литература

Что касается получаемого от игры «удовольствия», то терминология здесь еще не устоялась. На ранних этапах дизайнеры слишком злоупотребляли словом fun («веселье, забава») с заранее определенным допущением того, что игроки должны переживать и ощущать. Это породило волну критики: самые известные ее примеры – работы Дага Черча (1999) и Грега Костикиана (2002), в которых использование слова fun называется слишком конкретным и ограниченным. В своих статьях они используют термин enjoyment («удовольствие») как менее нагруженное, но все же предполагающее интерес, внимание и вовлеченность субъекта. То, от чего мы получаем удовольствие, воспринимается все же больше как игра, а не как работа.

Психология игрока

Что касается темы исследований личности и ее модели «Большая пятерка», то очень подробное исследование истории и текущего состояния данной модели дают Джон и Сривастава (John and Srivastava, 1999). Интересующиеся читатели могут найти в интернете предварительную версию их работы (John and Srivastava, дата не указана).

Помимо моделей личности, полезно будет рассмотреть различные взгляды на психологию игрока. На уровне конкретных психологических эффектов интерес представляет работа *Getting Games* (Madigan, 2015), в которой описано, как различные индивидуальные психологические факторы (такие как стремление избежать поражения или «ловушка невозвратных затрат») порой очень сильно влияют на поведение игрока и как ими пользуются в гейм-дизайне. На более общем уровне книга «Виртуальные экономики» (*Virtual Economies*, Lehdonvitrta and Castronova, 2014) описывает игры как экономические системы и предлагает воспользоваться экономическими моделями для понимания стимулов и мотивов, объясняющих поведение игрока.

Пролить свет на мотивацию людей в целом, а не только в играх, помогает теория самодетерминации. Райан и Деси (Ryan and Deci, 2000) описывают ее общую конструкцию – существующие у человека потребности в компетенции, автономии и взаимосвязи с другими людьми, служащие мотивами для его действий. Райан с коллегами (Ryan et al., 2006) предлагает применить эту теорию для предсказания получаемого от игр удовольствия, а Дин (Deen, 2015) подробно исследует возможность использования ее конкретно в гейм-дизайне.

Дизайнерские теории

Помимо эмпирических исследований, с течением времени были выдвинуты различные дизайнерские теории. Кроме описанных в этой главе моделей Бартла и Костера можно отметить еще две, достойные особого упоминания. Дополнительный пример классификации (таксономии) можно найти в книге Йи «Профиль мотивации игрока: модель и открытия» (*Gamer Motivation Profile: Model and Findings*, Yee, 2016).

Одна из ранних моделей мотивации игроков – классификация Кайуа (1962), делящая игровой опыт на четыре основные категории:

- «агон» (примерно соответствует «состязанию»);
- «алеа» (примерно соответствует возбуждению от азартных ситуаций);
- «мимикрия» (примерно соответствует «фантазированию» или «воображению»);
- «илинкс» (примерно соответствует скорости, повышенной физической активности).

Метод Кайуа основан на интуиции и наблюдениях, сделанных в процессе исследования культуры и антропологии. Многие последующие классификации «удовольствия» от игры

также уходят своими корнями в интуитивный подход. Впрочем, в практике дизайна эта теория не используется и представляет собой только исторический интерес.

В отличие от нее предложенная относительно недавно модель «*Четырех факторов*» Лаззаро не только служит классификацией (таксономией) игрового опыта, но и в более широком смысле рассматривает модель мотивации игрока, объясняя различные аспекты игрового опыта. В модели «Четырех факторов» удовольствие от игры делится на четыре типа.

- *Удовольствие от трудности* (*fiero*) соотносится с их преодолением и решением сложных задач.
- *Удовольствие от легкости* соотносится с любопытством и радостью от исследования.
- *Удовольствие от серьезности* соотносится со сосредоточенностью и целеустремленностью.
- *Удовольствие от общения* соотносится с социальными связями и заботой о других.

Соответствующее исследование содержит практические объяснения связи этих переживаний с различными эмоциональными реакциями и дает советы, как можно добиться их с помощью конкретных элементов гейм-дизайна (Lazzaro, 2004). Работа рекомендуется к прочтению тем, кто интересуется конкретными примерами игр и пробуждаемых ими эмоций.

Упражнения для индивидуального выполнения

2.1. Относительность опыта

Выберите одну из ваших любимых игр. Обсудите, почему вы получаете удовольствие от этой игры? Перечислите три лучших по вашему мнению аспекта геймплея (механики, действия, системы и т. д.) и три аспекта, не относящихся к геймплею (визуальное оформление, тема, сюжет, персонажи, музыка и т. д.). Опишите каждый аспект в одном предложении.

Представьте, что в эту же игру играет другой человек, обладающий другими вкусами и предпочтениями, которому нравятся в ней иные элементы. Опишите этого игрока. Кто он, каковы его три самых предпочитаемых аспекта в каждой категории?

2.2. Модель Бартла

Вспомните некоторые из используемых вами многопользовательских игр (онлайн или по локальной сети). Замечали ли вы, что игроки в них делятся на описанные Бартлом категории? Соответствуете ли вы одному или нескольким из этих типов? Или же ваши мотивы не описаны в модели Бартла?

2.3. Портреты игроков

Вернитесь к своей любимой игре, которую вы рассматривали в упражнении 2.1. Представьте несколько различных «типов» игроков, получающих от нее различное удовольствие и опишите их как конкретных людей (портретов игроков). Кто они, что им нравится, какое место в их жизни занимают игры? Постарайтесь учитывать не только свой опыт, вообразите разных игроков с различными жизненными предпочтениями.

2.4. Модель Йи

Опишите свои игровые привычки и предпочтения в свете предложенной Йи модели двенадцати категорий мотивации игроков. Какая из этих мотиваций лучше всего отражает вашу? Какие типы не находят отклика? Принадлежите ли вы к какому-то кластеру или ваши мотивы более разнообразны?

2.5. Вопросы, которыми можно руководствоваться при дизайне опыта

Выберите одну из хорошо известных вам игр, представьте, что вы ее дизайнер, и ответьте на три вопроса из данного раздела.

а. *Для кого я делаю эту игру?* Особенно отметьте, какой тип мотивации удовлетворяет эта игра, а какие типы она не затрагивает.

б. *Какого рода удовольствие игроки будут получать от данной игры?* Какие элементы геймплея (действия, механики и т. д.) являются основными именно для этого типа мотивации игроков?

в. *От чего еще они будут получать удовольствие?* Если бы вам нужно было как-то дополнить игру, то какие элементы на основе существующего геймплея, по вашему мнению, послужили бы хорошим дополнением с учетом предполагаемой мотивации игроков?

2.6. Архетипы и жанры игрового опыта

Выберите знакомый вам жанр игр. Какие из предложенных Йи типов мотивации призваны удовлетворять его? Какие типы мотивации соотносятся с данным жанром меньше всего?

Подумайте, какая из известных вам игр задумывалась как кросс-жанровая или смешанного типа? Удовлетворяет ли она полностью мотивацию игроков двух этих жанров или же только некоторые из типов мотивации, не уделяя внимания остальным?

3 Механики

В предыдущей главе мы исследовали опыт игрока – то, как игроки воспринимают игры и взаимодействуют с ними, а также мотивацию, которой они могут руководствоваться во время игры. Обратимся же теперь к базовому уровню механик.

Наша задача как гейм-дизайнеров – создавать опыт. Хотим ли мы испытать стратегические навыки игроков, собираемся ли поведать им захватывающую историю, предложить экшен-сцену с выбросом адреналина или какой-то другой вид опыта – наша работа заключается в том, чтобы воплотить свое представление в реальной игре, в которую можно будет играть.

Трудность заключается в том, что такой опыт (переживание) нельзя вызвать непосредственно. Мы можем только манипулировать базовыми конкретными компонентами – игровыми объектами, правилами, персонажами и другими различными элементами, с которыми будет взаимодействовать игрок. Необходимо скомпоновать их и спланировать все так, чтобы у игрока возникли хорошие ощущения от игры, и он получил удовлетворительный опыт.

Рассмотрим теперь самые базовые элементы. В данной главе будут разобраны механики в отрыве от других аспектов дизайна, а в следующей мы поговорим о том, как они образуют системы, с которыми взаимодействуют игроки. В последующих главах перейдем к геймплею, но пока сосредоточимся только на этих базовых элементах.

Механики как строительные блоки

Термин «механики» описывает основные виды действий в игре и регламентирующие их правила. Начнем с примеров. В *покере* механиками можно назвать раздачу карт, повышение ставок, их уравнивание и т. д. – все эти основные действия описаны в правилах игры. В игре *«Монополия»* основные действия – это перемещение фишек по игровому полю, приобретение недвижимости и владение ею, получение ренты от других игроков, модернизация недвижимости (зданий и отелей). В таком «файтинге», как *Tekken*, базовыми действиями можно назвать удары ближнего боя и дальние атаки, защиту посредством

постановки блоков и парирования, распределенные по времени эффекты вроде усиления атаки и «отката», а также наступление и отступление ради получения стратегического преимущества.



Иллюстрация 3.1. Роль дизайнера: разработка механик и систем для достижения задуманного результата

Будем рассматривать механики как доступные игроку базовые действия и механизмы. Они в своей основе сводятся к реализации нескольких моментов: в случае с настольными играми – к пластиковым фишкам и книжкам с описанием правил, а в случае с компьютерными играми – к алгоритмам и структурам данных. Но сами по себе эти пластиковые фишки и правила не являются механиками – это их составляющие. Чтобы получилась механика, такие компоненты должны *объединяться конкретным и значимым способом*. Например, в *покере* отдельная карта – это не механика, как не является механикой и действие по перемещению фишек в банк. Но раскрытие своей «руки» – механика, базовое действие в игре с использованием таких простейших элементов, как карты и фишки.

Состав механик

Мы выяснили, что механики состоят из более простых компонентов, объединяемых конкретным и значимым способом. Теперь более подробно опишем эти составляющие части. Наиболее типичные компоненты механик следующие.

- *Объекты*, которыми можно действовать или манипулировать.
- *Виды действий*, которые можно осуществлять с помощью объектов.
- *Правила*, определяющие допустимые действия – какими объектами и в какие моменты можно действовать, к каким результатам приводят эти действия.

Пожалуй, самый интуитивно понятный элемент – это *игровые объекты*. В физических играх много примеров реальных объектов, которыми манипулируют. В *покере* имеются фишки, стол и банк с деньгами на нем. В *шахматах* имеются доска, фигуры и иногда часы. В *Settlers of Catan* – игровое поле, клетки с улучшениями, карты ресурсов и кости.

В играх также могут быть *виртуальные и абстрактные объекты*.

Виртуальные лишь изображают физические – например, листы персонажей ролевых игр заполняются описаниями мечей и боевых топоров, золотых монет, напитков и магических свитков, представляющих собой лишь символы на бумаге (или на экране), но функционирующих как объекты, которыми можно манипулировать. С *абстрактными* несколько сложнее. «Рука» игрока в *покере* – это просто набор других элементов, но иногда его следует трактовать, как отдельный объект. Также в *покере*, *шахматах* и *Settlers of Catan* имеется понятие «хода», который выполняет один из игроков, но это понятие имеет смысл, потому что некоторые действия доступны только игроку, выполняющему ход именно в данный момент, но при этом нет никакого объекта, физического или виртуального, соответствующего этому самому «ходу».

Другой вид элементов механики – это *игровые действия*. Если у нас имеются объекты,

то мы должны выполнять какие-либо действия с ними. Игроки в *покер* берут карты из колоды, делают ставки, повышают их с помощью фишек. Игроки в *шахматы* передвигают фигуры на доске, захватывают фигуры противника или теряют собственные, а иногда и заменяют одну фигуру на другую. В игре *Settlers of Catan* участники бросают кости для случайного определения клетки для сбора урожая, накапливают ресурсы, торгуют и строят улучшения. Гейм-дизайнер может придумать и определить практически любые действия. Они же, в свою очередь, могут влиять на объекты и их перемещение по игровому полю, менять свойства элементов, создавать или удалять их из игры и т. д. – возможности здесь безграничны и специфичны для каждой конкретной игры.

Правила игры описывают, какие именно действия допустимы в данной программе. Например, в *шахматах* описывается, как перемещать фигуры на доске, как можно их захватывать (в стандартной ситуации или «на проходе») и т. д. Как *шахматы*, так и *го* славятся лаконичностью своих правил, порождающих, тем не менее, огромную комбинаторную сложность. Правила других игр бывают гораздо сложнее. Возглавляют их список, пожалуй, такие коллекционные игры, как *Magic the Gathering* и военные.

Правила, действия и объекты взаимосвязаны между собой. Говоря о том, что в карточной игре «берем карту из колоды», мы описываем действие над соответствующими игровыми объектами, производимое согласно правилам, которые иногда указываются явно, а иногда просто предполагаются из контекста. Пример явно указанного правила – брать карту может только игрок во время своего хода, и только одну. Но некоторые правила настолько распространены, что их явно не указывают – например, что брать можно только верхнюю карту в колоде или что карта пополняет «руку» именно этого игрока, а не какого-то другого.

Такие компоненты используются в разных ситуациях, а не только при описании механик. Например, среди правил *шахмат* имеются определяющие победу или поражение в партии, а также определяющие счет игрока в рейтинговых матчах. Условия победы – это не механика, по крайней мере не в нашем понимании слова. Это цель, которую преследуют игроки с помощью механик, а уже они образуют собой часть *метаигры*, о которой мы поговорим позже, в главе 6 «Макроструктура». Игровые правила можно воспринимать как набор различных регулирующих игру положений. К ним относятся не только правила о том, как действуют и взаимодействуют объекты, но и какие-либо дополнительные предписания.

Языковая метафора

Согласно языковой метафоре, механика игры сравнивается с человеческим языком.

- Игровые объекты уподобляются существительным, поскольку те описывают объекты.
- Игровые действия уподобляются глаголам, поскольку те описывают действия.
- Правила игры уподобляются грамматике, поскольку она описывает сочетания существительных и глаголов.

Эта метафора используется широко, и вопросы вроде: «Каковы существительные и глаголы этой игры?» – используются как сокращение вопроса: «Какими объектами может манипулировать игрок и какие действия может выполнять в данной игре?»

Эта же метафора позволяет нам во время разработки механики игры задуматься о ее дополнительных вспомогательных элементах.

- *Прилагательные* уточняют существительные: *слабая* или *сильная* броня.
- *Наречия* уточняют глаголы: прыгнуть *высоко* или *низко*.
- *Предлоги* уточняют отношения: находится *на* столе или *под* столом.

Вернемся к идее таких «уточнений» в другом разделе, когда будем разбирать конкретные примеры механики управления ресурсами.

Пример: разбор «Монополии»

Для иллюстрации рассмотрим игру «Монополия» и постараемся выделить ее механики. Мы уже упоминали базовые элементы данной программы.

1. *Объекты, которыми могут манипулировать игроки* : игровое поле, кости, фишки, банкноты, участки недвижимости, дома, отели, набор карт случайных событий и т. д.
2. *Действия, которые могут выполнять игроки* : бросать кости, передвигать фишки по полю, вставлять на клетки, покупать, платить ренту, брать случайную карту и т. д.
3. *Регулирующие игру правила* : какие действия в каком контексте можно выполнять; какие необходимы ресурсы, как их тратить и получать; что происходит в результате действий.

Можно постараться выделить отдельные сочетания объектов, действий и правил, задавая себе один за другим вопросы о том, как устроена игра.

Кто я как игрок? Я представлен игровой фишкой на игровом поле (доске).

Что я делаю? Перемещаюсь по доске, а затем реагирую в зависимости от того, где остановился.

Как я двигаюсь? Бросаю шестигранную кость и двигаюсь по часовой стрелке на выпавшее количество ходов.

Что я делаю, остановившись на клетке? Это зависит от типа клетки. Если я останавливаюсь на клетке с чьей-то недвижимостью, то плачу ренту в зависимости от характеристик этой недвижимости.

Как я рассчитываю сумму, которую нужно выплатить? Существует базовая рента и модификатор, зависящий от того, входит ли этот объект в набор и имеет ли он улучшения.

Что такое набор? Это ряд прилегающих друг к другу клеток одного цвета, которыми владеет один и тот же игрок.

Что такое улучшение? Это дом или отель, который разместил владелец клетки.

Как это влияет на ренту? В диаграмме указано, как меняется рента в зависимости от набора или улучшений.

При желании этот ряд вопросов можно было продолжать, выясняя дополнительные подробности.

Такого рода «сократические диалоги» позволяют углубляться в детали игры. Мы уже выделили некоторые основные механики «Монополии»: перемещение по игровой доске, механику ренты для сбора денег при остановке игрока на чьей-то собственности, связанную с ней механику составления набора из соседних клеток для получения дополнительной ренты, и т. д. При разработке новой игры необходимо продумать все эти механики и разобрать их на объекты, действия и правила, чтобы описать в книге правил или закодировать в компьютерной программе.

Игры как пространство состояний

В связи с игровыми механиками можно познакомиться с такими используемыми гейм-дизайнерами терминами, как «состояние игры», «пространство состояний» и «пространство действий». Ими оперируют, рассуждая о развитии игры со временем, о том, как игроки воспринимают свои действия и как могут повлиять на траекторию развития программы.

Состояние игры

Как однажды выразился гейм-дизайнер Сид Мейер, игра – это «серия интересных решений» (2012). Крайне важный элемент гейм-дизайна – понимание выбора игрока и того, как он со временем меняет игру. *Покер* или *шахматы* – это не статичные игры. Игроки в них совершают какие-либо действия с помощью игровых объектов, и в результате этого меняется ситуация в игре и шансы игроков на победу.

Для описания развития игры со временем гейм-дизайнеры часто заимствуют терминологию из кибернетики и теории автоматов. Они рассуждают об игре и ее участниках как о динамической системе с *внутренним состоянием*. Взаимодействуя с игрой и друг с другом, игроки своими действиями меняют состояние системы, что, в свою очередь, заставляет их уточнять и исправлять решения, выполнять иные действия, тем самым снова меняя состояние системы, и так далее.

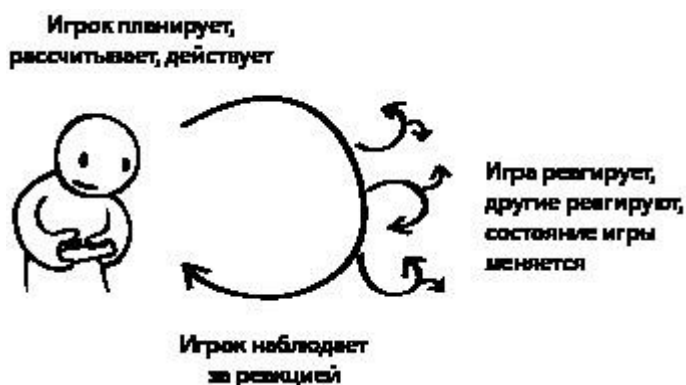


Иллюстрация 3.2. Действия порождают реакцию со стороны игровых систем и других игроков, на что игроку тоже приходится реагировать – так возникает непрерывный цикл

Иными словами, можно утверждать, что:

- *состояние игры* – это общее состояние всех игровых объектов и их свойств в определенный момент времени;
- действия в игре приводят к *переходу* между состояниями игры.

Состояние игры бывает полностью или частично наблюдаемым. Шахматы – пример игры с *полностью наблюдаемым* состоянием. Если посмотреть на шахматную доску в определенный момент времени, то она полностью описывает текущее состояние процесса. Можно сфотографировать доску, а позже точно так же расставить все фигуры на ней и продолжить сражение.

Но в большинстве игр имеется скрытая информация, их состояние только *частично наблюдаемое*. В программах вроде «*Монополии*» есть информация, доступная всем (например, кто является игроком конкретной недвижимости); есть та, что доступна только отдельному игроку (я могу не говорить остальным, сколько у меня денег), а есть информация, полностью недоступная участникам (порядок случайных карт в колоде). Общее состояние игры передает вся информация целиком, как видимая, так и скрытая. Если мы захотим сохранить состояние игры для его последующего воспроизведения, то придется сохранить и скрытую информацию.

Скрытое состояние чрезвычайно часто встречается в компьютерных играх, поскольку это очень мощное средство увеличения сложности игры (таких как головоломки, стратегии и т. д.), создания драматического напряжения (например, в сражениях и «хоррорах»), а иногда и намеренного введения игрока в заблуждение ради развития сюжета¹.

¹ Примеры скрытия или раскрытия информации в целях развития сюжета см. в книге Соларски «Ненадежный гейм-мастер» (*The Unreliable Gamemaster*, Solariski, 1017b).

Пространство состояний

С данной точки зрения игра – это серия переходов от одного игрового состояния к другому с течением времени. В шахматах количество возможных состояний и переходов огромно – обычно у игроков широкий выбор возможных действий, а игра всегда допускает переход в различные возможные состояния. Но в процессе каждый игрок может совершать лишь одно действие за ход, поэтому, несмотря на огромное пространство, игра движется по узкой траектории.

Рассмотрев все *возможные* состояния, в которых *может* находиться игра, можно добавить еще несколько определений.

- *Пространство состояний* – совокупность всех состояний, в которых может находиться игра, связанных между собою действиями и переводящих ее из одного состояния в другое.

- *Ширина пространства состояний* – общее количество состояний в пространстве состояний или количество разных состояний, к которым можно прийти из начального состояния игры.

- *Фактор ветвления состояния* – количество переходов (действий), допустимых из данного состояния.

Неформально можно говорить о «плотности» или «разветвленности» пространства состояний, когда у каждого состояния относительно высокий коэффициент ветвления. Если же этот коэффициент низок, то пространство можно назвать «разреженным», или «неплотным»². Понятие «плотности» относительно и зависит от жанра и ожиданий. Например, детские настольные игры не настолько плотные, как стратегические настольные игры для взрослых.

Плотность пространства и размер множества действий влияют на впечатления от игры. «Разреженная» игра, не предусматривающая особенно интересных решений (скажем, «крестики-нолики») не настолько увлекательна, как более «плотная». Но все зависит от конкретной аудитории. Маленьким детям нравятся «крестики-нолики», потому что в силу своей простоты программа лучше соответствует их еще только развивающимся когнитивным способностям. «Плотные» игры на другом конце спектра могут показаться интересными в силу своей сложности, но слишком большая плотность пространства состояний представляется помехой для новичков, только осваивающимся в процессе. Таким образом, вопрос о том, какую плотность можно назвать подходящей, субъективен.

Но даже во многих «разреженных» играх общее пространство состояний бывает довольно большим, что далеко не всегда понятно интуитивно. Подсчитать количество возможных вариантов состояний в такой простой игре, как «крестики-нолики», несложно: девять вариантов ходов для первого игрока, за которым следует один из восьми вариантов ходов для второго игрока, затем семь вариантов для первого игрока и т. д. – всего получается 362 800 возможных расстановок (хотя среди них есть и те, что не соответствуют правилам или дублируют друг друга, так что количество реальных позиций будет гораздо меньше).

Пространство «крестиков-ноликов» хорошо просчитывается и изучено досконально.

² Некоторые гейм-дизайнеры (Elias, Garfield and Gutschera, 2012) также рассуждают об «игровых деревьях», представляющих собой пространство состояний без циклов – другими словами, о тех случаях, когда состояние игры не может повториться (как в «крестиках-ноликах» или во многих карточных играх). Такого рода пространства состояний традиционно называются деревьями как в теории принятия решений, так и в информатике. Понятно, что определение «разветвленности» имеет гораздо больше смысла в применении к деревьям, чем к пространствам.

Коэффициент ветвления для *шахмат* или *го* настолько огромен, что полностью проанализировать все их пространство состояний не может даже большинство мощных компьютеров. Поэтому в программах для *шахмат* или *го* приходится прибегать к различным вспомогательным трюкам или эвристическим схемам, чтобы сократить количество просчитываемых состояний.

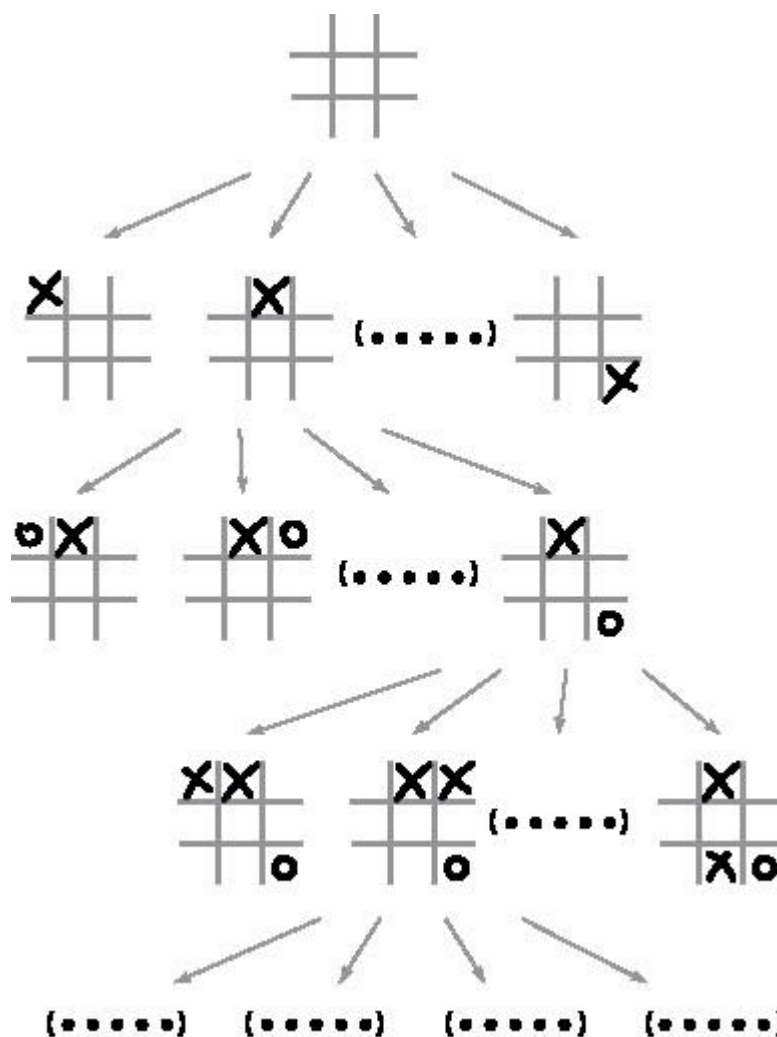


Иллюстрация 3.3. Иллюстрация небольшой части пространства состояний «крестиков-ноликов»

Пространство действий

Пространство состояний влияет на комбинаторную сложность игры, но ощущение контроля над процессом со стороны игрока и представление о возможностях в игре возникает благодаря тому, как эти состояния связываются между собой действиями, то есть благодаря топологии пространства состояний.

Формально *пространство действий* определяется как набор всех действий, доступных игроку. Но неформально гейм-дизайнеры столь же часто используют выражение «пространство действий игрока» как обозначение *действий, доступных в данном состоянии*.

Например, в *шахматах* общее пространство действий будет совокупностью всех возможных ходов в любое время, но неформально это только совокупность ходов, доступных игроку прямо сейчас. В начале новой партии белые ходят первыми, но количество вариантов ходов, доступных им из всего разнообразия на этом этапе, невелико: шестнадцать разных ходов для пешек и четыре хода для коней. И только со временем

пространство действий для них «расширяется».

В этой книге мы будем использовать термин «пространство действий» в его неформальном значении – как количество действий, доступных в определенном состоянии игры, а термином «общее пространство действий» будем обозначать все действия³.

Пространство действий играет очень важную роль и непосредственно влияет на восприятие сложности игры со стороны игрока. В «*крестиках-ноликах*» начальное пространство действий состоит из девяти возможных ходов, затем оно сокращается до восьми, семи и так далее, то есть постоянно уменьшается, что не доставляет большого удовольствия – в отличие, например, от *шахмат*, где пространство действий гораздо обширнее.

Во-вторых, как уже было упомянуто, со временем пространство действий может меняться, что представляет определенный интерес для игрока. В *шахматах* пространство действий в начале партии очень ограничено, но в середине игры (миттельшпиле) оно увеличивается, открывая игроку множество возможностей для контроля доски, а к концу игры (эндшпилю) снова уменьшается.

Отчасти привлекательность многих игр объясняется сложной задачей *контроля будущего пространства действий* – иными словами, необходимостью строить планы, заранее рассчитывать ходы в ожидании будущих возможностей и не загонять себя в угол в результате плохо обдуманных действий.

Воспринимаемое пространство действий

Под «пространством действий» понимают действия, доступные игроку в данной игре. Но столь же важно и *восприятие* действий игроком – его представление о том, какие действия доступны, а какие нет. Вот некоторые примеры случаев, когда воспринимаемое пространство действий может отличаться от реального.

- *Неизвестные действия*. Как игрок я могу не знать, что мне доступны какие-то действия. Например, будучи новичком в «файтинге» вроде *Tekken*, еще не научился выполнять блоки или уклонение, поэтому не знаю, что они мне доступны и что их следует учитывать.

- *Исключенные действия*. Даже если я знаю о каких-то действиях, то могу не принимать их во внимание, потому что они противоречат моим целям или стратегии и поэтому для меня как бы не существуют.

- *Иллюзорные действия*. Мне может показаться, что некоторые действия возможны, тогда как в действительности это не так. Например, передвигаясь по опасному скалистому выступу, я могу опасаться, что сделаю неверный шаг и полечу в пропасть, хотя в реальности гейм-дизайнер установил там невидимый барьер.

Такую совокупность действий, доступных в определенный момент времени, можно определить как *воспринимаемое пространство действий*⁴. Впечатления от игры и опыт игрока в очень большой степени зависят от его собственных представлений о том, что он может или не может делать в конкретной ситуации.

Воспринимаемые действия могут отличаться от реальных. Это часто используется для создания драматического эффекта – например, внушая мысль о том, что некоторые ситуации

³ Для устранения этой двусмысленности некоторые гейм-дизайнеры, например Аптон (Upton, 2015), для обозначения действий, доступных непосредственно или в ближайшем будущем, используют термин «горизонт действий», а другие (Wright, 2003; Salen and Zimmerman, 2004) описывают сочетание контекстуально доступных игроку действий и состояний термином «пространство возможностей».

⁴ В соответствии с указанным выше Аптон (Upton, 2015) использует термин «горизонт намерений».

опасны, хотя на самом деле они таковыми не являются. Но такие действия хотя бы в чем-то должны совпадать, иначе играть будет невозможно. Если представление игрока о доступных ему действиях постоянно расходится с реальностью, то он не сможет играть в игру, а если игрок долгое время воспринимает какие-то действия недоступными, то испытает разочарование, узнав, что заблуждался.

Явная и неявная механика

Написать правила для «*Монополии*» или *шахмат* не так уж и сложно. Гораздо труднее записать все правила какой-нибудь компьютерной игры, будь то стратегия вроде *Civilization*, нечто в духе «экшен», как *StarCraft*, или такая аркадная игра, как *Tekken*.

Здесь уже проявляется различие между явной и неявной механиками. Явно определенная механика – это то, что мы наблюдаем в физических играх вроде «*Монополии*» – все ее действия и объекты описаны в книге правил, а игроки видят перед собой все, что происходит в данной настольной игре. В таких играх игроки имеют полное представление о ее устройстве, а если не помнят какие-то подробности, то правила легко доступны и с ними можно свериться в любое время.

В видеоиграх же часто встречаются неявные механики, когда правила игрокам неизвестны заранее и часто существуют лишь в виде компьютерного кода, в который обычные игроки не могут заглянуть. В начале процесса игрок имеет представление лишь о некоторых механиках (благодаря обучающей кампании или наблюдению за другими участниками), но по мере игры открывает для себя новые механики и обучается им, узнавая, как они связаны между собой. Например, в файтинге *Tekken* разные персонажи выполняют различные движения, познакомиться с которыми игрок может посредством проб и ошибок. Или возьмем для примера военную стратегию, в которой игрок может столкнуться с новым типом вражеских юнитов, которых раньше не встречал, поэтому ему придется догадываться о том, как эти юниты ведут себя и как сражаться с ними.

Неявная механика доводит элементы неожиданности и открытия до такого уровня, которого трудно достичь в явно описанных играх; элемент награды за обучение в них тоже усилен. В таких играх игрок со временем осваивает разные механики и учится применять их в различных ситуациях. Но все эти эффекты в большой степени зависят от их реализации на практике, потому что скрытые правила могут слишком сильно усложнить игру по сравнению с той, правила которой известны заранее.

Некоторые программы находятся где-то между этими двумя крайностями. Например, у таких коллекционных карточных игр, как *Magic The Gathering*, есть явно сформулированные общие правила, но каждая карта может дополнять их. А поскольку редких карт могут быть десятки, то игрокам нужно покупать их отдельно, чтобы получить доступ к дополнительной информации, поэтому такие правила можно воспринимать как неявные. То же самое наблюдается и в настольных играх в стиле *legacy* («Наследие»), к которым прилагаются «секретные» дополнения к правилам; эти дополнения следует открывать в указанное время, они навсегда меняют правила игры – даже для будущих партий. Возьмем совершенно иной пример – сложную компьютерную военную игру. В нее можно играть так, словно многие правила неявные, но при этом существует очень объемное руководство. Взяв на себя труд прочитать его, самые преданные фанаты смогут ознакомиться со всеми правилами данной игры во всех деталях.

Дифференциация действий. Очень важно, чтобы игрок ощущал большое пространство действий. Но, помимо количества действий, огромную роль играют их отличия и разнообразные интересные особенности – чем больше таких отличий, тем лучше. Представим, какие заклинания могли бы фигурировать в любой фэнтезийной игре.

- Заклинания с разными названиями, но с одинаковыми ценами и эффектами.

- Заклинания с похожими, но немного разными ценами и эффектами.
- Заклинания с очень различающимися между собой ценами и эффектами.

Наибольший интерес для игрока представляет последний вариант. Выбор в первом варианте *иллюзорен*. Пусть внешне эти заклинания как-то и отличаются между собой, но по сути они действуют одинаково, а значит, и представляют собой один и тот же переход в пространстве состояний. В последних двух случаях свойства заклинаний различны, что создает разные направления развития игры – чем сильнее различия, тем интереснее анализ преимуществ и недостатков, тем увлекательней становится выбор.

Если предполагается, что игру можно будет проходить неоднократно, то крайне важно обеспечить разнообразие в пространстве действий, предусмотреть различные преимущества и цену разных действий. Если же предполагается, что игру будут проходить лишь один раз, то подходящими и весьма интересными могут оказаться даже иллюзорные или похожие виды выбора, но при повторных прохождениях игроки быстро поймут, что отличия между вариантами носят лишь поверхностный характер.

Различие между явными и неявными механиками – одна из причин, по которым бывает трудно использовать бумажные прототипы для компьютерных игр. Более подробно мы обсудим это в главе 7 «Прототипирование», но если говорить вкратце, то бумажные прототипы – это небольшие модели некоторых аспектов игры, выполненные с помощью таких физических материалов, как бумага и фишки. Часто гораздо легче и быстрее создать грубый бумажный прототип, чем писать код для игры на компьютере. Но хотя такие прототипы и хороши для тестирования явной механики, для неявной они годятся плохо, особенно если она основана на физической активности или подразумевает сложное управление ресурсами.

Примеры семейств механик

Вооружившись общепринятой терминологией, мы можем перейти к разбору примеров игровых механик в разных типах игр.

Неизвестно, сколько всего было разработано разных механик (не говоря уже о тех, что только предстоит придумать), поэтому невозможно перечислить их все. В нашей индустрии до сих пор отсутствует даже единая таксономия различных типов механики. Эта сфера гейм-дизайна слишком молода и еще не устоялась, хотя кое-какая работа в данном направлении уже ведется (см. раздел «Дополнительная литература» в конце данной главы).

Но даже сейчас можно рассуждать о механиках с точки зрения их широких «семейств». В этом разделе мы более подробно рассмотрим четыре семейства механик, а затем вкратце упомянем немного дополнительной информации.

Перечисленные ниже семейства механик широко распространены и постоянно встречаются в играх различных жанров, чем и заслужили подробное описание.

- *Механики контроля* : как игрок представлен в игровом мире и как его физические действия отражаются в контроле над игрой, в персонаже или в других аспектах процесса.

- *Механики прогресса* : предоставление обратной связи, информирование игрока о его прогрессе в игре и о том, как она меняется по мере прохождения ее игроком.

- *Механики неопределенности* : роль неопределенности (неизвестности, отсутствия сведений о том, что произойдет дальше), различные источники неопределенности, такие как случайность или скрытая информация, и их влияние на геймплей.

- *Механики управления ресурсами* : как игрок получает ресурсы и распоряжается ими; связанные с ними механики, валюты или запасы, которыми можно владеть в игре; как одни объекты можно преобразовать в другие или использовать («потреблять»).

- Конечно, есть множество других типов механик – специфических для форматов игры,

жанров или платформ. Вот лишь небольшой список примеров.

- *Механики шутеров от первого лица* : специфические для «стрелялок» действия, связанные с оружием и его свойствами, управлением вводом (например, механики прицеливания и передвижения), а также различными типами ролей, целей и динамики.

- *Механики гонок* : связанные с «автомобильными заездами», включая механики физической симуляции транспортного средства и его реакции на действия игрока, механики гоночной трассы, погодных условий и т. д.

- *Механики сражений в военных играх* : определяющие действия различных боевых единиц на поле боя и их влияние друг на друга; роль типов местности и даже более абстрактные концепции, такие как роль фактора внезапности в исходе боя.

- *Механики стратегий в реальном времени* : связанные с управлением войсками в динамичных тактических сражениях, а также с построением баз, управлением ресурсами и производственными цепочками.

- *Механики многопользовательских игр* : связанные с балансом игры, которая должна быть честной для всех игроков, но при этом предоставлять им вызов – например, посредством включения асимметрии, что приводит нас к рассуждениям о математической теории игр.

Последние из перечисленных семейств механик немного пересекаются друг с другом, поэтому было бы интересно исследовать их подробнее. Но мы не будем затрагивать данную тему, потому что она очень обширна, а механики носят специфический характер, зависящий от контекста.

Вместо этого рассмотрим четыре семейства более универсальных механик, указанных в начале этого раздела: механики контроля, прогресса, неопределенности и управления ресурсами.

Механики контроля

Многие игры ориентированы на управление игровым персонажем. В них игрок управляет своим «аватаром», который сражается, бежит, стреляет и действует внутри игрового мира. Такой «аватар» – общий элемент для многих жанров, от файтингов вроде *Tekken* или *Soul Calibur* до платформеров вроде *Super Mario Bros.* или *Prince of Persia* и от аркадных игр (таких как *Pac-Man* или даже *Asteroids*) до таких ориентированных на повествование, как *Uncharted* или *Journey*.

Экшен-ориентированные игры в основном ориентированы на быстроту рефлексов и отточенного мастерства в выполнении действий. Следовательно, если игра не успевает обрабатывать действия игрока и затрудняет управление персонажем, то игрок очень быстро это замечает и воспринимает как нечто неправильное. Даже если люди не могут объяснить, чем им не нравится игра, они все равно жалуются, что контроль в ней недостаточно «отзывчивый» или «не интуитивный», камера «дергается». Или просто испытывают разочарование, не осознавая почему.

Макэнти (McEntee, 2012) выделил три компонента экшен-игр, называя их «тремя С» (от английских слов character, camera и control).

1. *Персонаж* : кого или что представляет собой аватар игрока; что он может делать в игре, имеют ли его действия смысл, интересны ли они, можно ли их понять.

2. *Камера* : что видит игрок и как камера отображает его действия; под каким углом она показывает нужные объекты, не мешает ли играть.

3. *Контроль* : насколько хорошо устройство ввода переводит действия игрока в действия персонажа; ощущаются ли действия персонажа как интуитивные, точно ли они соответствуют действиям игрока; легко ли управлять персонажем и обучаться управлению.

Правила и свойства, определяющие поведение персонажа, можно назвать *механикой контроля*, *механикой передвижения* или *механикой боя* – в зависимости от конкретных особенностей игры.

Первые два компонента интуитивно понятны. В играх имеется *аватар персонажа*, олицетворяющий собой игрока. Если он не может осуществлять интересные действия или игрок не может понять, как заставить персонажа их выполнять, игрок начинает испытывать неприятные ощущения – разочарование, скуку, беспомощность, что сильно снижает впечатление от игры.

Еще один интуитивно понятный компонент – *камера*. Вид в играх бывает от первого лица (глазами персонажа) или от третьего (вид на игровой мир со стороны). Камера от первого лица показывает только то, что находится перед персонажем, угол ее обзора ограничен, а это может разочаровывать или создавать напряжение. Камера от третьего лица показывает намного больше, включая то, что находится за игроком, но ее движения иногда бывают неоднозначными. Обычно ожидается, что камера сама по себе следует за персонажем, показывая происходящие в игре действия, чтобы игрок не утруждал себя ее перемещением, а сосредотачивался только на игре. Но сделать камеру «умной» (показывающей происходящее под нужным углом и в нужный момент, но чтобы при этом игрок все же мог контролировать ее) бывает довольно трудно. Во время разработки игры порой очень много сил уходит на то, чтобы движения камеры не были заметны для игрока.

Третий элемент – *контроль*, то есть способы, с помощью которых игрок управляет своим персонажем. Для этого необходимо продумать систему ввода команд со стороны игрока и отображения соответствующих этим командам действий в игровом мире. В результате у игрока должно сложиться впечатление полного и безупречного контроля над персонажем. Например, как в игре про бокс будет выполняться удар – при нажатии на кнопку или с помощью джойстика, движения которого соответствуют движению кулака вперед и назад? И стоит ли предусмотреть дополнительную возможность ударить во время передвижения, а заодно и дать выбор между «высоким» и «низким» ударом? Что будет лучше для игрока и почему?

Трудности, с которыми гейм-дизайнеры сталкиваются при разработке средств контроля можно проиллюстрировать на примере вариантов преобразования ввода со стороны игрока в игровое действие.

Например, в экшен-играх существует проблема «сопоставления ввода и средств управления» (*input mapping*). Как правило, игроки в видеоиграх пользуются геймпадом или похожим средством управления, но часто в играх предусматривается гораздо больше действий, чем средств ввода на контроллере. В этом случае возникает непростая проблема сопоставления движений игрока с действиями персонажа в игре. В файтинге придется задуматься над следующим.

- Какие кнопки/средства ввода лучше всего подходят для конкретных действий и почему? Что в расположении или чувствительности делает их пригодными или непригодными для конкретного действия? Например, когда использовать кнопки под большим пальцем, а когда кнопки в верхней части контроллера или аналоговые джойстики и т. д.

- Использовать ли цифровые кнопки, принимающие только два состояния («включено» и «выключено») или аналоговые джойстики, допускающие непрерывное изменение вводимой информации? Для какого действия лучше подойдет цифровой ввод, а для какого аналоговый?

- Использовать ли сочетания (например, А + «вниз» для низкого удара, А + «вверх» для верхнего удара)?

- Использовать ли комбинации (например, нажимать кнопки А-В-А подряд для тройного удара)?

- Использовать ли удержание (например, нажать и удерживать кнопку А для

увеличения мощности удара)?

Однозначных ответов на эти вопросы не существует; гейм-дизайнеры должны проанализировать используемые игроками контроллеры и найти наилучшее решение для сопоставления физического ввода и игровых действий. Также следует учитывать ожидания игроков от игр определенного жанра и распространенные стандартные схемы управления для контроллеров.

Разобравшись с тем, какое средство управления отвечает за то или иное действие, перейдем к вопросу, что конкретно происходит во время ввода – например, каким образом Марио перемещается по уровню при нажатии кнопки или джойстика. Здесь мы сталкиваемся с разнообразными вариантами, о чем свидетельствуют приведенные ниже примеры.

В таких платформах, как *Super Mario Bros.*, при нажатии кнопки движения влево или вправо аватар перемещается с *постоянной скоростью*, падающей до нуля, если кнопку отпускают (то есть при ее «отжатии»). В некоторых играх (например, в *Donkey Kong Jr.*), скорость сбрасывается до нуля мгновенно, но в *Super Mario* персонаж сначала лишь немного замедляется – так же, как не сразу набирает скорость в начале движения, отчего возникает ощущение особой отзывчивости на нажатие кнопки. Прыжок тоже происходит по нажатию кнопки, но при краткосрочном *нажатии и отпускании кнопки* персонаж получает лишь слабый импульс в вертикальном направлении.

В играх, подобных *Asteroids*, использование кнопок заставляет корабль вращаться с *постоянной угловой скоростью*. Но при нажатии кнопки «вперед» или «назад» меняется не скорость, а ускорение движения. Корабль постепенно набирает или сбрасывает скорость, что затрудняет управление им, но сделано это намеренно.

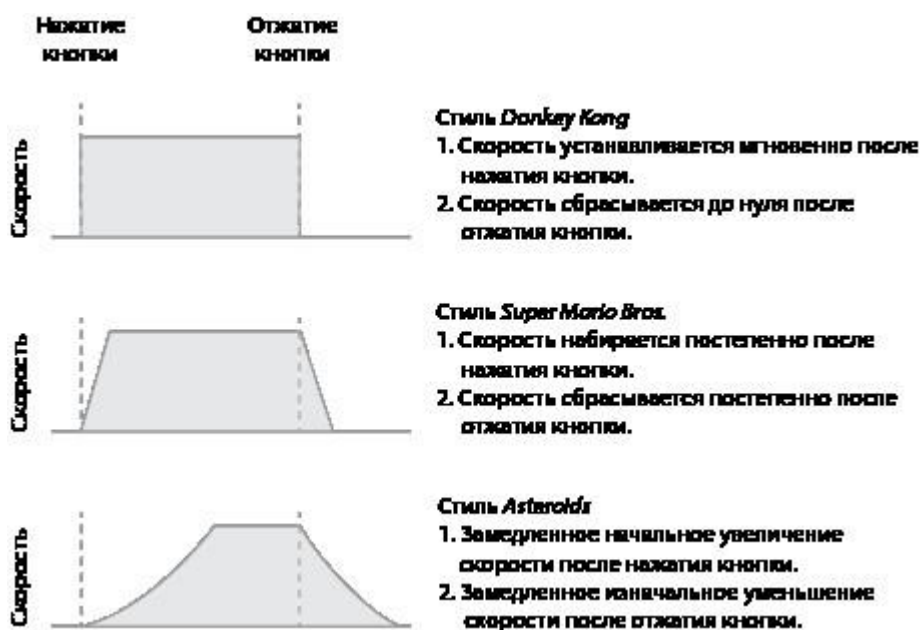


Иллюстрация 3.4. Примеры изменения скорости по нажатию кнопки, приведены по Swink, 2009

В современных шутерах от первого лица для персональных компьютеров игроки могут менять угол обзора персонажа, двигая мышью. Другими словами, мышь позволяет менять *угловое положение* камеры. В консольных же шутерах управление устроено иначе, в распоряжении игроков нет мыши, а есть только аналоговый джойстик, имеющий ограниченную чувствительность, поэтому его положение обычно соответствует *угловой скорости*. К сожалению, контроль скорости поворота камеры стиками геймпада происходит медленнее и не так точно, чем контроль положения камеры напрямую, поэтому в консольных играх, как правило, предусматривается компенсация этого неудобства в виде

автоматической доводки камеры и автоприцеливания .

Такие механики контроля иногда называют механиками «геймфила» («ощущения от игры») (Swink, 2009), поскольку они сосредоточены на том, как ввод со стороны игрока влияет на действия в игре и насколько легким или трудным ощущается взаимодействие с игрой. Более подробно об этом можно узнать из источников, указанных в разделе «Дополнительная литература».

Механики прогресса

Еще один вопрос, который нам предстоит задать, касается обратной связи: каким образом игрок будет узнавать о том, насколько хорошо или плохо он что-то делает? В игре *Asteroid* ответить на такой вопрос легко. Каждое успешное попадание увеличивает счет игрока, а на новых уровнях появляется больше астероидов и вражеских кораблей, которые нужно взорвать, поэтому игрок сразу же понимает, насколько далеко он продвинулся.

Но в ряде игр ответить на данный вопрос сложнее. В *go* или в *шахматах* игроки могут не иметь четкого представления о своем положении вплоть до конца игры. Для этого используются различные методы приблизительной оценки – например, подсчет оставшихся фигур и анализ занимаемых ими позиций, на основе чего составляется числовой показатель успеха.

Механики прогресса – это правила и элементы, позволяющие игроку получать обратную связь, то есть информацию о том, насколько хорошо он играет в игру и насколько далеко продвинулся в ней. Под «прогрессом» здесь понимается общий ход игры от начала к финалу. При этом полезно различать *прямые* и *непрямые механики прогресса* .

• *Прямые механики прогресса* непосредственно сообщают игроку о его продвижении в игре. При этом часто используются измеряемые показатели, связанные с успехом игрока. Вот примеры самых распространенных.

• *Очки счета* или *очки опыта* (XP) – число, которое увеличивается по мере того, как игрок совершает различные действия в игре (убивает врагов и т. д).

• *Уровни* , повышающиеся по мере прохождения игры.

• *Достижения* , присуждаемые за выполнение определенных целей, обычно не главных для игры.

• *Списки лидеров (таблицы рекордов)* , в которых показывается положение игрока относительно других участников.

• *Непрямые механики прогресса* меняют игру по мере продвижения игрока. Очень часто это проявляется в изменении игровой среды. В таких экшен-RPG, как *Diablo* , по мере прогресса игрок открывает новые территории, выглядящие иначе, чем прежние, встречает новых врагов или находит все более усовершенствованные предметы. Другой пример – увеличение пространства действий для игрока в ответ на его прогресс. В игре-симуляции жизни *The Sims* при улучшении ситуации растет материальное благосостояние семьи – она может позволить себе более дорогую и удобную мебель, переезжает в дом побольше или даже вообще может позволить себе не работать. Различные аспекты игры могут одновременно восприниматься игроками как показатели успеха и считаться частью механики прогресса. Например, о своем успехе игрок может легко судить по количеству накопленных денег или по величине армии.

Геймификация – предложенный относительно недавно термин описания различных видов механик прогресса для мотивации поведения вне игр. Например, сайт вопросов и ответов Stack Overflow присуждает очки опыта и достижения пользователям, внесшим по мнению сообщества определенный вклад в развитие сайта. При этом для них открываются новые возможности для публикации и модерации. Facebook и Twitter позволяет пользователям голосовать за сообщения друзей, ставя им «лайки», включая в списки «любимых сообщений» и «делясь»

ими с другими людьми. Эти продукты пытаются воспользоваться игровыми механиками для усиления привязанности пользователей к продукту. Но механика прогресса сама по себе еще не мотивирует игрока. Она лишь усиливает уже имеющуюся мотивацию.

Таким образом, геймификация – это вовсе не однозначный ключ к успеху. Она легко может обернуться провалом, если у игрока изначально нет мотивации участвовать в какой-либо деятельности. Механики прогресса лишь предоставляют пользователю информацию о его достижениях, но если сама по себе деятельность ему не интересна, то данные механики не сделают таковой. Подробнее о мотивации игрока мы поговорим в главе 6 «Макроструктура».

Как прямые, так и непрямые механики прогресса предоставляют игроку важную информацию о том, насколько хорошо он играет, а также поощряют продолжать игру. Счет или уровни могут быть связаны с разблокировкой определенного контента или объектов в игре, благодаря чему у игрока появляется цель двигаться дальше. Продвижение по сюжету также порождает у участника желание увидеть, как будет развиваться игра. Списки лидеров и достижения удовлетворяют его потребности в состязании и стремление выполнить задание, то есть затрагивают такие типы мотивации участника, как «состязание» и «завершение».

Данный мотивационный аспект механик прогресса может показаться манипулятивным, особенно когда они очевидны и откровенны. Но трудно представить себе игру, которая оставалась бы интересной, не предоставляя игроку информацию, насколько удачно он ее проходит. Считается хорошей практикой хотя бы постараться совместить данные механики с контекстом и с тематикой игры, иначе игроки попытаются судить о своем прогрессе по косвенным признакам, что далеко не лучшим образом скажется на их впечатлениях.

Механики неопределенности

Неопределенность в играх играет очень большую роль. Их исход зависит от многих факторов, действие которых бывает трудно предугадать, но игрок пытается уменьшить неопределенность посредством прогноза, анализа или тренировки. В чисто азартных играх, таких как *рулетка*, исход зависит исключительно от генератора случайных чисел, а вероятность успеха можно проанализировать статистическими методами. Но в *покере*, помимо неопределенности, благодаря элементам случайности появляются элементы стратегии и психологии. В *шахматах* случайность сведена до нуля, но все же они настолько нагружают умственные способности игрока, что исход часто тоже представляется непредсказуемым и в высшей степени неопределенным. Таким же воспринимается и исход состязаний в физических видах спорта, но здесь неопределенность зависит от физических способностей участников.

Механики неопределенности – это методы и способы добавления в игру интересных элементов неопределенности. Очень подробно они рассмотрены в работе Костикиана (Costikyan, 2013), поскольку большинство этих механик фундаментальны для многих жанров. Поняв природу источников неопределенности и научившись преодолевать ее, можно достичь большого мастерства во многих видах игр.

Перечислим несколько популярных типов данных механик, а еще больше источников неопределенности будет упомянуто в главе 5 «Геймплей».

Случайность – получение набора случайных данных, например в результате броска физических костей или в результате работы компьютерного ГПСЧ (генератора псевдослучайных чисел). Это виды механик, включающие в себя случайные показатели со специфическим распределением. Например, при броске шестигранной кости каждый раз с равной вероятностью выпадает число из множества 1, 2, 3, 4, 5, 6. Очень часто случайные числа используются во многих элементах геймплея – от детских настольных игр, где нужно бросать кости, чтобы определить количество ходов фишки, до сложных военных игр, где

различные генерируемые случайные числа определяют исход боевых действий. При очередном броске костей предыдущие броски не влияют на вероятность выпадения того или иного числа. Если вероятность остается неизменной для разных случайных результатов, то такие процессы называются *стационарными случайными процессами*. Но стационарность не означает, что вероятности выпадения различных чисел обязательно одинаковы. При бросках одной шестигранной кости наблюдается *равномерное распределение* (вероятности выпадения всех чисел одинаковы), но совсем другая ситуация наблюдается при одновременном броске нескольких костей или при сложении результатов нескольких бросков. Например, дети при игре в «Монополию» интуитивно догадываются, что при броске двух костей чаще выпадает семь очков и реже два или двенадцать, хотя и ничего не знают о центральной предельной теореме.

Тасование – это перемешивание множества физических или виртуальных объектов, например колоды карт. Этот элемент неопределенности довольно распространен как в физических (в первую очередь в карточных), так и в компьютерных играх. Тасуемая колода карт представляет собой генератор случайных результатов, вероятность каждого из которых меняется со временем в зависимости от уже полученных. Например, для только что перетасованной колоды вероятность вынуть из нее пикового туза изначально составляет $1/52$. Но по мере вынимания карт вероятность того, что следующей картой окажется пиковый туз, возрастает. Когда мы уже вынули туза, понятно, что вероятность вынуть эту же карту равна нулю. Поскольку каждый генерируемый показатель меняет вероятность последующего, такой процесс называется *нестационарным*. При этом реализация процесса не обязательно бывает такой же простой, как тасование колоды карт. Например, чтобы предотвратить серию случайных проигрышей, которые могут огорчить игрока, используются «умные» псевдослучайные алгоритмы, следящие за недавними итогами и, соответственно, меняющие вероятность выпадения того или иного результата.

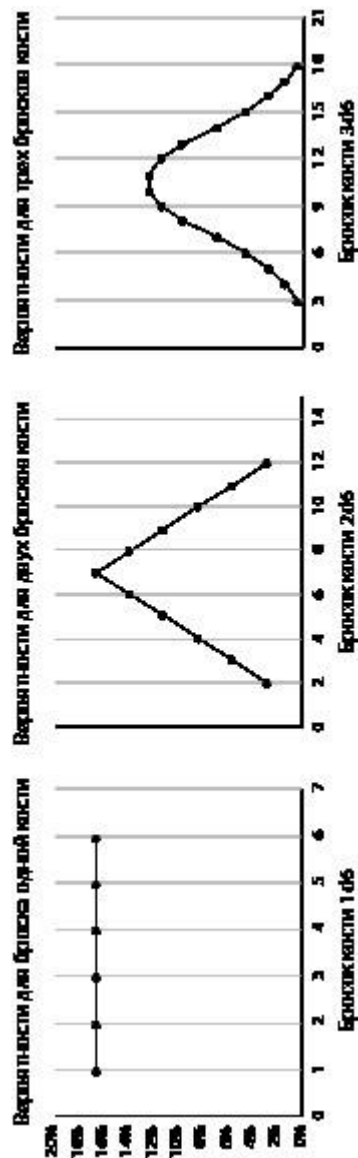


Иллюстрация 3.5. Вероятности выпадения чисел для броска одной шестигранной кости (1d6) и для суммы бросков двух или трех шестигранных костей (2d6, 3d6)

Попытки понять, как устроен случайный генератор и как использовать его для своих целей, – один из основных видов удовольствия от игр. В случае стационарных процессов задача игрока состоит в том, чтобы понять принципы распределения вероятностей различных случайных процессов. Например, какие вероятные результаты получаются в итоге, равномерное ли это распределение (вроде бросков одной кости) или нормальное – вроде броска нескольких костей. И, что самое важное, может ли игрок использовать данную информацию для получения преимущества.

Нестационарные процессы еще хитрее. Игрок сталкивается со сложной задачей предсказания очередного результата в зависимости распределения вероятностей со временем, а для этого нужно запоминать предыдущие результаты. Но и такие попытки могут оказаться весьма полезны. Когда опытные игроки обучаются «подсчитывать карты», они получают огромное преимущество над менее опытными соперниками.

Механики управления ресурсами

Последнее из распространенных семейств механики относится к имеющимся в распоряжении игрока ресурсам, с помощью которых можно одержать победу в игре и которые можно улучшать или менять на другие ради получения каких-то преимуществ.

Например, игрок имеет возможность управлять «юнитами» – военными отрядами, транспортными средствами, космическими кораблями или отдельными людьми – как в ролевых играх. Игрок может обладать такими ресурсами, как деньги или сырье, с помощью которых покупает другие элементы, производит объекты и т. д. Он даже может обладать своими собственными фабриками, городами или целыми планетами, что зависит исключительно от игры.

Мы будем использовать термин «ресурсы» для обозначения объектов, которые можем приобретать и с помощью которых имеем возможность достигать своих целей в игре. Вместе с правилами, объясняющими, как следует обращаться с такими ресурсами, они составляют *механику управления ресурсами*. Рассмотрим, какие здесь могут наблюдаться варианты.



Иллюстрация 3.6. Отношения между разными элементами управления ресурсами

Юниты. Это объекты игрового мира, которыми управляет или с которыми сталкивается игрок. Юниты, обладающие определенными свойствами, – распространенный элемент механики ресурсов. В качестве их примеров можно привести как фигуры в *шахматах* или *шашках*, так и объекты недвижимости в *«Монополии»*. Что касается компьютерных игр, то уже в ранних образцах (таких как *Space Invaders*) существовали разнообразные юниты с различными свойствами.

Механики юнитов отвечают на следующие вопросы.

- Какие в игре есть юниты и как их контролирует игрок или ИИ (искусственный интеллект)?

- Как они действуют? Каковы их способности?

- Как эти способности можно изменить или улучшить в стратегическом смысле?

Каждая игра обладает своим уникальным дизайном юнитов. При этом в отдельных жанрах у них наблюдаются некоторые общие свойства.

- В *классических военных играх* юниты соответствуют реальным боевым единицам (войскам, отрядам, транспортным средствам) со схожими параметрами передвижения и боевыми характеристиками, такими как дальность стрельбы или скорость передвижения по определенному типу местности.

- В *стратегиях в реальном времени*, таких как *StarCraft*, имеются разные типы юнитов, способности которых сочетаются между собой так, что игрокам необходимо объединять их в команды, чтобы они поддерживали друг друга. По мере прогресса игры благодаря «глобальному дереву исследований» могут становиться доступными новые юниты.

- В *ролевых играх* юнитов обычно меньше, но они сложнее. Это обитающие в игровом мире персонажи, со своей историей, своими характеристиками и разнообразными способностями, которые могут приобретаться по ходу игры.

- В *экономических стратегиях* («играх-менеджерах») имеются юниты, производящие и потребляющие различные ресурсы или их специфические сочетания. Они могут быть даже неподвижными (как здания в *SimCity* или города в *Civilization*). Игрокам приходится рассчитывать, какие выгоды они получают и какие понесут затраты в случае обмена тех или иных ресурсов с учетом своих способностей и будущих угроз.

- В некоторых играх бывают довольно неожиданные юниты. Например, гаражи с автомобилями в *Forza*, образующие группы с различными способностями.

Выбор юнитов и их характеристики, как правило, очень сильно зависят от тематики и жанра игры, оказывая различное влияние на геймплей.

В различных жанрах со временем сложились разные способы управления юнитами. Поскольку игроку обычно приходится одновременно иметь дело с огромным их множеством, особую роль приобретают вопросы, связанные с искусственным интеллектом, такие как выбор юнитов, их группировка, отдача им команд, наблюдение за состоянием и т. д.

Способности. Мы уже говорили про способности – это типы действий, которые могут осуществлять юниты. В некоторых играх они очень ограничены – например, в *шахках* каждая фигура следует очень простым правилам. Но в других играх бывает целый калейдоскоп юнитов и способностей, например в ролевых играх со сложными магическими системами или в карточных играх со своими свойствами у каждой карты.

Чем больше способностей у юнитов, тем больше пространство действий игрока и тем труднее ему рассчитать оптимальное решение. Сложность зависит не только от количества, но также и от разнообразия юнитов и способов их взаимодействия друг с другом. Например, в *шахматах* несколько фигур с крайне отличающимися способностями дают огромное пространство игровых состояний.

В некоторых играх со временем можно менять или улучшать способности, что используется для составления интересных игровых задач для игрока. Представим, например, что игрок в экшен-стратегии в реальном времени, набрал хорошо сбалансированную армию из наступательных и оборонительных юнитов, юнитов-лекарей и т. д. Потом в игровой сессии наступил момент, когда их можно модернизировать, благодаря чему сила атаки увеличится вдвое, но юниты при этом станут значительно слабее. Перед игроком возникает дилемма. Имеет ли смысл модернизировать все юниты? Как это скажется на структуре армии? Стоит ли добавлять дополнительных лекарей или защитников и доступно ли такое пополнение? Часто единственным способом дать ответ на эти вопросы бывает решение поэкспериментировать и посмотреть, что получится.

Гейм-дизайнеры, помимо прочего, отличаются друг от друга еще и тем, придерживаются они принципов *минимализма* или *максимализма*. Что лучше – иметь несколько строго дифференцированных юнитов и способностей или большое количество юнитов с широким спектром способностей со своими небольшими преимуществами и недостатками? Это очень субъективный вопрос, от ответа на который зависят предлагаемые игроку задачи и варианты игры. С одной стороны, некоторым участникам нравится богатство юнитов и способностей, требующее поистине энциклопедических познаний. С другой стороны, есть игроки, обожающие минимализм и предпочитающие, чтобы в игре было несколько правил, сочетающихся между собой сложными способами. Правильный выбор в большей степени зависит от жанра, от аудитории конкретной игры и от того опыта игрока, к которому стремится гейм-дизайнер.

Параметры (характеристики, «статистики»). При разработке юнитов с различными способностями, особенно если мы хотим, чтобы они менялись со временем, хочется выразить каким-либо образом эти способности в числовом виде – чтобы игроку было легче понимать их, а компьютеру – просчитывать.

Числовые параметры фигурируют в играх достаточно давно. В каждой военной игре

есть такие показатели, как очки атаки, защиты и передвижения, определяющие поведение различных видов транспортных средств и отрядов на поле боя (Dunnigan, 2000). Фэнтезийные ролевые игры вроде *Dungeons & Dragons* унаследовали этот подход вместе с другими элементами военных игр, а персонажи в них имеют такие характеристики, как сила, ловкость, мудрость и т. д. Отголоски данной модели до сих пор видны в ролевых играх и в других жанрах.

Данные параметры часто используются для следующего.

- Настройки вероятности (например, большее число очков атаки означает более высокую вероятность победить в бою).
- Настройки способностей (например, от количества очков передвижения зависит, сколько клеток юнит пройдет за ход; большая дальность означает, что юнит попадет в цель на большем расстоянии).
- Разблокировки способностей (с повышением очков силы у воина появляется возможность снабдить дополнительное оружие новыми способностями).

Изменение способностей юнитов или ввод новых юнитов с новыми способностями меняет способ игры и заставляет игроков разрабатывать новые тактики и новые стратегии победы. Как правило, это хорошо. Такие изменения не дают игрокам цепляться за одну-единственную выигрышную для них стратегию.

Модификаторы . С помощью таких «усилителей» игроки могут улучшать показатели и делать юниты более мощными. Механики, меняющие показатели, обычно называют «модификаторами», «баффами» или «бонусами». Идея состоит в том, чтобы ввести некий внутриигровой ресурс, который можно получать по ходу игры и стратегически использовать его для модификации способностей.

Модификаторы широко распространены в играх всех жанров.

- В *Mario Kart* подбираемые грибы автоматически увеличивают скорость, а банановая кожа меняет траекторию автомобиля – оба модификатора действуют непродолжительное время.
- В *Diablo* магические виды оружия или одежды увеличивают различные показатели персонажа, но имеется много ограничений по поводу того, кто может использовать тот или иной вид снаряжения.
- В *Civilization* различные технологии навсегда увеличивают продукцию ферм, шахт и других типов обрабатываемых клеток, но они дороги для приобретения.

Наконец, даже если в игре и есть элементы для улучшения, то их вовсе необязательно модифицировать. В таких играх, как *Space Invaders* и *шахматы* , есть юниты с разными способностями, но они не меняются со временем. Сложность и увлекательность игры достигается за счет сочетания способностей между собой.

Баффы . Улучшение показателей в пользу игрока часто называется «баффом», а уменьшение показателей называется «нерфом». Как бафф, так и нерф преследуют одну цель – изменить способности игрока, но они обладают разным эмоциональным эффектом. Игроки предпочитают, чтобы их юниты улучшались и наращивали мощь. Это ощущается как награда за тяжелый труд, а нерф воспринимается как наказание за какой-то проступок. По этой причине в целом лучше, чтобы в игре было больше объектов и событий, которые усиливают показатели, а не «нерфят» их.

Валюта и объекты . Выше мы говорили о юнитах, их способностях и показателях, но существуют и другие виды ресурсов.

• *Деньги и другая валюта* , с помощью которых можно покупать все что угодно; получать же деньги и валюту можно либо продавая ресурсы, либо в качестве награды за прогресс в игре.

• *Объекты* , которые можно использовать, такие как оружие в военных играх или одежда и книги заклинаний в ролевых; объекты можно хранить в инвентаре или надевать на своего персонажа.

• *Расходные материалы* («расходники») – такие как топливо для танков или еда для солдат, потребляемые в процессе игры, а также используемые для усиления или разблокировки других элементов.

• *Сырье* – материалы для создания (ремесла, или «крафта»), которые не потребляются непосредственно, а используются для создания других юнитов (например, сталь для танков) или объектов (например, железо и уголь, из которых делается сталь).

• *Пространство* само по себе тоже может быть ресурсом, как в игре *Risk* , где занимаемая игроком местность со временем дает определенное количество сырья.

Предполагается, что эти ресурсы нужны, чтобы их получать, обмениваться и потреблять. В идеальном случае они взаимодействуют друг с другом и с другими механиками игры, создавая внутриигровую экономику. К экономике, валюте и «крафту» мы вернемся в следующей главе, когда будем беседовать об игровых системах.

За пределами четырех семейств

В данной главе мы беседовали о четырех популярных семействах механик: контроля, прогресса, неопределенности и управления ресурсами. Они используются в самых разнообразных играх и жанрах, что делает их идеальными для первоначального знакомства.

Но всего типов механик существует гораздо больше. Механики автомобильных гонок отличаются как от механик пошаговых стратегий, так и от механик шутеров от первого лица.

Если вы хотите познакомиться со специализированными механиками, то лучше обратиться к работам, сосредотачивающимся на отдельных жанрах. Некоторые из них указаны в разделе «Дополнительная литература».

Дизайн механик

При создании новой игры мы редко разрабатываем совершенно новую, не виданную ранее механику. Чаще всего пересматриваем уже существующие и сочетаем их между собой новым способом, что уже может быть довольно трудной задачей. Это дизайнерская проблема, зависящая от контекста, но существуют *эвристические принципы* , относящиеся к общему улучшению механик в различных жанрах игр.

Эвристика дизайна

В своей книге «Эстетика игры» Аптон предлагает шесть эвристических принципов оценки и улучшения игровых механик (*The Aesthetics of Play* , Upton, 2015)⁵. Вот они.

Выбор. *Игрок должен осознавать ряд возможных действий.* Игроки сразу же реагируют на доступное им пространство действий. Если оно слишком узкое или ограниченное, то им становится скучно выполнять одно и то же. Но если игра предлагает

⁵ Хотя Аптон рассуждает о них в терминах действия и горизонта намерений, мы адаптируем его высказывания к тем определениям, с которыми к этому моменту познакомились в данном тексте.

слишком большой выбор, то участники могут запутаться и разочароваться в ней. Очень важно подобрать идеальное пространство действий, а оно зависит от аудитории, типа игры, ожиданий игроков и т. д. (сравните, например, ожидания от детской игры «Змеи и лестницы» и от игры *Magic the Gathering*). На практике стоит руководствоваться правилом, согласно которому игры с большим количеством вариантов часто кажутся более богатыми и открытыми по сравнению с играми, в которых мало вариантов выбора.

Разнообразие. *Действия не повторяются.* Действия должны быть многочисленными, но интереснее, когда они со временем меняются и игроку не приходится раз за разом выбирать их из одних и тех же вариантов. Когда вдруг наши любимые действия становятся недоступными или открываются дополнительные возможности, когда мы получаем целый набор новых способностей, которых не было раньше, то это неизменно привлекает внимание игрока. Такой поворот выбивает игрока из привычной колеи и заставляет его оценивать новую ситуацию. Это особенно важно, потому что игроки активно учатся использовать существующие механики, но если все остается неизменным, то люди быстро освоятся, им будет неинтересно играть дальше и появится ощущение того, что участники «наигрались». Смена механик со временем – хороший способ заставлять игрока активно учиться.

Последствия. *Действия имеют последствия.* Решение делать или не делать что-то должно иметь последствия, которые игрок мог бы приписать своему решению. Если действия игрока не имеют последствий (например, происходит то, что обязательно должно было произойти), игрок теряет чувство агентивности (самостоятельности) или чувство контроля над ситуацией. А если последствия действий игрока нельзя приписать ему самому (например, когда результат происходит гораздо позднее действий), то это приводит к непониманию того, как устроена игровая вселенная. Представления о последствиях и о причинно-следственной связи в игровом мире необходимы для разработки долгосрочных планов и стратегий. Игроку интересно узнавать, какие действия приводят к наилучшим последствиям, рассматривать разнообразные возможности, последствия которых конфликтуют между собой (например, выбор между краткосрочной и долгосрочной выгодой).

Предсказуемость. *Исходы должны быть ожидаемыми.* При условии что у действий есть последствия, у игрока должна быть возможность хотя бы в какой-то степени усвоить, как они соответствуют друг другу, чего стоит ожидать. Умение создавать теории, как работает игровой мир, лежит в основе умения строить долгосрочные планы. Например, если механика боя работает недостаточно предсказуемо, игроку будет трудно разобраться в ней, он ощутит разочарование. Как полная предсказуемость, так и полная непредсказуемость нежелательны – самые интересные механики лежат где-то посередине. У игрока должна быть возможность при достаточном усердии узнать, как работают разные механики, а затем воспользоваться ими.

Неопределенность. *Исходы не предрешены заранее.* Это обратная сторона предсказуемости. Полностью предсказуемый результат представляет собой проблему, как и полностью непредсказуемый. Программа выигрывает от некоторой неопределенности, которой игрок может попытаться воспользоваться. Один из распространенных источников неопределенности – генераторы случайных событий, например броски костей или раскладывание карт. Даже в полностью детерминированных играх, таких как *шахматы*, наличие другого игрока придает каждому игровому моменту большую неопределенность.

Удовлетворение. *Желанные исходы достижимы.* Акцент здесь делается на вероятности и работе, необходимых для достижения желаемых результатов, таких как прохождение игры или победа. Если в игре совсем нельзя победить или игрок постоянно проигрывает и у него ничего не получается, он расстраивается и теряет интерес. Частый источник такого разочарования – слишком высокая сложность. Но если игрок все время получает задуманное и побеждает, то такая игра тоже не приносит особого интереса. Нам требуется выверенный баланс между интересным вызовом и возможностью достичь цели.

Эти эвристические правила полезны потому, что указывают на некоторые общие принципы, делающие механики интересными и увлекательными для игроков. Их следует держать в памяти, придумывая новые элементы и действия для своей игры.

Первичные и производные механики

Разрабатывая дизайн элементов и действий игры, нужно также учитывать тот факт, что осваивающие игру игроки на их основе будут порождать свои абстракции. Например, обучаясь играть в *шахматы*, мы знакомимся с тем, что такое фигуры, действия, правила игры. Учимся ходить ферзем или ладьей, узнаем, что пешка ходит на одну или две клетки по вертикали, а атакует по диагонали и т. д.

Но чтобы играть в шахматы *хорошо*, необходимо выработать иной набор абстракций. Нам нужно будет усвоить концепции защиты короля, контролирования доски и принесения фигуры в жертву. Придется узнать, что такое «вилка», «связка», эндшпиль и овладеть энциклопедическими знаниями о стандартных дебютах. И этот принцип вовсе не ограничивается шахматами – у состояющихся в *Hearthstone* игроков своя терминология, позволяющая им рассуждать о колодах и их типах; игроки в многопользовательские онлайн-игры много говорят о тактике сражений с различными мобами и т. д.

Здесь мы наблюдаем различие между первичными и производными механиками. *Первичные механики* – это то, что требуется для игры. *Производные механики* – это абстракции, созданные и получившие определения в сообществе игроков, обычно имеющих большой опыт в данной игре и подметивших какие-либо дополнительные закономерности. В некоторых играх они могут называться *стандартными тактиками* или *механиками второго порядка*⁶.

Такого рода экспертная ментальная модель не ограничена лишь играми. Во всех сферах знаний эксперты обычно находят и выделяют какие-либо закономерности. Производные механики для игры не требуются, но они очень полезны, поскольку отражают знания экспертов о структуре игры помимо самых очевидных.

Мы как гейм-дизайнеры не обязаны придумывать производные механики или предсказывать их появление. Можно даже не знать о них до момента открытия сообществами игроков. Тем не менее, когда мы разрабатываем и тестируем игру, надо быть настороже и выявлять механики, делающие игру слишком легкой или не приносящей особого удовольствия по каким-то иным причинам.

Краткий итог

В данной главе мы исследовали начало дизайнерского процесса. Ключевые положения ее таковы.

- *Механики* – это базовые строительные блоки игр. Это идиоматические комбинации более мелких компонентов, таких как игровые объекты, действия игроков и правила, определяющие их взаимодействие. Существует множество различных типов механик, но в этом тексте мы описали только четыре семейства.

- *Механики контроля*, относящиеся к контролю над аватаром игрока в игровом мире.
- *Механики прогресса*, относящиеся к пониманию процесса игры и к контролированию

⁶ Формулировка механик «первого порядка» и «второго порядка» используется, в частности, в книге «Характеристики игр» (*Characteristics of Games*, Elias, Garfield and Gutschera, 2012) и в статье «Формальные абстрактные инструменты дизайна» (*Formal Abstract Design Tools*, Doug Church, 1999). Но такое «числовое» определение порождает вопросы, на которые мы не знаем, как ответить. Например, если есть механики первого и второго порядка, то какими могут быть механики третьего порядка? Мы не будем развивать эту тему, а просто поговорим только о первичных и других известных механиках.

прогресса в ней.

- *Механики неопределенности*, относящиеся к попыткам понять и предсказать неопределенные события в будущем.

- *Механики управления ресурсами*, относящиеся к тому, как определенным образом управлять элементами игры как ресурсами.

- Иногда игры описываются в терминах «существительных и глаголов», то есть для рассуждения о механиках используется языковая метафора.

- Игру можно воспринимать как *чередующуюся последовательность состояний*, а в результате взаимодействия игрока с игрой ее состояние меняется. Данный прогресс можно описывать как траекторию по возможному *пространству состояний*. Точно так же множество доступных игроку действий можно представить в виде *пространства действий*, а восприятие этого пространства со стороны игрока играет в данном случае огромную роль.

- При разработке специфических механик наших игр мы можем воспользоваться *эвристическими правилами* Аптона, помогающими оценивать взаимодействие механик с игроком и друг с другом.

- Стоит ожидать, что игроки могут открывать новые способы использования созданных нами объектов, действий и правил, расширяя множество задуманных механик иным и непредсказуемым способом.

В главе 4 «Системы» мы рассмотрим, как механики, в свою очередь, образуют более крупные системы, после чего в последующих главах обратим внимание на геймплей.

Дополнительная литература

История механик

Концепция *игровой механики* и *динамики* относится по меньшей мере к эпохе холодной войны, когда некоторые исследователи стали воспринимать игры как инструмент исследований и познания, а не просто как развлечение – например, для симуляции военных конфликтов или преподавания таких сложных систем, как бизнес или поведение популяций. В материалах того времени уже использовались знакомые нам термины, пусть и не совсем в том же смысле: «механиками» назывались сочетания игровых правил и дополнительных элементов (Greenlaw, Herron, and Rawdon, 1962, 149–150), а «динамикой» – поведение игры и ее разворачивание во времени (Abt et al., 1965, 43–44; Adams, 1970).

В сфере развлечений концепцию *механик* первыми приняли на вооружение энтузиасты различных батальонных игр, поскольку этот термин уже встречается в дизайне военных игр начала 1970-х. Затем с распространением видеоигр как рыночного феномена он все чаще начинает использоваться в речи как разработчиков видеоигр, так и освещающих их деятельность журналистов⁷. К середине 1990-х «механика» стала очень распространенным термином, что видно на примере выступлений на Конференции разработчиков игр или из статей в *Game Developer Magazine* и разборов игр в популярной прессе.

Возможно, именно из-за того, что это понятие имеет давнюю историю и используется в различных сферах, у него нет единственного и четкого определения. Почти всегда подчеркивается важность правил – например, Брайтуэйт и Шрайбер (Brathwaite and Schreiber, 2009, 28) описывают механику как «еще один термин для того, что другие обычно

⁷ В сфере военных игр термин появляется в игре *The Avalon Hill* еще в 1971 году (Avalon Hill Inc., 1971, 3). В сфере видеоигр термин «механики» мимоходом упоминается в печатных руководствах игр *Archon* (Westfall, 1983) и *Batalyx* (Llamasoft, 1985), из чего можно сделать вывод, что он уже использовался в гейм-дизайне. Что касается прессы, то, по-видимому, впервые данный термин начал использоваться в журнале *Electronic Games* в начале 1980-х при описании геймплея различных аркадных игр.

назвали бы “правилом”». Но, помимо этого, различные сообщества подчеркивают разные аспекты механик (обзор определений см. в «Определение игровых механик», *Defining Game Mechanics*, Sicart, 2008), а гейм-дизайнеры, имеющие разные подготовку и образование, могут иметь совершенно не похожие мнения по данному вопросу. К счастью, небольшие семантические различия не оказывают значительного отрицательного воздействия на практику дизайна.

В нашем тексте термин «механики» базируется на его современном использовании именно в дизайне видеоигр и весьма соответствует определению Селлера (Sellers, 2017, 101), определяющему их как «семантически дееспособные (то есть осмысленные) комбинации токенов и правил».

Классификации

Также было предпринято несколько попыток создания стандартных таксономий (классификаций) механик, но на практике в индустрии они пока не используются.

- «*Шаблоны в гейм-дизайне*» (*Patterns in Game Design*, Bjork and Holopainen, 2004) – исследование закономерностей в стиле «программных шаблонов», среди которых указан и ряд игровых механик. К сожалению, эта книга больше не продается.

- «Проект 400» (*The 400 Project*) Хола Барвуда и Ноа Фалстейна – заброшенный проект по документированию 400 правил гейм-дизайна, который может представлять исторический интерес (Barwood and Falstein, дата не указана).

Подробные исследования

Многие дизайнеры рассуждали о специфических типах или семействах механик со своей перспективы дизайнерского пространства. Среди отличных источников такого рода можно назвать следующие.

- «Эстетика игры» (*The Aesthetic of Play*, Upton, 2015), особенно главы 3 и 4, где обсуждается пространство действий игрока, различие между актуальным и воспринимаемым пространством и эвристические правила их дизайна.

- «Ощущение игры» (*Game Feel*, Swink, 2009) содержит подробное исследование экшен-механик, включая анализ тайминга ввода/вывода и вариаций в возможностях различных видов устройств ввода.

- «Неопределенность в играх» (*Uncertainty in Games*, Costikyan, 2013) предлагает тщательное исследование типов неопределенности и их использования на примерах многих существующих игр.

- «Характеристики игр» (*Characteristics of Games*, Elias, Garfield and Gutschera, 2012) – весьма рекомендуемое исследование многопользовательских механик, включая разнообразие точек принятия решений для дизайнеров, особенно в настольных и карточных играх. Связанный с этой книгой «Справочник Кобольда по дизайну настольных игр» (*The Kobold Guide to Board Game Design*, Selinker, 2011) объединяет различных дизайнеров настольных игр, делящихся своими мнениями о нескольких специфических механиках «настолок».

- «Продвинутый гейм-дизайн» (*Advanced Game Design*, Sellers, 2017) погружает читателя в тему юнитов, ресурсов, валют и настройки экономик и систем, в которых используются эти объекты.

- «Руководство по военным играм» (*Wargames Handbook*, Dunnigan, 2000) – великолепный ресурс по истории механик юнитов и ресурсов в контексте настольных военных игр.

Кроме того, имеется хороший сайт *GDC Vault*, где практикующие дизайнеры обсуждают свои работы, представленные на ежегодной *Конференции разработчиков игр*. Их

обсуждения послужат отличным источником подробных дизайнерских знаний в широком разнообразии игр и жанров. Слайды и видео можно найти по адресу <http://gdcvault.com>.

Упражнения для индивидуального выполнения

3.1 Механики

Выберите хорошо известную вам настольную или видеоигру. Опишите три разные механики данной игры. Для каждой механики укажите используемые в ней объекты и действия, а также правила, определяющие работу этой конкретной механики.

3.2. Пространство состояний и пространство действий

Выберите хорошо известную вам настольную игру.

а. Подумайте, какая информация полностью описывает ее состояние в отдельный момент времени (например, позиции всех игровых объектов и т. д). Какое это пространство состояний – полностью или частично наблюдаемое?

б. Подумайте, какие действия может выполнять игрок. Меняются ли со временем, в процессе движения от начала к концу игры, количество и типы действий? Как это влияет на ощущение от игры?

3.3. Явная и неявная механика

Выберите нравящуюся вам компьютерную игру. Перечислите несколько неявных механик, которыми игрок овладевает со временем или на примере других игроков. Представьте, что могло быть, если бы игрок знал об этих механиках с самого начала? Как поменялось бы его восприятие (опыт) игры?

3.4. Механики контроля

Выберите нравящуюся вам компьютерную игру, в которую играют с помощью консольного контроллера. Перечислите различные выполняемые игроком действия и укажите, с какими кнопками или джойстиком контроллера они связаны.

Как вы думаете, почему дизайнеры назначили именно такое управление вводом? Что в этих кнопках делает их подходящими именно для данных конкретных действий? Соответствует ли расположение этих средств управления привычным схемам для данного жанра?

3.5. Механики прогресса

Выберите нравящуюся вам игру. Какого рода механики прогресса предоставляют игроку обратную связь о его действиях в игре? Насколько они успешно возвращают игрока в игру и удерживают его интерес?

3.6. Управление ресурсами

Выберите знакомую вам игру. Опишите по меньшей мере три разных типа юнитов в ней и укажите их различия. Играют ли какую-то роль в разнице этих юнитов их параметры или способности?

Упражнения для выполнения в группе

ГЗ.1. Дизайнерское задание: изменить механику в *шашках*

В этом упражнении мы рассмотрим, как изменение механики шашек влияет на опыт игрока.

а. Сыграйте несколько партий в шашки, чтобы вспомнить их геймплей. Затем выберите

какую-нибудь механику в игре и измените ее интересным и необычным способом.

- Можно добавлять или модифицировать правила относительно существующих игровых объектов и действий. Например, как насчет того, чтобы в начале каждой игры у игроков уже было по «дамке»?

- Можно добавить новые объекты и регулирующие их новые правила. Например, изобрести фигуру с поведением, отличающимся от простых шашек и «дамок».

- Можно также удалить некоторые механики, хотя шашки и так достаточно минималистичны в данном отношении.

б. Обсудите, как по вашему мнению, эти изменения повлияют на геймплей и игровой опыт.

в. Сыграйте несколько партий с новыми механиками и проанализируйте свои впечатления. Обращайте особое внимание на следующее.

- Скорость. Стала ли игра быстрее или медленнее? Если да, то почему.
- Вызов. Играть стало труднее или легче? Почему?
- Увлечательность. Игра стала более или менее увлекательной? Почему?

И для завершения упражнения выполните следующие задания.

г. Во всех подробностях опишите новые механики, включая действия, объекты и затронутые правила.

д. Опишите, как перемены в механике повлияли на ваши впечатления от игры; как, по вашему мнению, они *должны были* повлиять на игровой опыт и как *на самом деле* повлияли. Опишите свои ощущения от измененной и от оригинальной игр.

Г3.2. Дизайнерское задание: добавление посторонних механик в шашки

В этом упражнении мы рассмотрим, что будет, если в *шашки* добавить несколько неожиданных механик.

а. Сыграйте несколько партий в шашки, чтобы вспомнить их геймплей. Затем добавьте какую-нибудь механику, не имеющую никакого отношения к шашкам, ориентируясь на описанные в этой главе общие типы.

- *Механика прогресса* – вроде подсчета очков и т. д.
- *Механика неопределенности* – вроде броска костей, выкладывания случайных карт и т. д.
- *Механика управления ресурсами* – вроде добавления показателей или способностей.

Можно добавить неопределенности, поставив условием взятия фигуры бросок кости. Можно добавить очки атаки и здоровья, забирая фигуры только после того, как очки их здоровья опустятся до нуля и т. д.

б. Обсудите, как эти изменения повлияют на геймплей и игровой опыт.

в. Сыграйте несколько партий с новыми механиками. Проанализируйте свои впечатления, обратив особое внимание на то, насколько более или менее увлекательным кажется данный вариант игры по сравнению с оригинальным.

Для завершения упражнения выполните следующее.

а. Во всех подробностях опишите новые механики, включая действия, объекты и затронутые правила.

б. Опишите, *как* перемены в механике повлияли на ваши впечатления от игры. Каковы были ваши ожидания, что вы наблюдали на самом деле.

в. Проанализируйте, *почему*, по вашему мнению, эти изменения механик так изменили

опыт. Что именно в данных изменениях произвело такой эффект?

4 Системы

В предыдущей главе мы говорили о механиках и перечислили несколько их типов с примерами. Теперь поговорим о том, как они работают совместно друг с другом и как можно производить их анализ в виде динамических систем, охватывающих разнообразие механик и их взаимодействий.

Во-первых, рассмотрим, каким образом механики вступают друг с другом в цепочки отношений (вроде «производителей» и «потребителей») и как можно анализировать их. Во-вторых, разберем, что происходит, когда цепочки начинают взаимодействовать друг с другом и сами с собой, получая обратную связь. Наконец, проанализируем, как подобные динамические системы можно составлять при создании игры. Начнем с примера.

Мотивирующий пример: Diablo

Diablo III – это экшен-RPG в стиле так называемого данжен-кроулера. Игра разворачивается в опасном фэнтезийном мире, где от игрока требуется исследовать многочисленные загадочные подземелья, развалины и тайные области, заодно разбираясь с ордами населяющих их чудовищ и вражеских созданий.

Геймплей ориентирован на повторяющийся шаблон «исследуй, сражайся, возвращайся». Мы управляем пестрой командой искателей приключений, покидающих лагерь и устремляющихся в очередное подземелье в поисках чудовищ и спрятанных сокровищ. Команда должна зачистить каждую зону, подбирая «падающие» с чудовищ и хранящиеся в сундуках вещи. Со временем инвентарь искателей приключений заполняется, поэтому они возвращаются в лагерь, чтобы продать добычу и восстановить силы, пополнить здоровье, создать необходимые зелья, починить оружие или обучиться новым заклинаниям. Затем приходит пора вновь отправляться навстречу приключениям.



Иллюстрация 4.1. Скриншот игры *Diablo III*

В этом упрощенном примере показан ряд пересекающихся между собой механик. Опишем их как составные части нескольких систем высокого уровня.

Инвентарь

- В результате сражений мы получаем предметы (оружие, сокровище, ремесленные ресурсы и т. д).

- Оружием можно экипировать персонажей команды.
- Другие предметы можно класть в инвентарь или оставлять на месте.
- Объем инвентаря ограничен и изначально мал, но его можно увеличить.
- В лагере предметы из инвентаря можно продать или переместить из него в хранилище.

Экономика

- В результате сражений мы получаем предметы (оружие, сокровище, ремесленные ресурсы и т. д).
- В лагере предметы можно продать и получить золото.
- Золото можно потратить различными способами.
 - а. На покупку разных предметов (оружие, зелья, заклинания и т. д).
 - б. На усиление параметров или предметов (например, на увеличение объема инвентаря).
 - в. На услуги (обучение новому заклинанию, починку оружия и т. д).

Крафтинг

- В результате сражений мы получаем предметы (оружие, сокровище, ресурсы и т. д).
- Ресурсы можно использовать для крафта новых видов предметов.
- Некоторые предметы можно «разобрать» обратно на ресурсы.
- Для создания предмета необходим рецепт, в котором указано, какие ресурсы и в какой пропорции требуются.
 - Рецепты для создания сами по себе являются предметами, которые можно покупать, продавать, находить и т. д.

Это лишь небольшая иллюстрация трех разных *игровых систем* : инвентаря, экономики и крафтинга. Мы пока что не рассматриваем сражение, исследование и некоторые другие системы, присутствующие в играх, сосредотачиваясь на этих трех для простоты рассуждения.

Один из любопытных аспектов данных систем заключается в том, что они *пересекаются* , деля между собой различные механики (такие как объекты или валюту), но используют их для разных целей. Именно это пересечение систем делает игру интереснее.

Игровые системы

Систему можно рассматривать как *совокупность механик, предназначенных для совместной работы* . Работа системы зависит от сочетания используемых элементов и способа их совмещения.

Выше были приведены три примера систем.

- Инвентарь: экипировка предметов, владение ими, ограничение на хранение вещей.
- Экономика: обмен ресурсами, предметами и валютой.
- Крафтинг: превращение некоторых ресурсов и предметов в иные элементы.

Системы состоят из механик. Чтобы взаимодействовать, они должны *делиться между собой механики* , но использовать их для разных целей. В нашем примере система инвентаря взаимодействует с системой экономики с помощью предметов (их можно переносить в инвентаре, покупать и продавать), а также с системой крафта посредством ресурсов, которые также можно переносить, покупать и продавать. Общие механики создают сеть взаимоотношений и компромиссов между системами и побуждают игрока принимать интересные решения.

Из-за такого пересечения игрок, встречающий новый предмет, должен оценить его в контексте разных систем и компромиссов между ними. Например, полезен ли данный

предмет для крафта? Можно ли продать его позже и стоит ли он того, чтобы таскать с собой? Достаточно ли места в моем инвентаре? Достаточно ли у меня силы, чтобы носить все предметы? Механика с широким охватом, вроде механики предметов или валют, может взаимодействовать со многими системами, включая, например, систему боя.

Сеттинг и системы

Системы – это очень полезные абстракции. Разработку дизайна новой игры можно начать с механик. Но иногда лучше стартовать обсуждение с уровня общих игровых систем, потому что это позволяет «покрыть холст более широкими мазками», то есть начать с более высокого уровня, прежде чем переходить к деталям. Некоторые гейм-дизайнеры (вроде Селлерса) идут дальше и утверждают, что уровень систем – это самый важный уровень абстракции для игр, потому что именно взаимодействие между системами делает данные программы уникальными по сравнению с другими видами развлечений.

Часто хорошей отправной точкой при начале работы над игрой бывает выбор «сеттинга и систем». В каком сеттинге разворачивается игра? Каковы системы, с которыми будет взаимодействовать игрок? Ответы на эти вопросы помогут задать дизайну определенное направление и упростят процесс. Допустим, мы остановились на «фэнтезийном сеттинге со сражениями, исследованием, оружием и случайными подземельями» – и игра идет в направлении *Diablo*. Или же решили сосредоточиться на «исторической игре с экономикой, завоеваниями и управлением страной», что направляет нас в сторону стратегии вроде *Civilization*. Это, конечно, не все детали, но данных уже вполне достаточно для начала работы и для наполнения игры специфическими механиками.

Программная (кодовая) система. Слово «система» часто используется в другом контексте, а именно для описания отдельных программных модулей, таких как система «физики» или «графики». Они не являются совокупностью механик, хотя на них может основываться механика «гравитационной пушки». Имеются ли в виду программные или геймплейные системы, обычно бывает ясно из контекста.

Уровни систем

Игра обычно состоит из множества систем, взаимодействующих между собой – в этом ее можно уподобить часовому механизму со сцепленными шестеренками. В качестве примера приведем неполный список систем стратегической игры вроде *Civilization V*.

- География: случайная карта с многочисленными ресурсами, на потреблении которых основаны другие системы.
- Территория: клетки, которыми владеют игроки, производящие ресурсы для своих владельцев.
- Города: неподвижные юниты, порождающие другие юниты и собирающие ресурсы с окружающих клеток.
- Здания: более мелкие юниты внутри городов, производящие ресурсы или валюту, конвертирующие их и т. д.
- Армии и сражения: подвижные юниты, исследующие территорию, нападающие на вражеские элементы, улучшающие ресурсы на клетках и т. д.
- Соседние цивилизации: искусственный интеллект, состязающийся с игроком.
- Торговля и дипломатия: обмен соглашениями или ресурсами с соседями, вступление в войны и выход из них.
- Экономика: использование валюты для покупки ресурсов или усовершенствования юнитов, клеток и т. д.
- Наука: продвижение по дереву технологий, благодаря чему разблокируются новые

юниты и здания, появляются вновь открытые ресурсы на клетках и т. д.

- Культура и золотой век: система для разблокировки крупномасштабных «усилений».
- Религия: пассивная генерация ресурсов, требующих большой базы обращенных городов.
- Путь к победе: несколько цепочек условий, выполнение которых дает первому выполнившему их игроку победу в игре.

Этот список не претендует на полноту, а лишь дает представление о том, насколько разными могут быть системы внутри одной игры.

В перечне также не указано, как системы взаимодействуют между собой, но взаимодействие между ними, конечно же, наблюдается. Примером общей механики может быть механика таких природных ресурсов, как пшеница или уголь. На клетках карты случайным образом появляются природные ресурсы (география), а если мы устанавливаем контроль над данными клетками (территория), то можем добывать эти ресурсы или конвертировать их в другие (посредством зданий). Другие ресурсы, в свою очередь, обеспечивают рост населения (в городах) и позволяют создавать армии, чтобы нападать на врагов (сражения).

Правильное разделение игры на уровни систем и нахождение точек взаимодействия между ними – это трудная, специфическая для каждой конкретной игры задача, часто ее начинают с небольших набросков и создания правильных базовых систем, на основе которых можно будет продолжать процесс разработки. Иногда такую задачу выполняют описательным образом, «на бумаге», но иногда в процессе дизайна бывает полезно создавать простые *системные прототипы*, чтобы понять, как эти системы будут вести себя в реальной обстановке, после того как ими начнут пользоваться игроки.

Систематическое мышление

Игровые системы – это вид абстракции, игнорирующий детали отдельных механик, когда они не нужны, но позволяющий сосредоточиться на том, каким образом вся игра составляется из более крупных частей.

Концепция систем широко распространена и в других сферах дизайна – в науке, инженерном деле. Например, в дизайне автомобилей можно рассуждать о трансмиссии или о двигателе так, как если бы они были отдельными взаимодействующими между собой единицами, а только потом начать продумывать необходимые для их реализации детали. В биологии и медицине можно размышлять о системах различных типов органов и тканей, работающих согласованно (например, таких как нервная система), спускаясь на более низкий уровень лишь при необходимости.

В гейм-дизайне системы предоставляют три специфических преимущества.

1. *Изоляция*. Легче рассуждать о том, что какие-либо механики работают совместно, если отвлечься от всего внешнего (как при описании циклов обратной связи).

2. *Повторное использование*. Существующие системы решают некоторые конкретные дизайнерские проблемы. Иногда ими можно воспользоваться для решения наших вопросов или в качестве источника вдохновения при разработке игр. Например, игра с удачной системой крафта может послужить источником вдохновения для схожего дизайна в других играх.

3. *Масштабирование*. Такие элементы геймплея, как темп или прогрессия игры, легче обсуждать на примере больших строительных блоков (перейдем к ним позже в этой главе).

Практикующий гейм-дизайнер должен иметь представление о разнообразных существующих системах и уметь привести множество примеров их использования. Столкнувшись с какой-либо дизайнерской проблемой, он сможет воспользоваться своим

опытом в других играх с разными типами систем и различными их реализациями (со своими преимуществами и недостатками), найдя благодаря этому хорошее решение.

Цепочки и циклы механик

Часто обнаруживается, что механики образуют *цепочки взаимодействий*, формирующих *циклы*, такие как циклы конверсии или обратной связи. Цепочки и циклы – это распространенные, фундаментальные схемы дизайна, которые могут существовать в отдельной системе или охватывать сразу несколько.

Разберем подробнее, как это происходит. Продолжим анализировать экшен-RPG *Diablo III* и уделим особое внимание системам *экономики* и *инвентаря*. Разберем, как механики добычи ресурсов и конверсии образуют цикл, а также обсудим некоторые способы анализа поведений данных циклов.

Цепочки конверсии

Рассмотрим упомянутую нами экшен-RPG. Мы находим зону с троллями, имеющими при себе много всяких собираемых предметов («лута» или «добычи»), поэтому посылаем команду искателей приключений зачистить данную локацию. Предположим, что это происходит по следующему сценарию.

1. Команда побеждает как можно большее количество троллей и собирает добычу. В какой-то момент инвентарь героев заполняется, после чего они возвращаются в лагерь. Участники потеряли 60 очков здоровья и использовали 100 стрел, поэтому им нужно восстановить здоровье и пополнить припасы.

2. Герои продают 10 полученных от троллей предметов за 10 золотых монет каждый и получают +100 золота.

3. Участники пополняют снаряжение на –10 золота и находят целителя, который восстанавливает очки здоровья за –60 золота, поэтому для того, чтобы снова отправиться за приключениями, на восстановление потрачено –70 золота.

Отметим, что всего прибыль экспедиции составила +30 золотых монет, поэтому игроки остались в выигрыше.

На иллюстрации 4.2 приведена диаграмма данной последовательности.

Это разновидность диаграммы ресурсов, показывающей, как они добываются и потребляются с помощью различных доступных для игрока действий. Диаграмма довольно грубая, поскольку в ней опущены подробности того, сколько всего ресурсов можно конвертировать, при каких обстоятельствах и т. д. Прямоугольниками показаны потребляемые и производимые ресурсы, а кругами и овалами – действия, в результате которых ресурсы потребляются и производятся.

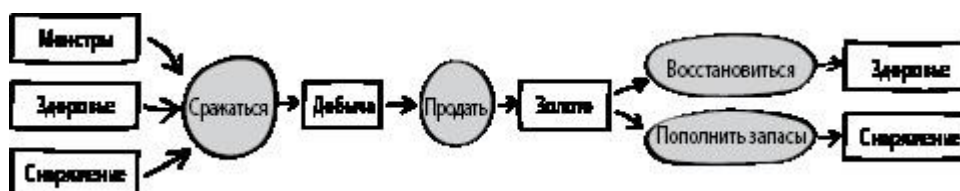


Иллюстрация 4.2. Пример диаграммы ресурсов

Механики экономики (продажа, исцеление, пополнение запасов) и сражения (добычи предметов) вместе составляют *цепь конверсии* производства и потребления. Каждый шаг дает нечто, что служит основой для последующего. В данном примере каждое сражение стоит каких-то очков здоровья и в результате из игрового мира удаляется один монстр, но он

приносит добычу. Затем эту добычу можно продать за золото, а за золото восстановить здоровье и снаряжение, которые будут потрачены на очередное сражение.

Другими словами, данная диаграмма иллюстрирует *конверсию ресурсов*, происходящую в ходе охоты на троллей.

- *Сражение* конвертирует троллей, здоровье и снаряжение в добычу.
- *Продажа* конвертирует добычу в золото.
- *Исцеление* конвертирует золото в здоровье.
- *Пополнение запасов* конвертирует золото в снаряжение.

На этой диаграмме мы не указали *настраиваемые значения*, то есть точное количество потребляемых или производимых ресурсов, но упомянули их в описании и отметили, что цепочка прибыльна. Каждый шаг требует определенных усилий, но в процессе мы получаем доход (+100 золота, снаряжение, здоровье, добыча), несем некоторые расходы (-70 золота, снаряжение, здоровье, добыча) и заканчиваем цепочку с прибылью в +30 золота.

Расчет курса обмена

Можно также включить в схему настраиваемые значения и рассмотреть ресурсы с точки зрения их *курсов обмена* – сколько ресурса должно тратиться по сравнению с другими.

- *Продажа* : 1 предмет добычи → 10 золота.
- *Пополнение запасов* : 10 золота → 100 стрел (или 1 золото → 10 стрел), следовательно, 1 предмет добычи → 100 стрел.
- *Исцеление* : 60 золота → 60 очков здоровья (или 1 золото → 1 очко здоровья).
- *Сражение* : как описано выше, в среднем 10 предметов добывается с 10 монстров, но при этом тратится 60 очков здоровья и 100 стрел, следовательно, 1 монстр → 1 добываемый предмет, -6 очков здоровья, -10 стрел.
- Все вместе: сражение, затем продажа, пополнение запасов и исцеление.
1 монстр → 1 предмет добычи, -6 очков здоровья, -10 стрел
→ 1 монстр → 10 золота, -6 очков здоровья, -10 стрел
→ 1 монстр → 9 золота, -6 очков здоровья
→ 1 монстр → 3 золота.

Последняя строка анализа конверсии ресурсов содержит довольно любопытный вывод: по сути, монстры – это источник золота. Если мы находим его и готовы потратить некоторое количество золота (и усилий!), чтобы сразиться и разобраться с последствиями, то получим еще больше золота. Если повторять это раз за разом, то каждый монстр будет приносить нам в среднем +3 золотые монеты, не считая усилий или сопутствующих затрат.

Циклы конверсии

Теперь представим, что мы повторяем эти действия. Пройгнорировав затрачиваемые на сражения и странствия усилия и сосредоточившись только на добываемых ресурсах и затратах, получим повторяющуюся цепочку в виде цикла, показанного на иллюстрации 4.3.



Иллюстрация 4.3. Диаграмма ресурсов из иллюстрации 4.2, показанная в виде цикла

Цепочка повторяется и образует *цикл*, потому что результаты каждого шага в итоге повторяются и служат основой для последующих аналогичных шагов в будущем. Описанный нами цикл всегда производит одинаковое количество ресурсов, но это не обязательно.

Отметим, что данный цикл *прибыльный*. Каждый раз, проходя через него, мы теряем 70 золота, но получаем 100, так что наши доходы выше расходов. Следовательно, пройдя эту цепочку не один, а несколько раз, можно получить больше денег. Легко догадаться, что будет, если отправляться на охоту постоянно. Количество золота в нашем инвентаре со временем будет изменяться примерно так, как показано на иллюстрации 4.4.

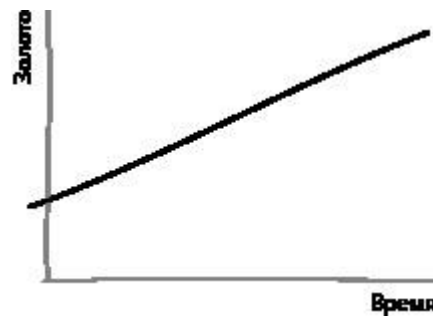


Иллюстрация 4.4. Количество золота растет с постоянной скоростью

Получаем симпатичный линейный график. Поскольку каждая итерация дает больше, чем мы тратим, то со временем можно накопить немало золота.

Предположив, что тролли возрождаются бесконечно, мы получим *бесконечный источник* золота. Но это плохо. Как только в игре появляется легкий источник бесконечной прибыли, игроки начинают недобросовестно пользоваться им и разрушать тщательно настроенные системы обмена и прогрессии, на которых основаны многие игры. Имея много золота можно получить дорогие и очень мощные виды оружия и брони, из-за которых игра станет совсем легкой и перестанет доставлять удовольствие.

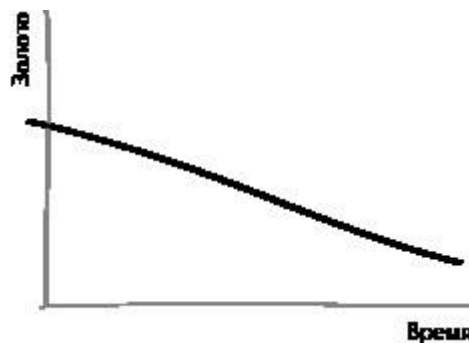


Иллюстрация 4.5. Количество золота уменьшается с постоянной скоростью

Исправить ситуацию можно несколькими способами. Во-первых, поменять

настраиваемые значения, то есть коэффициенты стоимости разных вещей. Например, что будет, если удвоить стоимость зелий исцеления и стрел на одного тролля с -7 золота до -14 ? Тогда график богатства игрока со временем будет выглядеть примерно так, как на иллюстрации 4.5.

Цикл превратился в *неприбыльный*, потому что игрок теряет золото с каждым прохождением по нему. Если он продолжит повторять свои действия, то со временем обанкротится. Это нежелательно, а игроки могут сказать, что разработчики «поломали» цикл.

Поэтому нужно рассмотреть какие-то другие варианты. Игрок не должен бесконечно получать или тратить ресурсы, ведь если предполагается, что игрок будет добывать их благодаря циклу производства, то необходимо предусмотреть механизм его ограничения.

Вот несколько идей по поводу того, как это сделать.

1. Ввести некий лимит – как для игрока (предусмотреть механику повреждения оружия, ломающегося через некоторое количество атак), так и для добываемых ресурсов (ограничить количество троллей). Это жесткий ограничитель цикла, включающийся через некоторое количество итераций и заставляющий игрока отвлекаться от цикличности. Не очень хорошее решение, но оно может установить реальные ограничения на эксплуатацию ресурсов.

2. Сделать исход менее предсказуемым (чтобы игрок мог возвращаться в лагерь с пустыми руками). Это добавляет неопределенности и потенциально случайным образом увеличивает расходы. Но опытные игроки все равно смогут добиться ожидаемой награды, повысив среднюю результативность своих попыток.

3. Увеличить затраты на сражение или путешествие (чтобы повторение цикла не было таким привлекательным). Это увеличит траты на прохождение цикла, но затраты – это не ресурсы, а нечто внешнее – например, потраченное на действие время. Однообразные действия во время прохождения цикла, отнимающие много времени, на игровом жаргоне называются «гриндом». К нему многие игроки относятся с неприязнью, но если цикл достаточно прибылен, то они, несмотря на жалобы, продолжают этим заниматься.

4. Масштабировать кривую прибыли со временем так, чтобы цикл превращался из прибыльного в неприбыльный по мере повторения его игроком.

Последний пункт особенно интересен. Это пример так называемой *динамической настройки*. Допустим, мы сделаем, чтобы затраты на прохождение цикла изменялись динамически, а стоимость снаряжения и зелий исцеления со временем повышалась. В начале снаряжение и здоровье будут стоить нам -70 золота за один поход, но затем станут увеличиваться на дополнительные -10 золота за каждую вылазку – при условии, что за продажу мы получаем прежнее количество денег. В итоге график количества имеющегося у игрока золота будет выглядеть так, как на иллюстрации 4.6.

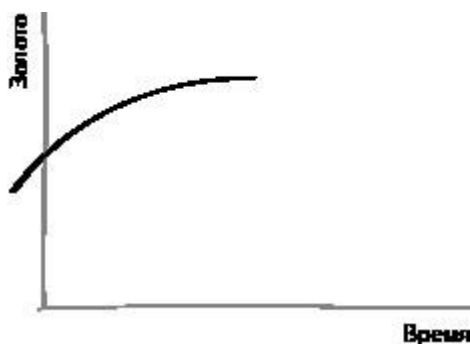


Иллюстрация 4.6. Скорость увеличения количества добываемого золота уменьшается

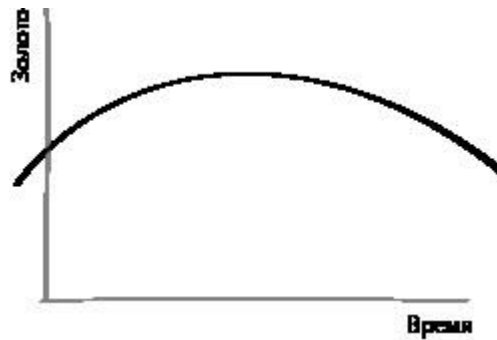


Иллюстрация 4.7. Скорость увеличения количества добываемого золота уменьшается, а затем начинает уменьшаться само количество добываемого золота

Если цикл начинается со значения в +30 прибыльности за поход, то дополнительная настройка, о которой мы говорим, со временем уменьшает его прибыльность. С каждым разом игрок будет получать все меньше и меньше. Через какое-то время цикл превратится из прибыльного в неприбыльный, поэтому если игрок продолжит его выполнять, то начнет терять деньги, как показано на иллюстрации 4.7.

Такие настройки препятствуют «гринду». Игрокам будет выгоднее сократить расходы до того, как они слишком вырастут, и перейти в другую зону игры с иными заданиями и циклами.

На практике динамическая настройка прекрасно работает, если игрок сам по своему желанию может принимать решение, продолжать ли цикл или заканчивать его. Для игрока гораздо интереснее, если он может выбирать – продолжать ли привычное занятие или сокращать расходы и переходить к чему-то новому, возможно, более прибыльному или увлекательному.

Циклы обратной связи

До сих пор мы рассуждали о циклах, поведение которых стабильно и относительно статично. Каждый раз игрок вносит какой-либо вклад, а завершая цикл, получает определенный исход (результат) – например, охотится на троллей и получает добычу за каждого убитого из них. Но многие циклы не такие статичные.

Система обратной связи – это иной вариант системы, в которой прошлое поведение определяет будущее. Обычно это означает, что имеется некое *внутреннее состояние*, определяющее результат работы системы, а затем данный результат служит одним из вводимых в систему элементов, определяющих ее состояние в будущем.

Типичный пример в автомобилях – это система автоматического регулирования скорости («круиз-контроль»), поддерживающая *установленное значение* скорости. Во время передвижения система проверяет текущее значение скорости, и, если оно слишком низкое, автомобиль разгоняется. Это, в свою очередь, приводит к увеличению скорости, но в какой-то момент она увеличивается настолько, что система отключает ускорение, в результате чего автомобиль начинает замедлять ход. Так состояние системы и результат обратной связи взаимно влияют друг на друга с течением времени.

Системы обратной связи чрезвычайно широко распространены как в естественных, так и в искусственных процессах – от систем контроля численности популяции до управляемых человеком. Распространены они и в играх, где часто используются для увеличения или уменьшения различий со временем.

Положительная обратная связь

Цикл положительной обратной связи – это цикл, в котором разница между текущим состоянием и заданным значением обрабатывается и складывается так, чтобы разница между

состояниями со временем увеличивалась. Из реальной жизни можно привести экономический пример со *сложными процентами*. Предположим, у нас имеется банковский счет с определенной суммой на нем, по которому каждый месяц выплачиваются проценты. Затем проценты откладываются на этот же счет, поэтому сумма процентов за второй месяц рассчитывается уже от первоначального вклада и дополнительной суммы, полученной в первом месяце. Третья выплата рассчитывается от первоначального вклада и дополнительной суммы, полученной за два месяца.

На иллюстрации 4.8 показана схема такого цикла.

При условии что процентная ставка положительна и со временем не меняется, то построив график изменения со временем количества денег, мы получим нечто вроде кривой, изображенной на иллюстрации 4.9.

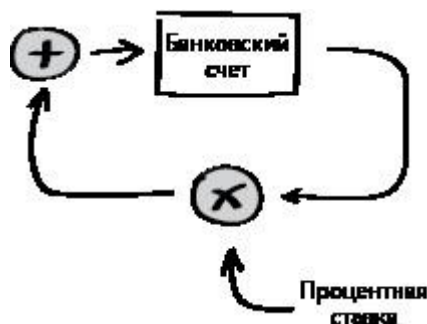


Иллюстрация 4.8. Схема увеличения суммы в банке благодаря выплатам по процентам

Разумеется, это прибыльный цикл, но с каждым прохождением он становится *все более и более прибыльным*! Это цикл положительной обратной связи, потому что она поддерживает то, что система уже делает. С каждой итерацией разница между текущим и заданным начальным значением умножается на процентную ставку и прибавляется к банковскому счету, с каждым разом увеличивая *разницу*⁸.

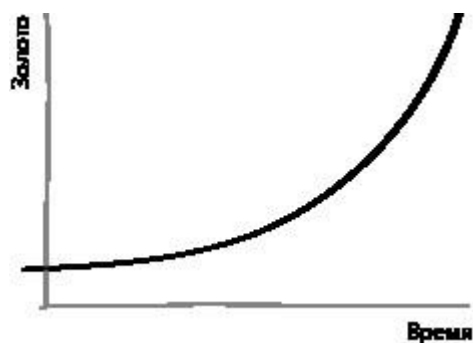


Иллюстрация 4.9. Сумма на банковском счете увеличивается с растущей скоростью

Такой вид обратной связи значительно укрепляет и усиливает начальное состояние. Шаг с умножением обуславливает «взлетающий» экспоненциальный рост.

Циклы положительной связи широко распространены в играх. Любимый пример цикла долгосрочной обратной связи любого гейм-дизайнера – «*Монополия*». Игрок, у которого больше денег, чем у других, может покупать или модернизировать больше участков с недвижимостью, благодаря чему получает больше ренты, что еще больше увеличивает его шанс получить больше денег. С другой стороны, игрок, у которого мало денег, имеет меньше

⁸ Иногда такие циклы называются «*дивергентными*», или «*расходящимися*», потому что с каждой итерацией текущие и установленные значения расходятся все сильнее. Селлерс (Sellers, 2017, 61) также использует термин «*подкрепляющие циклы*», потому что они усиливают и увеличивают начальные преимущества.

шансов получить ренту и меньше шансов приобрести дополнительные клетки.

Расходящиеся циклы используются в стратегиях реального времени и в стратегических играх. В игре *StarCraft* игроки добывают минералы, которые тратят на покупку юнитов и других объектов. Но один из этих юнитов – комбайн (harvester), позволяющий собирать еще больше минералов за единицу времени. К счастью, данный цикл положительной обратной связи (получить минералы, создать комбайны, получить еще больше минералов, создать еще больше комбайнов и т. д.) ограничивается жестким лимитом на количество добываемых ресурсов, а также другими важными системами, их потребляющими.

Схожие эффекты наблюдаются и в однопользовательских играх. Например, в таких играх на фермерскую тематику, как *Stardew Valley* или *FarmVille*, имеется очень интуитивный цикл положительной обратной связи. Чем больше растений высаживает игрок, тем более обильный урожай он получает, тем больше семян или денег получает для следующей посадки и т. д.

Отрицательная обратная связь

Цикл отрицательной обратной связи минимизирует разницу между текущим и требуемым состоянием. Один из типичных примеров – уже упомянутый «круиз-контроль». Другой стереотипный пример – термостат, подключенный к системе отопления и охлаждения помещения (Wiener, 1961, 96).

Предположим, что мы установили на термостате некую нужную нам температуру, которая периодически проверяется. Если в комнате слишком холодно, текущая температура оказывается ниже установленного значения, поэтому включается обогреватель, повышающий ее и сокращающий разницу, как показано на иллюстрации 4.10.

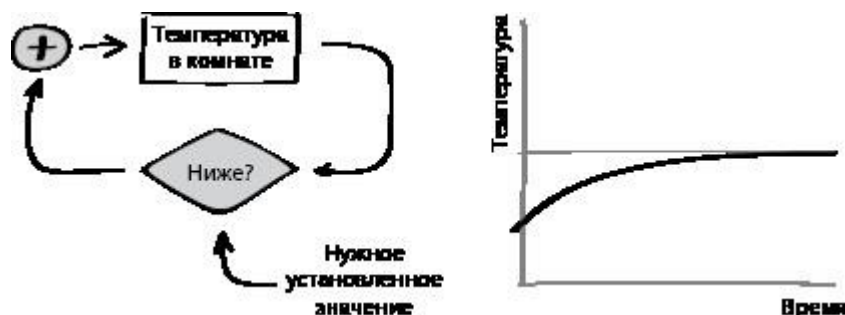


Иллюстрация 4.10. Схема обогрева и соответствующий температурный график

Аналогично, если в комнате слишком тепло, текущая температура оказывается выше установленного значения, поэтому включается охлаждающий элемент, чтобы понизить температуру и тем самым сократить разницу.

Циклы отрицательной обратной связи называются так, потому что обратная связь препятствует тому, что делает система. Разница между установленным значением и текущим состоянием приводит к действиям, сокращающим эту разницу в будущем. В таких системах обратная связь заставляет состояния системы сходиться к установленному значению⁹.

Отрицательная обратная связь менее распространена в играх, но все же существует. Один из примеров – динамическое изменение параметров игры в зависимости от действий игрока, благодаря чему легкая игра становится труднее, а трудные испытания – легче.

В гоночных играх иногда используется так называемый эффект *резиновой ленты* – если игрок-человек сильно отстает от других машин, компьютерные игроки замедляются, чтобы не уехать слишком далеко, но если игрок-человек их обгоняет, то компьютерные

⁹ По этой причине они иногда называются «конвергентными» («сходящимися») циклами или, как у Селлера (Sellers, 2017, 61), «балансирующими».

участники ускоряются, чтобы человеку было труднее победить. Похожего эффекта в гонках можно достичь благодаря изменению качества модификаторов, усиливающих или ослабляющих способности игрока. Например, в игре *Super Monkey Ball* проигрывающие участники по сравнению с другими чаще встречают усилители скорости (Salen and Zimmerman, 2004, 221).

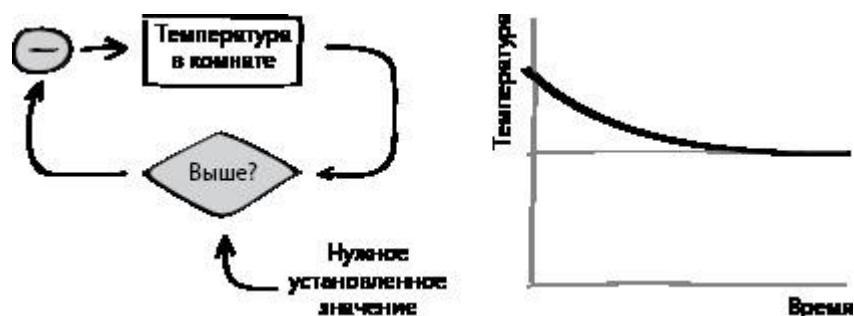


Иллюстрация 4.11. Схема охлаждения и соответствующий температурный график

Подобным образом некоторые ролевые игры следят, когда участник заходит в новую область, автоматически подстраивая показатели врагов в зависимости от уровня и показателей игрока. Это делается, чтобы человек получил задуманный дизайнером опыт независимо от того, «прокачал» ли он предварительно своего персонажа или не старался «прокачиваться» по ходу игры, придя в данную область неподготовленным.

Такие процессы называются простыми методами динамической настройки сложности. Возможны и дополнительные методы, такие как изменение вероятности выпадения добычи или ее качества, а также изменение распределения врагов или материалов на уровне в зависимости от того, насколько показатели игрока отличаются от ожидаемого состояния.

Эффекты положительной обратной связи

Циклы положительной обратной связи оказывают положительное влияние на игровой опыт. Всегда приятно видеть, как увеличивается твоя армия или растет банковский счет. Также они находят свое отражение во многих известных нам естественных и искусственных системах, таких как быстрый рост популяции в экосистеме или тенденция к захвату победителем всего неконтролируемого рынка¹⁰.

Вместе с тем циклы обратной связи имеют тенденцию к усилению ранних преимуществ. В однопользовательских играх такой эффект может быть весьма положительным. Игрок, инвестировавший на ранней стадии в расширение производства, позже будет вознагражден увеличением своего состояния. Чтобы такое развитие не превратилось в скучный взлетающий рост, гейм-дизайнеры обычно добавляют в игру некоторые противовесы. Игроку придется дважды подумать, стоит ли ему тратить ресурсы на подпитку такого цикла или потратить их на что-то еще.

Если воспользоваться примером из *StarCraft*, то на раннем этапе игры перед участником встает вопрос, как потратить имеющиеся минералы, представляющие собой валюту для создания различных юнитов и постройки зданий. Доступен ряд вариантов, но один из них – это комбайн (harvester), позволяющий добывать больше минералов за единицу времени.

¹⁰ Есть целый поджанр «ленивых игр», включая такие, как *Cookie Clicker* и *Adventure Capitalist*, базирующихся исключительно на постоянно усиливающихся циклах положительной обратной связи. В этих ленивых играх игроки тратят деньги или ресурсы на постройку юнитов, все быстрее и быстрее производящих еще больше денег или ресурсов, но в них бывает на удивление приятно раздувать циклы производства до невероятно абсурдных пропорций (Pescorella, 2016).

Если игрок потратит свое первоначальное количество минералов на постройку одного комбайна и больше ничего создавать не будет, то его бюджет будет выглядеть примерно так, как на графике на иллюстрации 4.12. Предположим, что скорость добычи минералов постоянна.

Один комбайн добывает минералы со скоростью x , тем самым производя ресурсы для будущих покупок. Если же игрок решит потратить эти ресурсы на второй комбайн, то ему придется дольше ждать последующего поступления минералов, хотя два комбайна будут добывать их со скоростью $2x$. Этот эффект отображен на иллюстрации 4.13.



Иллюстрация 4.12. Количество минеральных ресурсов в *StarCraft* до и после покупки комбайна



Иллюстрация 4.13. Количество минеральных ресурсов в *StarCraft* с учетом покупки двух комбайнов

Это весьма интересное стратегическое решение для игрока, встающего перед выбором: «съесть яйцо» или «вырастить курицу» – то есть потратить дополнительные ресурсы прямо сейчас или немного «поголодать», но купить второй комбайн, приносящий впоследствии больше ресурсов. И как сопоставить ценность такой покупки и ценность вооруженных юнитов, с помощью которых можно нападать или обороняться? Необходимость делать выбор между несколькими циклами обратной связи приводит к увлекательным стратегическим дилеммам.

Однако в многопользовательских играх (или в играх с ИИ-игроками) положительная обратная связь может приводить к нежелательной ситуации. Если участники начнут всю пользоваться циклами обратной связи, замкнутыми на самих себя, то игрок, раньше всех получивший какое-то преимущество, пусть и небольшое, в итоге обгонит остальных. Если разница между ресурсами игроков станет настолько большой, что другие игроки уже не смогут догнать первого и сократить разрыв, то это разрушительным образом скажется на честности игры и на общем духе состязания.

Представим, например, ранний этап военной игры, где нужно с помощью войск захватывать территории, позволяющие создавать новые войска, которые будут завоевывать новые территории и так далее. Допустим, имеются два игрока, и Игрок 1 получает в начале процесса некоторое преимущество. Поскольку численность войск и количество территорий

образуют цикл положительной обратной связи, то общее количество ресурсов может изменяться согласно графику, показанному на иллюстрации 4.14.

Оба показателя растут по «дивергентному» принципу, но поскольку у Игрока 1 в начале было преимущество, то так же быстро увеличивается и *разрыв* между игроками. В какой-то момент они могут разойтись настолько, что Игрок 2 уже никогда не догонит Игрока 1. С того момента как Игрок 2 поймет, что его позиция невыигрышная, процесс станет приносить ему гораздо меньше удовольствия.

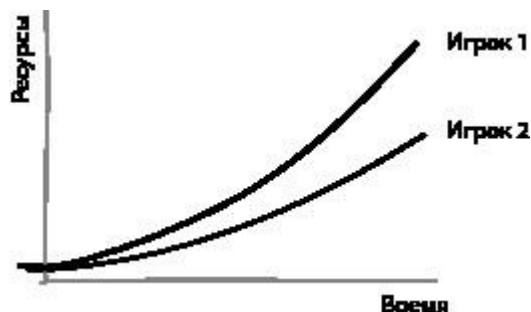


Иллюстрация 4.14. Траектории ресурсов, расходящиеся со временем в результате небольшой разницы в начальных условиях

Пример того, как может разворачиваться игра, показан на иллюстрации 4.15

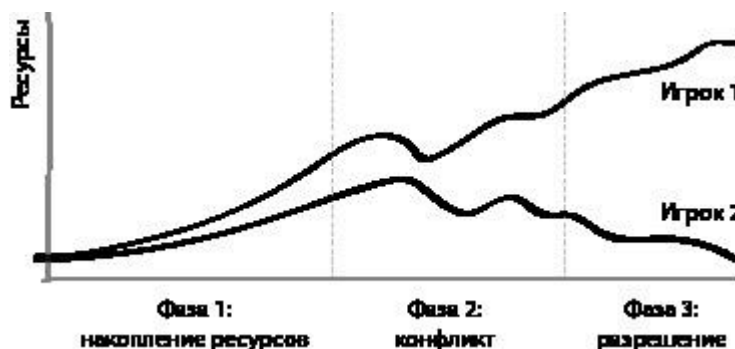


Иллюстрация 4.15. Изначальное расхождение траекторий может повлиять на поздние этапы игры

Когда разница между игроками станет слишком велика, а Игрок 1 осознает свое преимущество, он тут же постарается вовлечь аутсайдера в конфликт (боевой, гонку вооружений и т. д.), в котором Игроку 2 придется очень туго. Затем в какой-то момент лидер станет очевиден, а игра перейдет в *фазу разрешения* с эскалацией конфликта, где все игроки тратят свои ресурсы на определение победителя. Финал будет совершенно не радостным для участников, понимающих, что у них нет ни единого шанса выиграть.

Особенно славится быстрым увеличением шансов на победу «Монополия», где бывает очень длинная и скучная фаза разрешения, в процессе которой лидер постепенно опустошает банковские счета других игроков, приводя их к банкротству. Это своего рода медленная война на истощение, где стороны понемногу откусывают друг у друга ресурсы, а победителем, скорее всего, окажется сторона с наибольшим количеством запасов.

«Война на истощение» – распространенная тема жалоб среди более опытных игроков. Исход игры становится понятным на ранних стадиях, за которыми следует долгое разрешение. Развитие игры «Монополия» может выглядеть подобно графику, показанному на иллюстрации 4.16.

На фоне этого более увлекательными кажутся игры с более интересными фазами накопления ресурсов и конфликта, когда больше времени тратится на состязание, а игроки пытаются достичь победного состояния; фаза разрешения в таких играх короче. Стратегии в реальном времени, такие как *StarCraft*, часто ориентированы на фазу конфликта с

неопределенной победой, а у опытного игрока имеется много способов выбраться из, казалось бы, проигрышного состояния. Развитие таких игр похоже на график, показанный на иллюстрации 4.17.

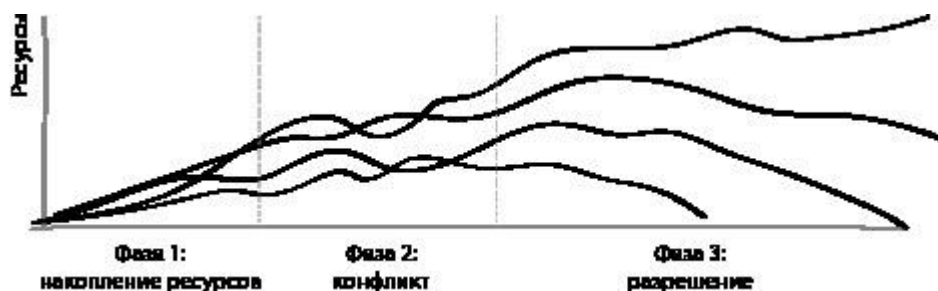


Иллюстрация 4.16. Пример игры с медленно расходящимися траекториями ресурсов и затянутой фазой разрешения

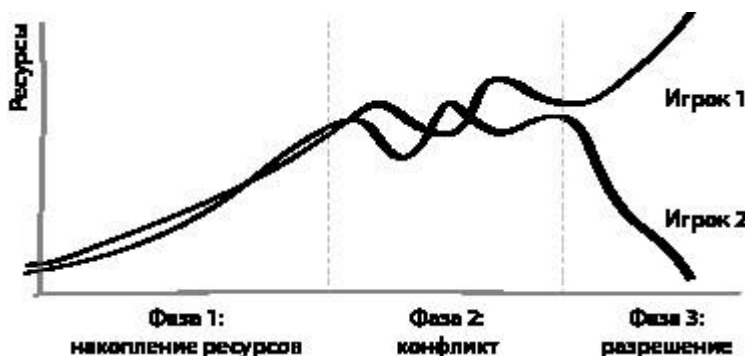


Иллюстрация 4.17. Пример игры с близкими траекториями ресурсов и короткой фазой разрешения

Эффекты отрицательной обратной связи

Циклы отрицательной, или «балансирующей», обратной связи можно использовать для подстраивания под способности игрока или под требуемый стиль игры, но это может иметь отрицательные последствия.

В однопользовательских играх, таких как описанная нами выше, можно автоматически настраивать уровень врагов в зависимости от текущего уровня игрока и его показателей. Например, постоянная сложность будет поддерживаться независимо от предыдущих удач или неудач игрока. Но в то же время это сделает удачи игрока *менее значимыми*. Наградой за достижения должно быть уменьшение сложности, а если совершенствование навыков для участника означает, что мир просто становится сложнее, то игрок задается вопросом, зачем ему вообще стараться.

В многопользовательских играх последствия немного иные. Например, в гонках с эффектом «резиновой ленты» отрицательная обратная связь принуждает игроков держаться ближе друг к другу, несмотря на то, насколько хорошо или плохо они действуют в игре. По сути, их ранние действия значат гораздо меньше, чем финишные маневры перед группой гонщиков на последнем круге. В этом случае значимыми становятся только последние моменты игры и появляется ощущение, что программа не настолько уж сильно награждает игроков, старающихся показать свои умения на протяжении всего времени.

Игры, в большой степени полагающиеся на случайность (например, карточные, детские настольные и некоторые компьютерные), тоже часто страдают от подобного эффекта. Если элементы случайности слишком велики, возникает тот же эффект «резиновой ленты» – все предыдущие достижения участников легко сбрасываются со счетов, а их усилия на ранних стадиях становятся бессмысленными.

В идеале игры должны сочетать положительную и отрицательную обратную связь,

награждать за усилия на ранней стадии игры, делая их значимыми и полезными, но также не допускать слишком раннего определения исхода. К сожалению, это легче сказать, чем сделать, а на практике близкое к идеальному положение, как правило, достигается за счет множества мелких настроек, многократных исправлений и постоянных тестов.

Эмерджентность ¹¹ и хаос

Отдельные системы сложны уже сами по себе, но еще более трудным становится анализ сочетаний нескольких систем. Подобно тем деревьям в поговорке, за которыми не видно леса, мы, наблюдая за поведением какого-то одного компонента, можем не замечать некоторых свойств и закономерностей, проявляющихся только на более высоком уровне при взаимодействии нескольких сущностей. В данном разделе мы рассмотрим два таких феномена системного анализа: эмерджентное и хаотическое поведение.

Эмерджентное поведение

Постаравшись выйти на более высокий уровень и рассматривая совокупность сущностей как единый объект, мы можем наблюдать поведение (динамику), являющееся следствием поведения составных элементов, но не предопределяемое ими. Приведем несколько природных примеров.

- Косяки рыб и стаи птиц можно моделировать как единую группу, перемещающуюся в пространстве и избегающую препятствий, даже если такое поведение является следствием нескольких очень простых правил, которым следует каждый член группы, таких как правила сохранения скорости, недопущения столкновения с соседями и стремление к центру группы (Reynolds, дата не указана).

- Муравейники и ульи с пчелами представляют собой в высшей степени организованные структуры, образованные насекомыми и следующие простым локальным правилам, которые можно описать без необходимости понимания общей структуры.

- На крайнем конце спектра можно привести пример организмов-эукариотов – таких как мы, представляющих собой совокупность организованных сущностей, состоящих из бесчисленных отдельных клеток, следующих правилам и объединяющихся в ткани, органы и системы с определенными свойствами.

Поведение считается эмерджентным, если его нелегко предсказать на основе поведения составляющих элементов, но все же при этом оно демонстрирует сохраняющиеся со временем свойства ¹². Поскольку эмерджентное поведение предсказать сложно, то его трудно придумать и спроектировать намеренно, а легче открыть в процессе экспериментов с разными реализациями систем.

Эмерджентность – противоречивое свойство. В зависимости от цели (создания определенного опыта для игрока) попытки включить в дизайн эмерджентность требуют различных экспериментов, наблюдений, настроек и проверок того, насколько поведение системы оправдывает ожидания. Из-за этого она мало подходит для игр, требующих большей определенности и предсказуемости игрового опыта, таких как боевики с линейным

¹¹ Данный термин рассматривается в контексте исследования игровых систем и не имеет отношения к эмерджентному нарративу, хотя корреляции имеются. – *Прим. науч. ред.*

¹² Селлерс (Sellers, 2017, 78) называет такой вид поведения *метастабильным*, поскольку оно заметно отличается от поведения входящих в состав систем элементов, но все же остается стабильным даже тогда, когда поведение отдельных составляющих динамически меняется. Например, отдельные птицы летают по разным траекториям, но все же кажется, что вся стая ведет себя как единое целое.

сюжетом.

В играх-симуляторах часть получаемого игроком удовольствия связана с комплексом сложных систем с непредсказуемыми эмерджентными свойствами и с попытками взять их под контроль и управлять ими или просто наблюдать за их развитием. Этот факт иллюстрирует одна байка про игру *Dwarf Fortress*, в которой игроки строят подземные поселения гномов и управляют ими. В игре присутствует множество систем с эмерджентными свойствами и хорошо известен один из примеров такого взаимодействия. На определенном этапе обнаружилось, что кошки в поселении иногда оказываются в состоянии алкогольного опьянения, что не было напрямую предусмотрено в коде, поэтому игрокам стало интересно, почему же так получилось. Оказалось, что к такому неожиданному результату привело взаимодействие нескольких систем. В одной из версий в игру добавили таверны, где гномы, покупающие в них пиво, иногда из-за своей неуклюжести проливают его на пол. В другой системе была предусмотрена система отслеживания по мокрым следам, а система поведения кошек требовала от этих животных поддерживать гигиену и очищать себя. Еще одна система руководила свойствами различных веществ и их влиянием на организм при употреблении. Все это привело к неожиданным результатам. Гномы проливали пиво, кошки ходили по нему, их лапы намакали, они пытались почистить себя, облизывали пиво и напивались (Fenlon, 2016).

В многопользовательских играх также часто встречаются примеры эмерджентности, потому что многочисленные игроки привносят в нее свои ожидания и представления о социальном поведении и структурах. Они пытаются как-то организовать, сотрудничать, состязаться друг с другом, образовывать крупные социальные структуры для достижения общих целей. Такое поведение часто встречается и намеренно поддерживается в крупномасштабных ММО (массовых многопользовательских онлайн-играх). В ММО вроде *EVE Online* предусмотрены очень большие социальные структуры, в которых отдельные игроки работают сообща как корпорации, существуя в экосистеме, поддерживающей эмерджентные единицы разных размеров – от весьма многочисленных команд до «волков-одиночек» (см. блестящие примеры в Lehdonvirta and Castranova, 2014). Рассуждать об эмерджентных свойствах человеческих социальных структур еще труднее, но мы в этом вопросе обладаем прекрасной интуицией и большим жизненным опытом, чего не скажешь об анализе других систем.

Эмерджентное поведение возникает в результате взаимодействия систем или элементов, часто удивляя нас своей непредсказуемостью. Оно может стать как великолепным достижением, так и огромной помехой – в зависимости от намерений гейм-дизайнера, стремления контролировать системы и предполагаемого опыта игрока. Такое поведение порождает геймплей, неожиданно получившийся от взаимодействия с отдельными элементами. При этом попытка проследить за тем, чтобы эмерджентное поведение оставалось в нужных рамках, остается трудной задачей.

Хаотические системы

В играх часто используются многочисленные системы с одновременно действующими и взаимосвязанными циклами обратной связи. Такого рода нагромождение систем часто становится *хаотическим*, накапливающиеся изменения в котором приводят к непредсказуемым результатам. Со временем становится трудно проанализировать и предсказать поведение систем в какой-то определенный момент в будущем. Этот термин позаимствован из *теории хаоса*, занимающейся исследованием динамических систем с такими свойствами.

Известный пример хаотической обратной связи, которая может демонстрировать или не демонстрировать метастабильные свойства – динамика популяций в экосистеме. Например, каждый вид животных в лесу имеет свою динамику популяции. В каждой популяции рождаются и умирают отдельные особи, а численность растет, если рождаемость

превышает смертность, но при этом количество ресурсов (например, пищи и пространства) позволяет поддерживать растущую численность популяции. Но что если пищей для одной популяции служит другая – как в случае лис, охотящихся на кроликов? Этот случай описывается в экологических моделях «хищника и жертвы». Две популяции зависят друг от друга прямо противоположным образом. Популяция хищника растет, когда для них много добычи в виде жертвы, а популяция жертвы растет, когда в экосистеме мало хищников.

В результате образуются два цикла обратной связи, колеблющиеся друг относительно друга. Растущая популяция лис уменьшает популяцию кроликов, но со временем кроликов становится все меньше, а лисы начинают голодать. Это, в свою очередь, позволяет размножиться и восстановиться популяции кроликов, так что эффект выглядит примерно так, как показано на иллюстрации 4.18.

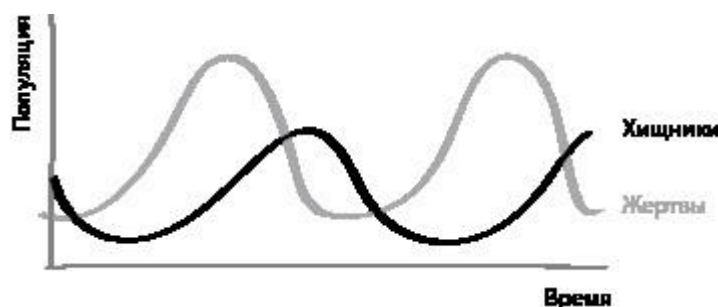


Иллюстрация 4.18. Пример изменения численности популяций хищников и жертв, приводится по Sellers (2017, 64)

Простые хаотические системы иногда возможно анализировать и моделировать с помощью дифференциальных уравнений (в качестве примера можно привести модель взаимодействия хищников и жертв Лотки – Вольтерры). Но более сложные экологические системы не поддаются аналитическим решениям и их приходится симулировать. Реальная экосистема определенного участка местности (например, местного леса или бассейна реки) представляет собой совокупность огромного числа циклов обратной связи, которые в силу своего масштаба демонстрируют хаотические свойства, не поддающиеся анализу.

Хаотическими бывают и человеческие системы. Возьмем для примера экономику, представляющую собой очень сложный комплекс созданных человеком систем обратной связи – обширную сеть многочисленных производителей и потребителей, одновременно сотрудничающих и конкурирующих друг с другом. На микроуровне можно изучать *цепочки поставок*, анализируя реальные циклы обратной связи производства ресурсов и потребления, но такие исследования трудно масштабировать до анализа на макроуровне. Они также часто демонстрируют хаотические свойства. Небольшое изменение начальных условий со временем может усилиться и привести к совершенно различным последствиям, что затрудняет предсказание их поведения.

Приведем пример из игр, показывающий как эмерджентные, так и хаотические свойства систем, а также иллюстрирующий некоторые из трудностей, с которыми сталкиваются разрабатывающие их гейм-дизайнеры. Одна из первых ММО-игр, *Ultima Online*, наряду с несколькими другими постаралась первоначально смоделировать реалистичную экосистему. Новым игрокам нужно было охотиться на кроликов и мелкую дичь для добычи мяса и шкурок, которые они затем продавали, покупая на эти деньги меч и кое-какую броню, после чего уже могли отправиться на поиски приключений. Но охота уменьшала популяцию животных, а ее восстановление со временем замедлялось; уменьшение популяции добычи влияло и на другие виды экосистемы, которые зависели от нее.

Как только к тестированию были привлечены дополнительные игроки, эти циклы привели к катастрофическим последствиям. Новые игроки изначально могли охотиться только на мелкую добычу, поэтому в поисках денег опустошали леса полностью. Следующая

волна игроков не находила себе добычи; они не могли охотиться и вообще заниматься чем-то осмысленным, поэтому игра для них *теряла увлекательность* .

В конце концов динамическую симуляцию системы пришлось удалить и заменить ее на цепочки стабильной добычи ресурсов без циклов. Животные просто возрождались с нужной скоростью, поддерживая численность популяции, а местные торговцы покупали их по фиксированным ценам, чтобы у новых игроков всегда был источник легкого заработка. И хотя такая система не отличалась реалистичностью, но она была предсказуемой и ее можно было настраивать, привнося тем самым для новых участников смысл в игру (Ward, 2003; Garriott, 2017).

Дизайн систем

Теперь, после того как мы рассмотрели различные системы в деталях и в целом, перейдем к вопросу, как же приниматься за их разработку.

Основной подход носит итеративный характер. Нужно раз за разом создавать строительные блоки, собирать их и смотреть, что получилось. Как мы уже говорили в главе 1, на начальной стадии существуют два направления процесса дизайна.

Дизайн снизу вверх : создавать и тестировать одну систему, затем добавлять следующую поверх предыдущей, тестировать их совместную работу и так далее. Таким образом системы со временем разрастаются и накапливаются.

Дизайн сверху вниз : представить для начала геймплей, описать его, а затем подумать, какие системы нужны для его реализации и как они должны взаимодействовать друг с другом.

При разработке игр используются оба метода. Дизайн снизу вверх носит более исследовательский характер и позволяет дизайнеру экспериментировать, нащупывая верный путь. Дизайн сверху вниз – когда поставлена конкретная конечная цель или описан игровой опыт, к которому мы стремимся, более распространен, а нам остается лишь решить, как именно мы должны к ним прийти. Оба подхода можно использовать и совместно – в этом случае применяется анализ сверху вниз для определения нужных систем, но в процессе мы экспериментируем и исследуем, обнаруживая неожиданные направления.

От пользовательских историй до систем

Дизайн сверху вниз может показаться сложным. Вполне возможно, что прекрасные идеи по поводу геймплея будет трудно воплотить в конкретных системах. Для облегчения данного процесса можно воспользоваться итеративным методом для определения списка разных систем, которые нам понадобятся.

Начнем с «пользовательской истории» – сюжетного описания небольшой части геймплея, после чего решим, какие системы или механики подразумевает это повествование. Затем анализируем, какие системы или механики влекут за собой первые и так далее – итеративно заполняя наш дизайн с каждым проходом.

Допустим, мы хотим создать игру в жанре tower defense, в которой ряды монстров, или «крипов» (от англ. creep – красться, подползать), идут по определенному маршруту к базе игрока, а игрок расставляет вдоль дороги оборонительные сооружения, отстреливающие этих монстров, чтобы они не успели добраться до базы.

Прежде всего создаем сюжетное описание («нарратив») предполагаемого игрового опыта.

Крипы рождаются волнами и следуют по определенному маршруту к домашней базе игрока. Участник может строить башни, стреляющие в крипов. Игра заканчивается, когда хотя бы один из крипов добирается до базы.

Описание	Термины, которые нужно определить
<u>Крипы</u> рождаются волнами и следуют по <u>определенному маршруту</u> к домашней базе игрока. Участник может <u>строить башни, стреляющие в крипы</u> . Игра заканчивается, когда хотя бы один из крипов доберется до базы.	Крипы Рождение крипов Следовать по маршруту Башни Стрелять Стрельба Конец игры

Можно составить гораздо более подробное описание, но для простого примера этого будет достаточно.

Затем анализируем это описание – слово за словом – с маркером в руке. Какие системы оно подразумевает? Что нужно для его реализации?

В таблице 4.1 в качестве примера перечислены упомянутые в данном описании элементы.

Таблица 4.1. Описание с подчеркнутыми элементами, которые нужно определить и реализовать в игре

- Нам нужны «крипы», то есть враги, которые пытаются добраться до дома игрока.
- Нам нужна система рождения врагов волнами.
- Нам нужна система следования по маршруту или заранее заданные дороги и спроектированные вручную уровни.
 - Нам нужны башни, которые будут стрелять в крипов.
 - Нам нужна возможность строить и размещать эти башни.
 - Нам нужна боевая система, согласно которой будет работать стрельба.
 - Нам нужны условия победы и поражения.

После этого просматриваем список и спрашиваем себя, как будет работать каждый пункт.

Нам нужны «крипы», то есть своего рода враги, которые пытаются добраться до дома игрока. Что это значит?

«Крипы» должны быть простыми и несообразительными существами, которые просто идут вперед к своей цели. Игрок будет в них стрелять, поэтому им понадобятся какие-то защитные способности. Предположим, что некоторые из крипов смогут контратаковать. Тогда им понадобятся и атакующие способности. Для разнообразия нужно предусмотреть несколько типов крипов.

Нам нужна система рождения врагов волнами. Что это значит?

Мы хотим, чтобы вначале крипов было немного и они были легкими. Затем игра будет усложняться. Нам понадобится механизм порождения крипов нужного типа и необходимого количества для каждого уровня. У дизайнера должна быть возможность определять параметры волн для каждого уровня.

Нам нужна система следования по маршруту или заранее заданные дороги и спроектированные вручную уровни. Что это значит?

Предположим, что на уровне игрок сможет расставлять башни в любом месте. Нам понадобится система нахождения пути, чтобы крипы смогли проложить себе дорогу до базы.

Нам нужны башни, которые будут стрелять в крипов. Что это значит?

Это будут строения, которые можно размещать, но нельзя передвигать. Их

местоположение имеет значение. Как и у крипов, у башен должны быть как оборонительные, так и атакующие способности, но атака должна вестись на определенное расстояние – так будет интереснее. Для разнообразия можно предусмотреть разные типы башен – каждый со своими преимуществами и недостатками: одни сильнее в обороне, другие в атаке, с разной дальностью стрельбы и т. д. У игрока должна быть возможность модернизировать башни или усиливать их показатели.

Нам нужна возможность строить и размещать эти башни. Что это значит?

Для размещения или модернизации башен должна быть своя цена. Если есть цена, то должна быть валюта и способ ее заработать. Предположим, игрок зарабатывает валюту, расчищая волны крипов. Крипы должны иметь различную стоимость в валюте в зависимости от типа и показателей.

Нам нужна боевая система, согласно которой будет работать стрельба. Что это значит?

Мы собираемся использовать очки атаки и защиты и полоски здоровья – подобно тому, как они используются в ролевых играх. По мере получения повреждений очки здоровья уменьшаются. У башен (а также у атакующих крипов) должен быть показатель дальности атаки – свой для каждого типа юнитов.

Нам нужны условия победы и поражения. Что это значит?

Игра заканчивается, когда один из выживших крипов добирается до базы игрока. Мы решили, что в игре нельзя «победить» – это просто бесконечная последовательность уровней, сложность которых будет повышаться. Для того чтобы следить за прогрессом, нам понадобятся очки счета; на трудных уровнях счет растет быстрее. Также добавим таблицу лучших результатов и таблицу лидеров для состязания между игроками.

Итак, простое описание геймплея на три предложения быстро превратилось в описание достаточно большого количества механик и систем – боя, строительства, нахождения маршрута, прогрессии, экономики и т. д. И мы уже начали выделять внутри них отдельные специфические механики – очков атаки и защиты, усилений и усовершенствований, оружия, дальности действия эффектов, способов заработка и расходов валюты.

В процессе дизайна сверху вниз мы составляем подобные описания для разных частей геймплея, а затем стараемся подобрать системы и механики, благодаря которым этот геймплей будет реализован. После того как появились идеи насчет систем и механик, можно переключиться на дизайн снизу вверх, определить порядок реализации и попытаться создать простой прототип для тестирования наших дизайнерских идей.

Но для того, чтобы приступить к экспериментам с дизайном, необязательно дожидаться стадии прототипирования. До создания первого прототипа можно размышлять о различных дополнительных идеях «на бумаге», представляя в подробностях оказываемое ими влияние.

Что будет, если добавить систему заклинаний и магии?

Заклинания могут дополнять бой и предоставлять новые типы оружия и усовершенствований, но вместо валюты на них тратятся очки заклинаний.

Как получать заклинания?

Возможно, игроку понадобится собирать свитки, чтобы разблокировать заклинания, и накапливать очки маны, чтобы эти заклинания вызывать.

Как получать ману?

Возможно, очки маны оставляют при передвижении какие-то типы крипов, а свитки можно исследовать, потратив валюту на постройку исследовательской башни. Значит ли это, что нужно добавить исследовательские башни? Как они работают?

И так далее.

Сочетая анализ сверху вниз на основе нарратива с экспериментами в новых направлениях и их оценкой, гейм-дизайнер пускается в своего рода исследование пространства дизайна системы. Проводить такое исследование становится легче при знакомстве с многочисленными примерами систем в других играх и в других сферах жизни, где можно находить интересные детали в существующих моделях и переносить в игру.

Балансировка систем

В предыдущем разделе мы описали общую структуру систем и механик в вымышленной игре жанра «башенной защиты». При реализации данной структуры нам нужно будет также определить, *как именно* будут взаимодействовать между собой части данной структуры. Какова будет стоимость башен и какую валюту мы будем использовать? Сколько будет очков защиты и атаки у каждого крипа? Сколько валюты мы получаем за убийство одного врага?

Процесс поиска точных численных показателей и принятия решений на нижних уровнях называется *балансировкой* игры. Балансировка – это часто итеративный процесс, в очень большой степени определяемый посредством экспериментов и игрового тестирования с целью нахождения верного баланса всех этих переменных величин. Если игра настроена слишком *строго* (когда игроку трудно прогрессировать и добиваться успеха), она становится более требовательной и тем самым, привлекательной для одних игроков, но слишком сложной и неинтересной для других, недостаточно опытных. Если же игра настроена *нестрого*, когда наград имеется в изобилии, а вызовы просты, то новичкам становится легче, но это расстраивает опытных игроков, у которых процесс вызывает скуку.

При этом в балансе может не оказаться никакой пресловутой «середины», удовлетворяющей игроков всех типов, поэтому многие игры в виде подручного средства прибегают к системе нескольких уровней сложности. Еще хуже, когда разные системы игры балансируются по отдельности, в результате чего игра кажется непоследовательной. Например, если сражения настолько просты, что крипы умирают очень быстро, а экономика настолько сурова, что не позволяет легко создавать новые башни, то это может разочаровать игроков независимо от их опыта.

Способы балансировки

Существуют четыре общих способа балансировки систем: ручная балансировка, создание внешних моделей игры, использование надстройки к игре для симуляций игровых событий, сбор и анализ статистики плейтестов игроков.

Принцип *ручного баланса* понятен из его названия. Допустим, у нас появились какие-то идеи для улучшения. Заходим в редактор игры, меняем некоторый контент (настраиваем параметры, карту уровня и т. д.) и играем, чтобы посмотреть, как это работает. На основании тестирования формулируем очередные идеи и повторяем процесс. Иногда у нас действительно получаются улучшения, но регулярно возникают проблемы – тогда приходится отменять изменения. Так мы улучшаем игру итеративно, раз за разом повторяя шаги по настройке и тестированию, пока не добьемся удовлетворительного результата. Данный способ медленный, но тщательный: так как в игру постоянно играет сам гейм-дизайнер, он может замечать не только влияние изменений на настраиваемую систему, но и их побочное влияние на смежные. К сожалению, время, которое в данном случае придется потратить на игровое тестирование, непозволительно велико (поскольку у игр огромное пространство состояний), поэтому чаще дизайнеры прибегают к изолированному тестированию отдельных частей игры и стараются понять, как на основании этого программа будет вести себя в целом.

Чтобы избежать длительного и многократного ручного тестирования геймплея, дизайнеры иногда создают *модели* игровых систем с помощью разного программного обеспечения – например, Microsoft Excel или Google Sheets. Особенно это полезно для систем с многочисленными числовыми показателями, таких как экономика или сражения. В этом случае гейм-дизайнер создает упрощенную модель только одной рассматриваемой системы и проводит симуляции (то есть смотрит, как эта модель будет развиваться при действиях игрока). Этот способ гораздо быстрее, чем ручное тестирование, но обладает одним значительным недостатком: поскольку электронная таблица – это всего лишь модель какой-либо игровой системы, она неизбежно проще самой игры и будет отличаться от нее. Таблица не охватывает всех игровых деталей и может привести к ошибочным выводам – в зависимости от того, насколько хорошо модель соответствует игре.

Такое расхождение между моделью и собственно игрой можно преодолеть иным способом – используя надстройку к игре, которая симулирует действия игрока, после чего проанализировать последствия. Такая *симуляция* – своего рода автономное тестирование, которое бывает гораздо быстрее ручного (например, система сама по себе проигрывает все действия на гораздо большей скорости или же параллельно совершает множество действий на ферме). Но данный подход сопряжен с трудностями, к тому же он может быть весьма дорогостоящим. Игра должна подходить для быстрой симуляции, к тому же нужно придумать, как реализовать ввод различных действий фиктивного игрока, что довольно трудно для программ с высоким уровнем сложности. Один из возможных способов обойти такие проблемы – ограничить тестовую симуляцию одной системой (такой как экономика или сражения). Это значит, что нужно предусмотреть ввод действий игрока, связанных только с данной системой (таких как покупка, продажа и зарабатывание валюты). Это легче, чем заставлять компьютер проигрывать всю игру, но в результате появляются те же недостатки, что и в модели с электронными таблицами (Tozour, 2013).

Последний способ – предложить игрокам поиграть в игру (сразу всем или ограниченной группе тестировщиков), собрать данные об их действиях, а потом произвести соответствующие настройки. Такой основанный на *аналитике* подход имеет несколько преимуществ. Результаты основаны на реальном поведении игроков, а не на предположениях или моделях, плюс игроки в большом количестве, как правило, неплохо находят слабые стороны и доминирующие стратегии. Но это вовсе не заменяет остальные варианты балансировки. К ним обязательно нужно прибегать – прежде всего потому что тестировщикам предлагается плохо сбалансированная игра, и они могут столкнуться с трудностями, а мы не получим хорошей обратной связи. Тем не менее аналитика – отличное средство для тонкой настройки систем, уже находящихся в относительно неплохом виде, или для поиска не предусмотренных дизайнером доминирующих или вырожденных стратегий. Аналитику часто используют для запуска и итеративной настройки какой-либо особенности – то есть для ввода в игру новой черты, которую сначала испытывают на небольшом количестве игроков. Таким образом можно быстро убедиться, ведет ли она себя так, как было задумано, а если этого не происходит, перебалансировать игру прежде, чем предоставить ее игрокам.

Роль балансировки в производственном процессе

Независимо от способов в течение срока жизни игры к балансу приходится прибегать довольно часто – во время *разработки* отдельных систем и интеграции готовых доработок, а также во время стадии *отладки*, когда игра собирается в единое целое.

При разработке отдельной системы неплохой идеей будет черновое тестирование прямо в процессе (например, для того чтобы убедиться, что персонаж платформера двигается правильно), прежде чем переходить к дизайну уровней. Баланс не должен быть идеальным, но как только мы объединим несколько систем, недостатки одной системы повлияют на их общий игровой опыт, так что лучше предусмотреть все заранее.

После интеграции нескольких систем очередная балансировка понадобится для того, чтобы они более гладко взаимодействовали друг с другом. Например, в шутерах принято как можно раньше совместно тестировать три главных компонента – «персонаж», «камера» и «контроль», потому что все они влияют друг на друга, а вместе с остальными системами и на опыт игрока.

На альфа-стадии, когда примерно на своих местах находятся все системы, можно провести общую балансировку с учетом опыта взаимодействия игрока со всеми системами и контентом. На данном этапе вносимые изменения обычно бывают небольшими, но при тестировании сразу всей игры всплывает большое количество недостатков и возможностей для улучшения.

В этом смысле процесс балансировки схож с процессом работы над произведением изобразительного искусства – художник начинает с грубого наброска, а затем итеративно наполняет холст деталями.

Краткий итог

В этой главе мы рассмотрели игровые системы, состоящие из более мелких отдельных механик. Ключевые ее положения таковы.

- *Системы* представляют собой совокупность механик, настроенных для совместной работы определенным образом. Экономика, инвентарь, сражение и другие примеры систем обсуждаются на примере игры *Diablo*.

Системы – это полезные абстракции, позволяющие нам анализировать игры как наборы различных групп механик. Но системы не следует воспринимать как полностью независимые; они должны взаимодействовать и влиять друг на друга – например, посредством общих механик, используемых в различных целях.

- Бывает полезным рассматривать системы как *цепочки* или *циклы* более мелких взаимодействий для производства определенных ресурсов или для их конверсии в другие. На примере игр из серии *Diablo* можно понять, как механики сражения, сбора добычи, продажи ее за золото, пополнения запасов и исцеления совместно образуют цикл ресурсов, связывающий между собой несколько систем (таких как экономика, инвентарь и сражения).

- *Циклы конверсии* можно анализировать, рассматривая их производительность и изменение со временем по мере того, как в них участвует игрок. Анализ помогает настроить их и предотвратить недобросовестное использование (например, «гринд») в целях накопления чрезмерного количества ресурсов.

- *Циклы обратной связи* наблюдаются, когда система имеет внутреннее состояние, влияющее на результат действия данной системы, а этот результат служит основанием для изменения внутреннего состояния и создания сценария, при котором результат и внутреннее состояние взаимно влияют друг на друга с течением времени. Два типичных примера – это *циклы положительной обратной связи*, усиливающие изначальную разницу, и *циклы отрицательной обратной связи*, противодействующие изменениям. Циклы обратной связи очень широко распространены в играх, но циклы «взлетающей» обратной связи могут привести к нежелательным долгосрочным последствиям, таким как чрезмерное усиление имеющихся на ранней стадии преимуществ, вследствие чего в игре становится невозможно проиграть или победить. Балансировка этих циклов – одна из главных задач гейм-дизайна.

- Системы также бывают *эмерджентными* или *хаотическими*.

- Они распространены в реальном мире, но в играх бывает довольно трудно сделать так, чтобы эти системы вели себя корректно и доставляли удовольствие игроку.

- Как понять, какие системы должны быть в задуманной нами игре? Можно начать с составления нарративного описания геймплея, с историй пользователей, а затем попытаться подобрать конкретные системы и механики, необходимые для реализации данного нарратива.

- Наконец, при реализации систем нужно помнить об их *балансировке* – например, о

назначении подходящих цен предметам в экономической системе, о задании подходящих параметров для оружия в системе сражений и т. д. Успех взаимодействия игрока и систем или совокупности систем в большей степени зависит от правильного баланса параметров – на основе опыта или исходя из результатов экспериментов.

Теперь, рассмотрев механики и системы как отдельные инструменты, мы готовы перейти к следующей теме наших рассуждений. Далее поговорим о геймплее и о том, как взаимодействие с механиками и системами со временем превращается в субъективный игровой опыт.

Дополнительная литература

Анализ механик, систем и их взаимодействия – центральная тема книги «Продвинутый гейм-дизайн» Селлерса (*Advanced Game Design*, Sellers, 2017), крайне рекомендуемой для увлеченных этой темой читателей. Тем, кто интересуется более формальным моделированием систем, можно порекомендовать книгу «Игровые механики: продвинутый гейм-дизайн» (*Game Mechanics: Advanced Game Design*, Adams and Dormans, 2012) с очень подробным описанием методов моделирования и проведения симуляций с помощью моделей, показывающих, как они ведут себя до их интеграции в игру. Такого рода формальные модели известны также под названием «*игровые грамматики*» (*game grammars*). Возможно, самое раннее упоминание о них встречается в презентации Костера «Грамматика гейм-дизайна» (*A Grammar of Gameplay*, Koster, 2005).

Вне контекста игр системы были объектом исследований в различных сферах, наиболее интуитивно знакомые из которых – экономика и бизнес. Великолепным введением в современное понимание крупномасштабных и сложных систем вроде мировой экономики послужат такие учебники, как *The Economy* (CORE Team, 2017). Исторический интерес могут представлять классические исследования динамики систем – «Индустриальная динамика» Форрестера (*Industrial Dynamics*, Forrester, 1961) и «Кибернетика» Винера (*Cybernetics*, Wiener, 1961).

Упражнения для индивидуального выполнения

4.1 Игровые системы

Как механики, так и системы – это строительные блоки гейм-дизайна, но они отличаются по масштабу и составу. Как отличить механику от системы? Приведите пример игровой системы и механик, которые ее образуют.

4.2 Системные уровни

Выберите хорошо известную вам игру.

а. Постарайтесь перечислить как можно больше систем в этой игре с коротким описанием того, что они делают (одно-два предложения на каждую).

б. Теперь выберите некоторые из этих систем с одной или несколькими общими механиками. Опишите, как они взаимодействуют между собой, какие механики для них общие.

4.3 Цепочки и циклы

Выберите хорошо известную вам игру. Найдите цикл ресурсов в ней (их может быть несколько – в этом случае выберите один) и подробно опишите его. Какой это, по вашему мнению, цикл – прибыльный или неприбыльный? Если прибыльный, то может ли он привести к «взлетному» росту ресурсов? Объясните свой ответ.

4.4 «Гринд»

Термином «гринд» часто обозначаются циклы ресурсов, которые используются только для получения запасов или повышения показателей, но которые при этом не интересные и не увлекательные.

а. Приведите пример игры, в которой вы занимались «гриндом», при этом не получая удовольствия от процесса? Опишите эту цепочку ресурсов.

б. Опишите свою мотивацию для такого «гринда». Почему вы продолжали «гриндить» и проходить через этот цикл?

в. Представьте, что данный «гринд» был удален из игры. Стала бы она от этого лучше? Как бы это повлияло на ваше восприятие игры?

4.5 Циклы положительной обратной связи

Выберите хорошо известную вам настольную или видеоигру с яркими циклами положительной обратной связи.

а. Опишите, как цикл обратной связи ускоряет производство некоторых ресурсов в зависимости от количества произведенных ранее.

б. Представьте, что вы можете изменить игру так, чтобы цикл обратной связи слабее подкреплял сам себя. Возможно ли такое вообще? Если да, то будет ли это благоприятным изменением? Как это повлияет на геймплей?

4.6 Циклы обратной связи

Выберите игру, в которую можно играть против других игроков или против ИИ, и опишите циклы положительной и/или отрицательной обратной связи в ней. Как эти циклы влияют на фазы конфликта и развязки?

Упражнения для выполнения в группе

Г 4.1 Дизайнерское задание: ускорение «Монополии»

Цель данного задания – ускорить геймплей «Монополии», продумав возможные изменения, реализовать их и протестировать новый вариант. Предполагается, что у вас имеется доступ к данной настольной игре. Вместо нее преподаватель может предложить другую.

а. Сыграйте в настольную игру «Монополия» или в один из ее вариантов, чтобы напомнить себе геймплей. Затем подумайте, как можно ускорить последнюю стадию игры, изменив некоторые механики и системы. Можно вносить любые изменения по вашему желанию. Вот лишь некоторые предложения: усилить цикл «богатые становятся богаче» (повысить ренту или сделать так, чтобы она накапливалась быстрее), удалить элементы отрицательной обратной связи, помогающие отстающим игрокам оставаться на плаву (денежное вознаграждение за прохождение клетки GO («Вперед»), удалить случайные карты и возможность отдавать недвижимость в залог и т. д.

б. Запишите свои предложения и обсудите, как, по вашему мнению, они повлияют на скорость игры.

в. Сыграйте в игру снова, наблюдая за тем, насколько быстро по сравнению с оригиналом игроки накапливают или теряют деньги, а также выигрывают или проигрывают.

г. Подробно опишите внесенные изменения с указанием затронутых ими действий, объектов и правил.

д. Объясните, почему вы предложили именно такие изменения и какого результата вы от них *ожидали*.

е. На основании своих наблюдений обсудите, как эти изменения повлияли на геймплей

на самом деле .

ж. Объясните, как, по вашему мнению, это произошло – что именно в новых правилах изменило игру? Воспринимается ли она как более быстрая или нет? Чем объясняется такое отличие?

5 Геймплей

В предыдущих главах мы рассматривали механики, строительные блоки геймплея и собранные из них системы, образующие цепочки, циклы и взаимодействующие между собой механизмы. В данной главе мы рассмотрим, как все эти элементы вместе образуют геймплей.

Геймплей – это то, что происходит, когда мы участвуем в продолжающемся взаимодействии с игровыми механиками и системами, а также с другими игроками в многопользовательских играх. Геймплей – это *динамическая активность*, возникающая в результате взаимодействия игроков с механическими элементами, а участие в таком взаимодействии производит определенное впечатление на игрока, то есть формирует его игровой опыт.

В данной главе постараемся подвергнуть геймплей систематическому анализу. Ранее мы рассуждали о циклах обратной связи и о системах. Теперь обратим внимание на другой вид циклов, в которых игроки взаимодействуют с игрой; это взаимодействие носит циклический, повторяющийся характер.

В первой части главы мы научимся анализировать игры с точки зрения геймплейных циклов различной периодичности и исследуем их взаимодействие с системами. Во второй части рассмотрим мотивацию игрока, принимающего участие в данных циклах, и подумаем, как можно поддерживать мотивацию посредством дизайна геймплея.

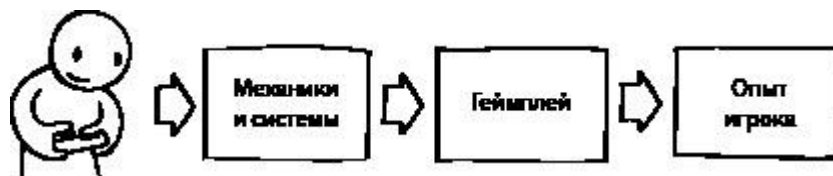


Иллюстрация 5.1. Повторение базовой модели

Мотивирующий пример: The Sims

Представим, что мы играем в игру *The Sims 4* и управляем жизнью вымышленных людей в симуляции небольшого городка. Мы обладаем полным контролем над жизнью наших симулированных жителей, но нам необходимо поддерживать их хорошее настроение и удовлетворять многочисленные, зачастую конфликтующие между собой потребности.

Допустим, игра идет в доме Элис и Боба. Вернувшийся с работы Боб увидел, что Элис устроила вечеринку. Он совершенно измотан, его энергия близка к нулю. Но в то же время Боб несчастен из-за недостатка социальной жизни и вправду рад общению.

Нам нужно решить, как ему поступить. Конечно, завести знакомство с кем-нибудь неплохо, но Боб и в самом деле очень устал. С одной стороны, он рискует «отрубиться» от усталости и проспать весь следующий день. С другой стороны, общение с новыми людьми сделает персонаж более счастливым и, как следствие, более уверенным в себе и работоспособным, а это неплохой способ получить повышение, к которому Боб так стремится. Его жизненное устремление – когда-нибудь стать генеральным директором, а повышение значительно приблизит персонаж к этой цели. Итак, мы принимаем решение – Бобу придется взбодриться, выйти из своей комнаты и пообщаться. Это ради его же собственного блага. Сон может подождать.



Иллюстрация 5.2. Скриншот из игры *The Sims 4*

Всего лишь одно решение заставило нас задуматься о выборе, целях и задачах на нескольких уровнях.

- *Краткосрочная цель* : необходимость поддерживать текущее благосостояние сима и удовлетворять его текущие потребности.
- *Среднесрочная цель* : необходимость подумать о том, что произойдет сегодня и завтра.
- *Долгосрочная цель* : необходимость позаботиться об исполнении мечты сима о том, чтобы стать генеральным директором.

Во время игры мы должны рассматривать цели и действия в разном временном масштабе одновременно. И это не случайно. Игры намеренно предлагают задачи («вызовы»), относящиеся одновременно к нескольким уровням, а разные уровни поддерживают друг друга. Без краткосрочных вызовов нам было бы нечем заняться прямо здесь и сейчас, а без долгосрочных целей не будет повода возвращаться в игру.

Выполнение целей, относящихся к разным временным масштабам, подразумевает взаимодействие с *разными системами* игры. В *The Sims* краткосрочные цели основаны на системе «желаний и потребностей», игра сама моделирует симам различные потребности, которыми необходимо управлять. Например, голодному симу нужно очень быстро найти еду, иначе ему станет плохо. Среднесрочные цели основаны на системе «работы» и игровой экономики – симы должны находить себе работу с разными условиями, оплатой и расписанием, чтобы зарабатывать деньги и получать то, о чем они мечтают в долгосрочной перспективе. Долгосрочные цели основаны на системе «жизненных целей» – у сима может быть мечта стать генеральным директором или известным музыкантом. Их успех зависит от предварительного планирования и от того, сколько денег и времени будет потрачено на различные вещи и занятия, эффект которых проявится позднее – например, на посещение вечерних курсов или на обучение какому-либо навыку.

Геймплейные циклы

Мы рассмотрели действия, которыми игроки занимаются неоднократно. Назовем их *геймплейными циклами*, поскольку они представляют собой цикличные действия – игрок постоянно принимает решения, действует согласно им, а затем возвращается в ту же точку принятия решений.

В нашем примере из *The Sims* самый маленький геймплейный цикл – это *выбор действия*. Мы выбираем, какое очередное действие должен выполнить сим (отправиться

спать, пойти пообщаться, сделать ужин), ждем его выполнения, наблюдаем за результатом (повысилась энергия, сим стал более довольным) и повторяем. Это происходит очень часто, порядка нескольких раз в минуту.

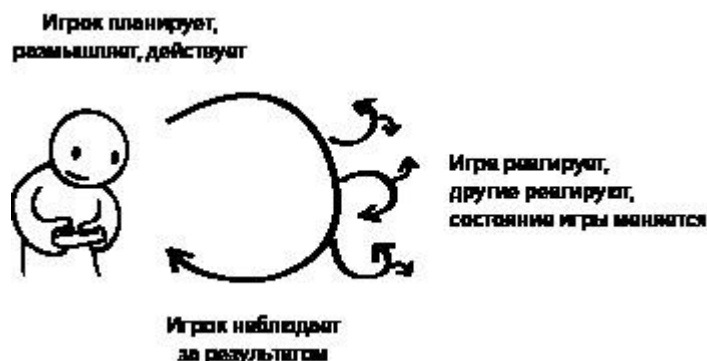


Иллюстрация 5.3. Действие игрока вызывает реакцию со стороны игровых систем и других участников, на которую игрок должен реагировать, и это создает продолжающийся цикл

В то же время в игре присутствуют и более крупные геймплейные циклы. Например, каждое утро нам нужно проследить за тем, чтобы сим проснулся отдохнувшим, отправился на работу и зарабатывал деньги. Это требует иного рода решений, принимаемых раз в несколько минут реального времени. Наконец, если рассматривать периоды в несколько дней, нам нужно следить за тем, чтобы симы совершенствовали свои навыки и занимались карьерой. Такие ответственные решения приходится принимать не так часто, возможно раз в десять – двенадцать минут, но они тоже требуют внимания.

Периодичность циклов

У геймплейных циклов бывает разная *периодичность* (частота). Быстрые циклы повторяются каждые несколько секунд или минут игрового времени, а очень медленные (например, достижение карьерной цели) требуют взаимодействия лишь изредка.

В качестве других примеров геймплейных циклов можно вкратце рассмотреть игру в жанре «данжен-кроулер» вроде *Diablo III*, описанную в предыдущей главе.

- Каждые несколько секунд (также называемые *микроуровнем*): двигаться, атаковать, обороняться, наложить заклинание, экипировать, взять, бросить.
- Каждую минуту или две: найти врагов, сразиться с ними, собрать добычу, найти сокровище.
- Каждые пять – десять минут: покинуть лагерь, зачистить участок, вернуться и продать добычу, исцелиться.
- Каждые пятнадцать – тридцать минут: зачистить всю зону, поменять состав команды, повысить навыки персонажей.
- Каждые несколько часов: исследовать весь регион игрового мира, продвинуться по сюжету.

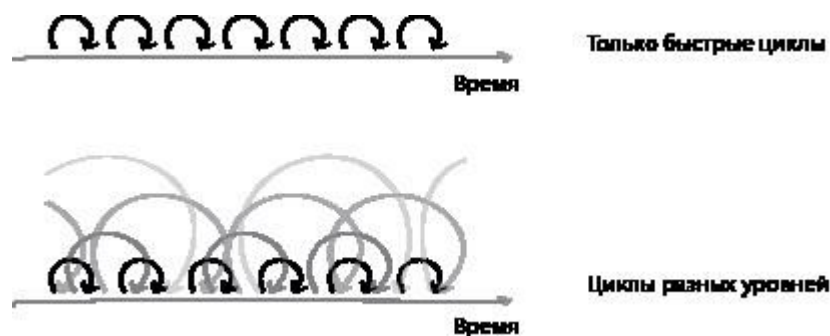


Иллюстрация 5.4. Сравнение последовательности одних лишь быстрых циклов с сочетанием циклов различных уровней, имеющих разную периодичность

Циклы с разной периодичностью активны *одновременно*. Внимание игрока больше всего сосредотачивается на микроуровневых действиях. Но эти действия имеют и макропоследствия, поэтому игроки во время игры должны помнить обо всех уровнях, чтобы понемногу двигаться по долгосрочным циклам, продолжая взаимодействовать с более короткими.

Нам как гейм-дизайнерам необходимо проследить за активностью всех циклов и предусмотреть в игре различную периодичность принятия решений.

Концентрические схемы

Популярный способ представления геймплейных циклов – изображение их в виде *концентрических схем* (или так называемых *луковичных диаграмм*). На иллюстрации 5.5 показан пример такой диаграммы, упрощенно передающей геймплей игры «*Монополия*».

Самый маленький и быстрый цикл расположен в центре, за ним в расширяющихся кругах располагаются циклы все большего периода. Проходя самый маленький цикл раз за разом, они также проходят и через внешние слои, но не так быстро и лишь частично.

На иллюстрации 5.6 показаны примеры из *Diablo* и *The Sims*, также в упрощенном виде.

Концентрические схемы – это средство коммуникации. С их помощью легко описывать существующие в дизайне геймплейные циклы и сообщать о них другим.

Поскольку считается, что циклы разной периодичности должны более или менее удачно распределяться по всей игре, то по таким схемам бывает легко понять, где в игре наблюдается разрыв, который можно заполнить какой-нибудь интересной деятельностью.



Иллюстрация 5.5. Упрощенная концентрическая схема «*Монополии*»



Иллюстрация 5.6. Упрощенные концентрические схемы игр *Diablo* и *The Sims*

Базовый цикл

Дизайн различных пересекающихся между собой активностей – занятие трудное. В этом помогает метод, согласно которому сначала нужно сосредоточиться на *базовом цикле (core loop)*, а потом постепенно отталкиваться от него. Базовый цикл – это минимально значимый уровень активности, самый маленький цикл, имеющий смысл для игрока и доставляющий ему то удовольствие, благодаря которому игрок продолжает играть.

В некоторых типах игр базовым циклом является объективно мельчайший, или *микроцикл*, потому что интерес игрока поддерживают как раз именно действия на микроуровне. Чаще это встречается в экшен-играх, например в гонках, где достаточно увлекательно просто стараться удерживать автомобиль на трассе, обгоняя других и стремиться занять первое место.

В других же играх базовый цикл должен быть более сложным по сравнению с микродействиями. Особенно это верно в отношении игр с множеством систем, потому что в них микродействия не так уж интересны и недостаточно мотивируют игрока на продолжение игры. Например, в *The Sims* не слишком увлекательно просто удовлетворять физиологические потребности персонажей, поэтому основной цикл смещается на уровень «пойти на работу, заработать денег, купить что-нибудь для дома, повторить». В *Diablo* это может быть «отправиться на приключения, сразиться, собрать добычу, вернуться в лагерь, продать добычу, купить или усовершенствовать снаряжение, повторить».

Базовый цикл требует усиленного внимания, поскольку игрок будет взаимодействовать с ним *все время*, поэтому нужно сделать так, чтобы в процессе участник всегда получал удовольствие, даже если в долгосрочной перспективе внимание игрока переключается на более трудные проблемы. Дело усложняется еще тем, что игроки бывают разных категорий, поэтому более опытных основной цикл может привлекать уже не так сильно, как новичков.



Иллюстрация 5.7. Концентрическая схема из иллюстрации 5.6 с выделенным базовым циклом

Распределение циклов по уровням

Базовый цикл – это удобная отправная точка для расширения игры. Например, с помощью следующих шагов.

1. Начать с микроуровня, на котором происходят самые мелкие действия в игре. Обычно они не очень интересны, но интерактивны и позволяют гейм-дизайнеру протестировать основы.

2. Исследовать и расширять микроуровень в поисках базового цикла. На данной стадии игра становится достаточно увлекательной, чтобы поиграть в нее хотя бы несколько минут. Нащупав этот уровень, мы методом итераций улучшаем его и исследуем различные идеи расширения.

3. Теперь можно добавлять более масштабные по времени циклы. Если мы еще не придумали их, то базовый цикл может подкинуть кое-какие идеи по этому поводу. Это может сделать игру интересной в течение длительного времени. Более крупные циклы должны поддерживать более мелкие, взаимодействуя с ними или используя их результаты.

В *The Sims*, например, мы описали увлекательный цикл зарабатывания и траты денег и удовлетворения физиологических потребностей симов. Но сам по себе данный цикл обладает относительно кратковременной увлекательностью. Добавляя более крупные уровни с такими целями, как привлечение в семью новых членов или подъем по карьерной лестнице, мы, помимо добавления долгосрочных задач, изменяем параметры базового цикла, освежаем его и делаем еще более интересным.

Продвижение по одному циклу не обязательно должно совпадать с продвижением по другим – бывает интереснее, когда разные циклы слегка рассинхронизированы. В некоторых типах игр такая рассинхронизация циклов особенно увлекательна тем, что поддерживает постоянный «челлендж». Возьмем для примера игру *Factorio*, посвященную производству ресурсов и логистике, где игрок управляет сложными производственными цепочками, а крайне важную роль играет планировка «фабрики». Но цели, преследуемые игроком на раннем этапе игры, заставляют его выбирать планировку, которая станет проблемой на более поздних стадиях – ее придется переделывать или даже полностью разбирать и создавать заново. Необходимость находить баланс между кратковременными целями в начале игры и будущими целями в долгосрочной перспективе становится основным источником удовольствия (хотя часто и разочарования).

Циклы и системы

Геймплейные циклы возникают в результате взаимодействия с механиками и системами, а также с другими игроками. Как уже рассматривалось ранее на примере *The Sims*, различную периодичность часто поддерживают разные игровые системы, работающие вместе.

Для другого примера вернемся к «данжен-кроулерам». Мы уже упоминали цикл «покинуть лагерь, зачистить зону, вернуться с добычей». Каждый раз, возвращаясь в лагерь, участник вспоминает, что у него есть разнообразные цели, к которым нужно как-то двигаться: экономическое развитие (продажа и покупка вещей), развитие персонажей (повышение их уровней или тренировка), крафт (усовершенствование оружия) и т. д. А эти цели, в свою очередь, помогают двигаться к долгосрочным: накопить денег, прокачать персонажа, пройти сюжет или добыть более подходящее оружие для последующих сражений.

Периодичность	Геймплейные циклы	Примеры соответствующих им систем
Каждые 1–2 минуты	Найти врагов, сражаться с ними, собрать добычу, выйти сокровища	Исследование местности, сражения, вепя, параметры персонажа, инвентарь
Каждые 5–10 минут	Покинуть лагерь, зачистить участок, вернуться, продать добычу, исцелиться	Исследование местности, вепя, инвентарь, экономика, крафтинг, модернизация, параметры персонажа, навыки
Каждые 15–30 минут	Зачистить всю зону, поменять состав команды, повысить навыки персонажей	Инвентарь, крафтинг, специализация персонажей, мидлсменские предметы
Каждые несколько часов	Исследовать весь регион, игнорить мир, пройти по сюжету	Специализация персонажей, управление командой, сюжетная арка компании/истории

Подобно тому, как мы перечисляли действия для каждого цикла, можно указать соответствующие системы и механики – например, как показано в таблице 5.1.

Таблица 5.1. Примеры игровых циклов и поддерживающих их систем

Видно, что в различных циклах с разной периодичностью часто задействованы специфические системы (в отличие от тех, которые работают при всех периодичностях). Это объясняется тем, что разные системы имеют различное поведение и настраиваются по-разному. Трудно придумать какую-то одну систему, бросающую игроку интересные вызовы при разной периодичности – от уровня микродействий до уровня долгосрочных целей.

Тем не менее циклы часто делят некоторые системы со своими «соседями». Например, инвентарь и параметры персонажа задействованы в нескольких циклах разной периодичности. И это хорошо, потому что означает, что одно действие, с помощью которого проходит цикл, также влияет на прогресс (прямой или обратный) других циклов, благодаря чему такие действия становятся стратегически более интересными.

Наконец, некоторые из данных циклов сосредотачиваются не только на взаимодействии, но и затрагивают элементы повествования и сюжета. В такой ориентированной на сюжет ролевой игре, как *Dragon Age: Origins*, некоторые из более коротких циклов, как правило, сосредоточены на сражениях, повышении уровней и других подобных системах. Но долгосрочная мотивация достигается благодаря затягивающему повествованию, из-за которого игроку хочется узнать, что случится дальше, способствуя

своими действиями развитию истории. Мы вернемся к анализу такого рода повествовательных элементов позже, в главе 6 «Макроструктура».

Мотивация игрока

Итак, мы рассмотрели, как механики и системы формируют геймплей, предлагая игроку различные активности и вызовы, возникающие в результате взаимодействия с ними. Взаимодействие игрока с механиками и системами носит повторяющийся характер, образуя циклы повторяющейся активности разной периодичности. Совмещая различные циклы разных уровней, мы предлагаем игроку геймплей, бросающий вызовы как в краткосрочной, так и в долгосрочной перспективах.

Но все это строится на том предположении, что игрок *захочет* выполнять данные циклы. Если они участнику неинтересны, то он не захочет играть в игру и не получит удовольствия от нашего столь элегантно сконструированного механизма.

Так что же мотивирует игрока сыграть в конкретную игру? В главе 2 «Опыт игрока» мы уже поговорили о мотивации высокого уровня – желании получить опыт определенного вида (например, вызов, стратегия или общение), о том, как испытываемое от игры удовольствие зависит от типа личности и как все это учитывается в общем дизайне игры.

В данном разделе обратим внимание на мотивацию низкого уровня – а именно на то, что удерживает игрока в игре на протяжении некоторого времени (от нескольких минут до нескольких часов), а также рассмотрим стоящие за этим психологические принципы.

Внутренняя и внешняя мотивация

Прежде всего поговорим о двух крупных типах мотивации.

- *Внутренняя мотивация* : когда игрок внутренне заинтересован в активности и в ее результате. Значит в активности есть нечто, что нравится игроку (например, танцевать в *Dance Dance Revolution* , обучение полету вокруг света в *Microsoft Flight Simulator*) и что соответствует его предпочтениям.

- *Внешняя мотивация* : когда заниматься какой-либо активностью игрока заставляет нечто внешнее по отношению к ней – например, желание получить результат, но не выполнить само действие. Примером внешней мотивации может служить «гринд» ради получения хороших вещей или состязание в матче ради приза.

Эти типы мотивации не исключительны, а большинство видов активности располагается где-то между исключительно внутренней и исключительно внешней мотивацией. Например, в такой активности, как игра в соревновательную многопользовательскую игру, обычно сочетается внешняя мотивация (желание попасть в список лидеров) с некоторым внутренним удовольствием от самой игры. Схожим образом в гоночной игре внутренней мотивацией может быть желание получить удовольствие от быстрой и непринужденной езды, а внешней – мечта получить приз в соревнованиях. Обе эти мотивации предполагают награды разного типа (Hodent, 2017, 65).

Подробности внутренних или внешних мотивов тесно связаны с самим игроком. Наблюдая со стороны, нельзя сказать, в какой степени человек играет ради награды, а в какой просто ради удовольствия, или же сочетает мотивы. Поэтому, хотя игра и не может напрямую пробудить внутреннюю мотивацию игрока, она по возможности должна *поддерживать* ее. Нельзя заставить участника танцевать в виртуальном состязании в *Dance Dance Revolution* – мотивация должна присутствовать внутри него. Но если она присутствует, то можно подготовить почву и создать платформу для развития навыков и получения удовольствия.

Что касается внешней мотивации, то ею легче управлять искусственными методами.

Получить больше очков, попасть в список лидеров, получить приз или победить в состязании с другими игроками – такого рода мотивация распространена во многих типах игр. Она взывает к наиболее основным человеческим желаниям стать лучше, проверить свои силы, сравнить себя с другими и получить награду за усердие.

В идеальном случае программы должны сочетать внутреннюю и внешнюю мотивацию. Игра, которая дает внешние награды, но не интересна сама по себе, превращается в работу. С другой стороны, активность, приносящая удовольствие, но не предоставляющая обратной связи или каких-то внешних наград может показаться бесцельной и «несвязной». Как правило, игры больше всего привлекают, когда внутреннее желание игроков заниматься какой-либо активностью дополняется внешней «морковкой», награждающей их за старание и за совершенствование в том, от чего они уже получают внутреннее удовольствие.

Внутренняя мотивация: поток и обучение

Что делает игру интересной для игроков? Почему им нравится раз за разом возвращаться, проводя в ней минуты за минутами и долгие часы? У каждого свои причины и мотивы, но есть и общие черты, широко распространенные среди множества игроков.

В этом разделе мы сосредоточимся на идее овладения мастерством и навыками как на внутренне интересной для игроков цели, которая служит мощной мотивацией, объясняющей наше желание играть. Не все игроки реагируют на вызовы одинаково. Реакция во многом зависит от нашего внутреннего состояния в тот момент, когда мы сталкиваемся с определенной трудной задачей, от нашей способности преодолеть это препятствие, от обучения в процессе преодоления и улучшения своих навыков. Но мы воспринимаем все это как внутренне интересное, если оно соответствует нашим собственным интересам и уровню мастерства.

Любопытно, что интерес к преодолению трудных задач («вызовов») не зависит от того, насколько навыки, необходимые для решения данных задач «полезны» – они могут относиться и к такому «бесполезному» времяпрепровождению, как видеоигры. Связь между способностями и мастерством с одной стороны и играми с другой была подмечена еще в первых теоретических трудах, посвященных играм (Caillois, 1962).

Теория потока

Центральной концепцией в понимании игры как получения удовольствия от мастерства является концепция «потока». Обычно он описывается как чувство «абсолютного слияния» при выполнении какой-либо деятельности. Любой, кто забывался в игре или часами погружался в какую-то активность, не осознавая хода времени, испытывал это глубокое и радостное чувство.

Люди, когда-либо попадавшие в потоковое состояние, часто описывают схожие психологические эффекты, самый распространенный из которых – утрата ощущения времени, как и исключительная сосредоточенность на текущей задаче или потеря осознания себя и своего присутствия в мире. При нахождении в состоянии потока остается только мы и деятельность, которой мы занимаемся – ничто не стоит между нами и ею. Такой эффект проявляется в различных сферах: в умственных занятиях, в творческой деятельности, в физической активности – например, когда спортсмены выкладываются на пределе своих сил.

Психологическая теория потока утверждает, что поглощение каким-либо видом деятельности напрямую связано с навыками человека и с тем, насколько предлагаемый этой деятельностью вызов соответствует уровню его навыков (Csikszentmihalyi, Abuhamdeh and Nakamura, 2005). Если вызов слишком легкий, то человек испытывает скуку, а если слишком сложен, то разочаровывается или испытывает раздражение. Но когда игрок сосредоточен на задаче, соответствующей его уровню навыков, он входит в оптимальное состояние, в котором полностью поглощен своей деятельностью, как это показано на иллюстрации 5.8.



Иллюстрация 5.8. Иллюстрация канала потока игрока, в котором уровень сложности задачи соответствует уровню навыков игрока

Чиксентмихайи с коллегами (Csikszentmihalyi et al., 2005) постулирует три основных условия перехода в состояние потока.

1. *Ясный набор целей*, позволяющий сосредоточиться на текущей задаче.
2. *Баланс* между воспринимаемой сложностью задачи и воспринимаемым человеком уровнем своих навыков, необходимый для того, чтобы увлечься текущей задачей.
3. *Ясная и немедленная обратная связь*, сообщающая человеку о том, как изменить свои действия.

Мимоходом исследователи упоминают еще об одном, дополнительном элементе, помогающем войти в состояние потока: о *соответствии* между целями конкретной задачи и более абстрактными долгосрочными целями.

Игроки стремятся войти в такое желанное для них состояние, а игры часто активно их к этому подталкивают. Как мы видели, игры используют ряд приемов для удовлетворения условий 1, 3 и 4. Контент игры и разнообразие систем предоставляют игрокам огромное разнообразие ясных и достижимых целей на различных уровнях и с разной периодичностью, которые при этом поддерживают друг друга. Также в играх широко присутствует обратная связь, включая механику прогресса, ведущая игрока по ходу игры.

Но остается баланс между вызовом и навыками игрока, которого сами по себе игры так уж легко достичь не могут. Достижение такого баланса до сих пор остается трудной дизайнерской задачей. Чаще всего дизайнеры ограничиваются тем, что предлагают игрокам самим выбирать для себя «уровень сложности», но такой способ нельзя назвать удовлетворительным. Например, игрок не знает, насколько в действительности развиты его навыки, или ему все равно, но тогда он не получает подходящего ему уровня вызова.

В многопользовательских играх проблема меньше. Можно попытаться сделать так, чтобы игра автоматически подбирала игроку противника со схожими навыками – на основе набранных очков или согласно рейтингу Эло как способу примерного измерения мастерства. Этот метод хорошо работает для большого периода времени, но его сложно корректно настроить для новых игроков. Многие многопользовательские игры страдают от слишком крутой «кривой обучаемости», когда новичкам неудачно подбирают соперников. В таких случаях предполагается, что период разочарований – это неизбежная часть процесса обучения.

Достижение потока – центральная составляющая успешного геймплея. Она требует

совместной работы целого набора различных строительных блоков игры – механик и систем, предоставляющих вызов игрокам; систем и контента, дающих им цели; наконец, удовольствие от взаимодействия со всеми этими элементами и получение хорошего опыта благодаря преодолению вызовов, исследованию и экспериментам.

Обучение и усиление вызова

Схожая, но немного другая аргументация представлена в работе Костера «Теория развлечений» (*A Theory of Fun*, 2004). Он утверждает, что удовольствие (или «веселье» – fun, как он его называет) от игры проистекает от обучения, овладения мастерством в том виде деятельности, который представлен в игре, а особенно от овладения навыком *предсказывать исход* внутри игры. Например, это может означать, что игрок учится проходить коварный уровень, понимает, как решить какую-то определенную задачу или следует по сюжету и догадывается, что будет дальше. Кроме того, данное обучение должно происходить в безопасном окружении, где ставки низки, но вызовы все еще интересны, благодаря чему у игрока остается пространство для экспериментов и получения удовольствия в процессе обучения.

Эта модель рассматривает вопрос под иным углом, чем теория потока (отражая скорее интуицию практикующих гейм-дизайнеров в отличие от теоретических рассуждений психологов), но приходит к похожим выводам: удовольствие проистекает от обучения, от понимания, как что-то устроено, и от имеющегося пространства возможностей, позволяющего выяснить это.

Но что происходит, когда игрок наконец-то действительно понимает, как все устроено? Логичный вывод заключается в том, что когда игрок понимает, чего ждать от игры и может предсказывать ее поведение, то без дальнейшего развития игровых систем и контента геймплей неизбежно будет становиться скучнее. В контексте предыдущего рассуждения о циклах и активностях это объясняет, почему нам нужны разные геймплейные циклы с разной периодичностью, а также контентные арки, вносящие разнообразие в них. Как только игрок освоит маленький цикл, тот уже не будет представлять для него секретов, поэтому, чтобы участник не терял увлеченности, их нужно варьировать. Например, можно не только повторять маленькие циклы, но и усиливать их (выставляя против игрока все более мощных врагов), добавлять более крупные и более трудные циклы (как в играх с многочисленными системами) или вводить циклы с совершенно иными вызовами (добавляя долгосрочную нарративную арку).

Основная трудность здесь заключается в том, что знания и навыки игрока постоянно повышаются, поэтому нужно тщательно рассчитывать синхронное усложнение циклов в зависимости от меняющихся навыков игрока. Если вызов усложняется слишком быстро, то игрок испытает раздражение от большой сложности. Одним игрокам нравится такое чувство, другие его ненавидят. И наоборот, если вызов растет медленнее навыков, то игра покажется слишком легкой и скучной. Некоторым игрокам нравится заниматься чем-нибудь легким и «расслабляющим», а других это раздражает. Иногда один и тот же участник испытывает разные чувства по отношению к различным активностям в одной и той же игре – в зависимости от своего настроения или целей (хочется ли ему испытать себя или отдохнуть).

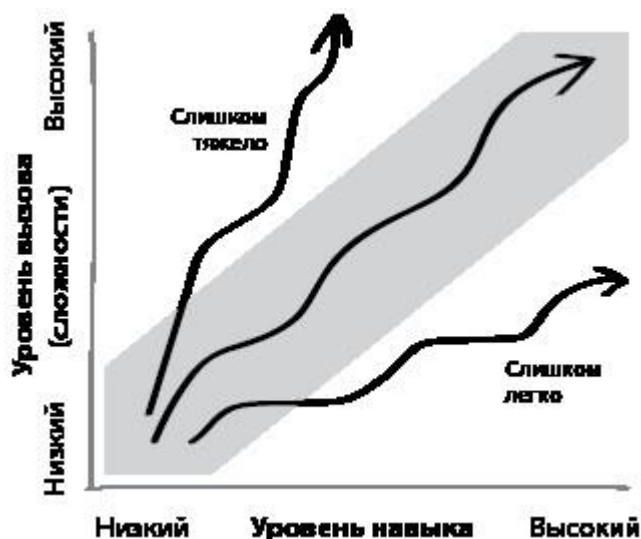


Иллюстрация 5.9. Примеры активностей, выходящих за пределы потока из-за слишком большой сложности или легкости по сравнению с уровнем навыка

Обучение как преодоление неопределенности

Итак, обучение навыкам и мастерство крайне важны для получения удовольствия, а Костер утверждает, что обучение – само по себе удовольствие. Это кажется интуитивно понятным, но с одной оговоркой – не все типы обучения доставляют удовольствие в равной степени. Легко можно привести противоположные примеры, такие как игры, ориентированные на обучение математике или иностранному языку. Хотя активность в них может предоставлять вызов и быть мотивирующей, можно ли утверждать, что обучение математике – это веселая *игра* ?

В своей книге «Неопределенность в играх» (*Uncertainty in Games*, 2013) Костилян утверждает, что одного обучения недостаточно, а одна из уникальных характеристик игр заключается в том, что они представляют собой платформу для *обучения управлению неопределенностью*. Играя в спортивные или киберспортивные игры, мы понемногу учимся контролировать свои движения и внимание. В спорте, как и в *шахматах*, учимся предвидеть действия противника. В гонках или в стратегических играх учимся анализировать и предсказывать поведение сложной механической системы. Эти системы обладают неопределенным поведением, а мы получаем удовольствие, узнавая, как они устроены и как контролировать их в игровом окружении, поддерживающем попытки, неудачи и эксперименты.

В своей классификации Костилян перечисляет следующие основные и второстепенные типы неопределенности.

- *Случайность* : непредсказуемые случайные процессы, такие как бросание костей или тасование карт. Умение определять свои шансы на выигрыш за столом в блек-джек или на попадание на определенную клетку в «*Монополии*» – примеры того, насколько важно иметь представление о случайности. Случайность – очень распространенный вид неопределенности, ее легко включать в разные игровые правила, вводя с ее помощью драматическое напряжение и «встряхивая» предсказуемый геймплей. В главе 3 «Механики» мы обсуждали два типа неопределенности: стационарную (вроде бросания костей) и нестационарную (вроде тасования карт). Они по-разному бросают вызов игроку, проверяя, насколько он понимает распределение вероятностей и насколько умеет предсказывать будущий исход на основании прошлых результатов.

- *Неопределенность навыка* : неумение действовать так, как требуется, и тренировка в целях преодоления этого неумения. Неумение действовать – большой источник неопределенности и отрицательных эмоций для новичков, но по мере того, как они учатся

играть, такая неопределенность уменьшается. Часто ее требуется преодолеть, чтобы освоить какую-то определенную игру или вид спорта. Поскольку в данном случае источник неопределенности заключается в самом игроке, то ее можно повышать или понижать, меняя сложность игр или бросая новые вызовы игрокам. Здесь можно выделить три подтипа.

- *Неопределенность действия* : игрок с трудом выполняет действия, требующие особого навыка или точности. Хороший пример – экшен-игры или виды спорта, требующие тренировки.

- *Неопределенность восприятия* : игрок с трудом понимает или ощущает, что от него требуется. Примером могут служить игры со слишком перегруженными или, наоборот, лаконичными интерфейсами – от стратегий в реальном времени до квестов по поиску спрятанных вещей.

- *Неопределенность решения* : игрок с трудом находит верное решение текущей задачи. Такого рода неопределенность характерна для игр-головоломок – от казуальных кроссвордов до более сложных пазл-квестов.

- *Непредсказуемость игрока* : незнание того, что сделает противник. Это значительный источник неопределенности в многопользовательских играх, потому что участник не знает, что будут делать его враги или союзники, а основной вызов здесь – научиться предсказывать их поведение с целью получения стратегического преимущества. Такая неопределенность встречается как в разнообразных многопользовательских играх, так и в разных видах спорта. О непредсказуемости игрока можно говорить и тогда, когда другой участник – это управляемый компьютером персонаж или игровая система (например, дружественные или враждебные неигровые юниты). Вызов здесь схож: игроку нужно понять, как устроен ИИ, чтобы увеличить свои шансы на успех.

- *Сложность* : неопределенность ситуации, в которой трудно разобраться. Такого рода неопределенность присуща стратегическим играм: *шахматам* , военным играм, экономическим стратегиям и прочим аналогичным. Здесь можно выделить два типа.

- *Аналитическая сложность* : анализ текущего состояния игры подвергает испытанию умственные способности игрока. Это очень распространенное явление для игр с огромным пространством действий и состояний с высоким коэффициентом ветвления, таких как *шахматы* , военные игры или игры с эмерджентным или хаотическим поведением. Аналитически сложные игры заставляют игрока рассчитывать будущие состояния, основываясь на текущем, в результате участник может научиться контролировать такого рода неопределенность экспериментируя, исследуя и постепенно выстраивая умственную модель устройства игры.

- *Скрытая информация* : игрок не обладает всеми данными, необходимыми для принятия верного решения, например, как в стратегических играх с «туманом войны» или в *покере* .

- *Предвосхищение* : попытка спрогнозировать, что произойдет дальше во всей игре. Иногда источником неопределенности может служить просто незнание того, что приготовил для нас гейм-дизайнер. Вызов такого рода неопределенности заключается в том, что игроки понимают, что по прихоти дизайнера дальше в игре может произойти что угодно. Им нравится строить прогнозы о будущем, а затем следить, насколько точными оказались предположения, и размышлять, можно ли было высказать более точные предсказания. Уровень такой неопределенности можно варьировать в разумных пределах, согласуясь с логикой игровой вселенной и предоставляя участникам подходящие намеки о том, как игра будет развиваться в дальнейшем. Здесь можно выделить три подтипа.

- *Предвосхищение нарратива* : игрок не знает, что произойдет дальше по сюжету, поэтому ощущает вызов своей способности быть готовым к переменам. Особенно это свойственно играм, ориентированным на сюжет, в которых траектория игрока намеренно обставляется различными интересными поворотами.

- *Эволюция игры* : игра просто меняется со временем, а вызов состоит в том, чтобы участники сумели к ней приспособиться. Это свойственно таким играм, как *Magic the*

Gathering , социальным вроде *CityVille* и киберспортивным (*Hearthstone*) , где программа неизбежно эволюционирует, а участники должны предвидеть перемены, чтобы не отстать от друзей или соперников.

• *Неопределенность реальной жизни* : игра бросает вызов игрокам тем, что ради прогресса или победы заставляет их приспособить свою реальную жизнь к требованиям программы. Это свойственно играм со значительным влиянием многопользовательских элементов (таким как ММО, киберспортивные или другие многопользовательские соревновательные игры), в которых участники объединяются в гильдии для рейдов, соревнуются между собой по графику и принимают участие в других подобных активностях, влияющих на график реальной жизни игроков.

Различные виды неопределенности используются в качестве вызова участникам, чтобы сделать процесс интересным. Часто в играх одновременно задействованы несколько видов неопределенности. Например, в коллекционной карточной игре *Magic the Gathering* переплетаются между собой различные случайные процессы (карты), скрытая информация, непредсказуемость игрока, высокая аналитическая сложность (в силу сложных механик сражения и ресурсов), а также заметная эволюция игры и элементы неопределенности реальной жизни.

Доминантные стратегии и «решение игры»

Подобно «удовольствию», ощущение вызова тоже субъективно и основывается на предыдущем опыте игрока. Детям нравится играть в «крестики-нолики» , потому что эта игра испытывает их способность к аналитическим предсказаниям. Но для взрослых в ней нет никаких тайн, а лишившись неопределенности, она уже не может доставлять былого удовольствия. Вызов, ставший предсказуемым и привычным, теряет свою привлекательность.

Удовольствие от вызова находится как раз между полной непредсказуемостью и полной предсказуемостью. Если игра совершенно непредсказуема, как *рулетка* или другие азартные игры, то это сокращает удовольствие от геймплея (хотя игроки могут продолжать играть по другим причинам, таким как азарт и риск). Но когда игра предсказуема, она также теряет свою привлекательность – если нам заранее известен исход, то зачем играть?

Чтобы уменьшить предсказуемость, гейм-дизайнеры стараются выявить в играх *доминантные стратегии* , дающие очевидно более выгодные результаты по сравнению с другими. Простой пример доминантной стратегии в «крестиках-ноликах» – первым шагом заполнить центральную клетку. Такой ход гарантированно позволяет дать отпор любым последующим ходам противника, поэтому каждая игра может закончиться победой или ничьей, но не поражением.

Доминантные стратегии бывают и не столь очевидными. Например, когда игра *Civilization V* только вышла, у каждой цивилизации имелись свои уникальные сильные стороны, но у французов они были особенно сильны и позволяли быстрее расширять границы по сравнению с другими. Играя за Францию и с ранних этапов сосредотачиваясь на определенных усовершенствованиях, игрок легко мог обогнать других к середине игры, что приводило к легкой победе. Как и ожидалось, такие несбалансированные настройки «понерфили» в последующих дополнениях.

Доминантные стратегии влияют на удовольствие от вызова, но их трудно обнаружить аналитически. Для этого, как правило, приходится проводить много тестов до релиза, а также добавлять аналитику в игры после него и следить за определенными схемами игры, приводящими к неожиданным результатам или диспропорциям.

Циклы и вызовы

Мелкие циклы не всегда предоставляют небольшие вызовы. Например, в экшен-играх

маленькие циклы с самого начала представляют довольно большую сложность. Управлять автомобилем и не врезаться в препятствия, стрелять по кораблю инопланетян и уворачиваться от их ракет – для этого требуется много практики. Для аркадных игр (от *Space Invaders* или *Asteroids* до недавней *Flappy Bird*) высокая сложность микроциклов стала своеобразной визитной карточкой. По сравнению с ними более крупные циклы данных игр легче (или их не существует), а участвуя в них опытные и умелые игроки ставят себе более абстрактные цели (например, состязание с другими).

Противоположная ситуация наблюдается в стратегических или настольных играх. Чем более стратегически ориентирована игра, тем легче микроцикл – задача передвижения фигур по доске может казаться скучной или требовать некоторых расчетов, но обычно она не воспринимается как бросающая вызов. Большой вызов наблюдается в среднесрочной и долгосрочной стратегиях – нужно знать, чем ходить, как распределять ресурсы, как постепенно двигаться к победе.

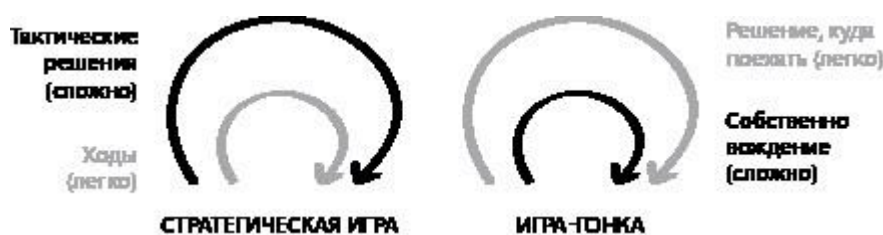


Иллюстрация 5.10. В некоторых играх маленькие циклы бывают сложнее больших, в других же играх ситуация противоположная (Koster, 2012)

В играх часто совмещаются два подхода. Например, в командных тактических шутерах от первого лица, таких как серия *Call of Duty*, смешиваются микроциклы, требующие физической ловкости и быстроты реакции, и более крупные циклы, связанные с развитием тактической ситуации для всего отряда, а также изрядная доля метаигры.

На иллюстрации 5.10 показана данная ситуация с помощью «луковичных» схем на основе приведенного Костером примера (Koster, 2012).

Внешняя мотивация: работа и вознаграждение

До сих пор мы сосредотачивались на внутренней мотивации, связанной с чувством удовольствия от деятельности. В противоположность этому, когда движущей силой игрока является *исход* действий (такой как награда или одобрение, а не сама деятельность), можно утверждать, что это внешняя мотивация. Она распространена в играх, ее поддерживают механики прогрессии и другие элементы гейм-дизайна.

Опишем несколько способов внешней мотивации игрока и как игры ее поддерживают, а также попытаемся связать между собой внутреннюю и внешнюю мотивации.

Прогрессия и награды

Очень часто игры вознаграждают игрока за участие различными достижениями и местом в таблице лидеров. Во многих платформах (включая консоли или платформу для настольных компьютеров Steam) такие системы наград предусмотрены изначально, поэтому участник с легкостью может просматривать свои игровые трофеи и списки лидеров и даже демонстрировать их другим игрокам.

Такие награды относятся к семейству *механик прогресса*. Эти механики охватывают различные способы получения игроком обратной связи и информации о прогрессе. Мы уже обсуждали в главе 3 «Механики», что это следующие элементы.

- *Счет* или *очки опыта* (увеличиваются по мере выполнения различных целей).

- *Уровни* (увеличиваются, когда игрок достигает какой-то значимой вехи в игре).
- *Достижения* (выдаются за выполнение каких-то специфических целей).
- *Списки лидеров* (показывают счет игрока, уровни и т. д. по сравнению с показателями других игроков).

Механики прогресса могут предоставлять игрокам различные внешние цели. Они очень широко используются в играх, потому что служат дополнительной мотивацией. Когда внутренняя мотивация участника со временем идет на спад, удержаться в игре ему помогает дополнительная «морковка», да и сами по себе награды приносят удовольствие на метауровне. Например, высокий счет может стать поводом похвастаться перед друзьями.

Некоторые механики вдобавок к своей основной роли предоставляют еще и награды. Например, накопление золота, ресурсов или юнитов может служить средством обратной связи и информирования о прогрессе. Накопленные ресурсы в данном случае воспринимаются как награда – независимо от того, как они будут использованы в дальнейшем. Аналогичным образом развивающийся сюжет игры может быть интересным сам по себе и давать игроку представление о его прогрессе.

Вместе с тем важно отметить, что самих по себе внешних наград недостаточно, чтобы игра начала доставлять удовольствие. Они предоставляют информацию, но игроку должны нравиться и сами игровые действия.

Режимы подкрепления

Предположим, мы решили предоставить игроку внешние награды, но нужно подумать и о том, *когда* их предоставлять. Каков наилучший способ вознаграждения игрока? С какой частотой оно должно производиться, на основании какой степени усилий? Какой способ игрок сочтет самым привлекательным?

Для ответа на данные вопросы можно обратиться к психологическим исследованиям вознаграждения и мотивации в целом, а не только в контексте игр. Широко известна концепция *оперантного научения* (обусловливания), то есть использование различных видов предсказуемых наград (или наказаний) для формирования какого-либо поведения¹³.

Например, в классических экспериментах исследователи учили голубей клевать отметку, за что давали им награду в виде пищи, а затем варьировали частоту наград и наблюдали, как от этого менялась мотивация голубей и их настойчивость. Термин *режим подкрепления* означает определенный график предоставления наград со временем на основе выбранных принципов (в качестве простого примера можно предоставлять одну порцию еды за пять поклевков или одну порцию еды каждую минуту).

Широко известно, что животных можно научить выполнять несложные трюки за еду, но одно из открытий заключалось в том, что различные варианты режима подкрепления приводили к очень разным результатам, а некоторые оказывались значительно удачнее других. Другое интересное открытие было связано с тем, что, делая награды более редкими, субъектов можно было научить выполнять большее количество работы за одинаковое количество вознаграждения (вплоть до того момента, когда они начинали терять интерес).

Расцвет теории обусловливания пришелся на середину XX века, но затем она подверглась *всесторонней* критике как в психологии, так и в педагогике как редуционистский и слишком узкий взгляд на обучение людей или человеческую мотивацию. Такая критика оправдана. Поведение людей гораздо сложнее, чем просто реакция на награду. Мы – существа, обладающие разумом, поэтому руководствуемся

¹³ Существуют и другие виды обусловливания. Например, *классическое обусловливание* (классический метод формирования рефлексов), когда одновременно происходящие явления ассоциируются друг с другом в мозгу субъекта – вспомним известные эксперименты с собаками, начинавшими ассоциировать звук колокольчика с едой. Здесь мы их не рассматриваем, но эти примеры подробно описаны в работе (Hodent, 2017).

сложной мотивацией. Но, несмотря на это, на наше поведение тоже можно влиять с помощью наград, пусть и ограниченным образом.

В играх очень широко используются режимы подкрепления и наград, хотя взгляды на них бывают противоречивыми. Их критикуют за то, что они заставляют игроков стремиться к внешним наградам в ущерб внутренней мотивации (Hodent, 2017, 61–66), или за то, что слишком манипулятивны. Некоторые критики выступают с точки зрения морали, обвиняя данные методы в эксплуатации человеческих слабостей ради прибылей корпораций и сравнивают их с азартным бизнесом (Juul, 2010).

Тем не менее практически все видеоигры используют внешние награды, чаще всего с режимом подкрепления с вариативным соотношением (описанным ниже), потому что они очень хорошо мотивируют игрока. Полезно знать принципы их работы и ограничения, чтобы использовать во благо.

Типы режимов

Были изучены многие типы режимов подкрепления, но мы затронем лишь самые основные.

- *Режим непрерывного подкрепления* : субъект получает награду за каждое действие непосредственно.
- *С постоянным интервалом* : субъект получает награду каждые n секунд, пока он выполняет действия.
- *С постоянным соотношением* : субъект получает награду за каждые n действий.
- *С вариативным интервалом* : субъект получает награду через случайное количество времени, пока он выполняет действия.
- *С вариативным соотношением* : субъект получает награду за случайное количество действий.
- Эти режимы можно сравнивать по двум критериям.
- *Частота ответов (реакций)* : сколько действий выполняется за единицу времени или насколько часто субъект «работает» ради получения награды.
- *Сопrotивляемость угасанию* : насколько долго субъект будет выполнять действия в ожидании награды, даже если награды больше не поступают.

Результаты на удивление стабильны в самых разных экспериментах и с различными субъектами. Реакцию поведения субъектов-животных можно обобщить в графиках, показанных на иллюстрации 5.11. Линии соответствуют действиям субъектов, а точки – наградам.

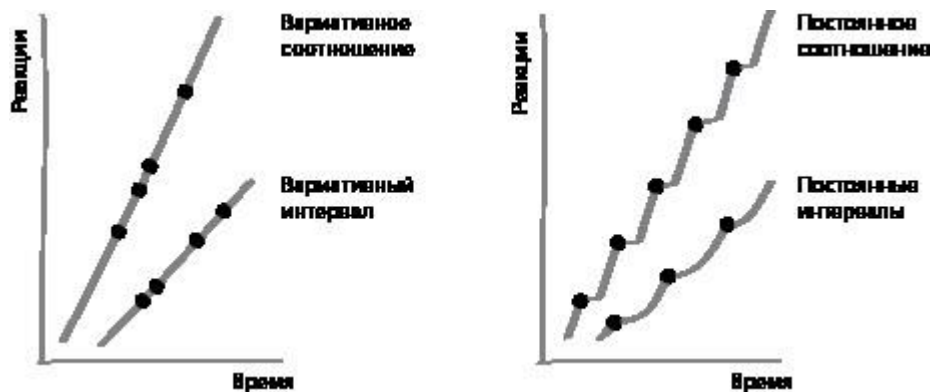


Иллюстрация 5.11. Частота ответов при разных режимах подкрепления, приведенных по Walker (1975, 81). Линии соответствуют действиям субъектов; точки соответствуют наградам

Опишем подробнее действия этих режимов и начнем с самого эффективного.

С вариативным соотношением : представим, что голубя награждают за клевание, но через разное количество действий. В первый раз ему могут предоставить награду за десять поклевков, в следующий раз – за двадцать, затем через пять и так далее. Такой режим дает самый высокий график ответов (голубь работает усерднее всего) и самую низкую сопротивляемость угасанию (голубь медленнее сдается) по сравнению с другими режимами. Он вознаграждает активные действия субъекта и учит его не сдаваться слишком быстро, если награды какое-то время не поступают¹⁴.

С вариативным интервалом : голубя награждают за клевание, но через разные периоды времени. Допустим, в первый раз его награждают через десять секунд, потом через пятнадцать или двадцать. Награда в данном случае зависит от случайного количества времени, а не от работы, но режим также демонстрирует низкую сопротивляемость угасанию (голубь не сдается легко), а субъект выполняет меньше работы за единицу времени (работает не так усердно), потому что награда не зависит от уровня активности.

С постоянным соотношением : голубь получает вознаграждение через каждые двадцать поклевков. Когда награды распределены равномерно по количеству действий, уровень активности неплох, но угасание происходит быстрее (голубь сдается, когда награды перестают поступать). Также существует вероятность, что активность снизится сразу же после получения награды.

С постоянным интервалом : голубь получает вознаграждение через каждые двадцать секунд. Когда награды распределены равномерно по времени, при низком подкреплении угасание происходит быстрее (голубь не работает усердно). Активность снижается после получения награды, но увеличивается ближе к моменту ее получения.

Непрерывное подкрепление : голубь получает награду за каждую поклевку. Такой режим (не показанный на графике) демонстрирует самое быстрое угасание и самый низкий уровень подкрепления, потому что субъекты быстро насыщаются и при прекращении вознаграждения теряют интерес.

Хотя самые известные результаты получены благодаря исследованиям животных, они в той или иной форме были воспроизведены в различных испытаниях с участием людей. Ожидание непредсказуемых наград в будущем оказывает схожий эффект как на животных, так и на людей, а вариативные режимы подкрепления учат субъектов не сдаваться, даже если награда не поступает.

Примеры из игр

Чтобы рассмотреть, как режимы подкрепления действуют в играх, вернемся к нашему постоянному примеру «данжен-кроулера» – серии игр *Diablo* .

Основной цикл состоит в том, что мы ходим и убиваем монстров, собираем добычу, возвращаемся в лагерь и продаем ее. Добыча собирается после сражения. Монстры при убийстве оставляют случайную добычу и случайное количество золота. На уровнях также присутствуют сундучки со случайной добычей, есть и другие ее виды. Более сильные монстры оставляют более желаемую добычу. Как монстры, так и сундучки иногда дают редкие, коллекционные вещи.

Если счесть сражение «работой» (успешное нахождение и убийство монстров), то добыча – это награда за нее. Но в каком режиме она предоставляется?

¹⁴ Существует еще и режим *со случайным соотношением* , при котором каждое действие может вознаграждаться отдельно по случайному принципу – например, в зависимости от броска костей, а не от случайного общего количества действий. Этот режим обычно наблюдается в игорном бизнесе, например в игровых автоматах (Schüll, 2012, 344). Но различие очень небольшое, а для наших целей два режима достаточно схожи, так что будем использовать термин «с вариативным соотношением» как сокращение для обоих.

Для убийства монстров требуется случайное количество работы, а после ее завершения юниты дают случайное количество наград, пропорциональное их уровню сложности. Это режим с вариативным соотношением: игрок получает награду согласно системе с элементами случайности, детали которой ему неизвестны.

Иногда монстры дают редкие и гораздо более дорогие коллекционные вещи, что похоже на неожиданный крупный выигрыш в слот-машине. Это второй режим наград, существующий поверх первого. Данный режим тоже с вариативным соотношением, как и первый, хотя и с более дорогой наградой за большее количество работы.

Но сражения – не единственный способ получить сокровище: можно также исследовать уровни и находить клад. В различных локациях скрыты сундучки, содержащие иные по сравнению с монстрами типы добычи. Для ее получения нужно эти сундучки найти, что, в свою очередь, требует риска (например, можно встретиться с более опасными монстрами) и дополнительных навыков или инструментов для открывания сундучков. Как и в случае с монстрами, награды выдаются согласно режиму с вариативным соотношением, но за иной вид деятельности.

Так же, как и монстры, некоторые сундучки содержат очень ценные вещи, поэтому и существует второй режим вознаграждения – более медленный, но с гораздо более ценными призами.

В целом в таком «данжен-кроулере» присутствуют четыре разных режима вознаграждения.

Награда за сражение

- Вариативное соотношение с высокой периодичностью.
- Вариативное соотношение с низкой периодичностью.

Награда за исследование и риск

- Вариативное соотношение с высокой периодичностью.
- Вариативное соотношение с низкой периодичностью.

Любопытно, что циклы сражений и циклы исследований имеют по несколько режимов подкрепления, настроенных по-разному, чтобы игра казалась интересной и непредсказуемой. Это великолепный пример цепочек множества существующих одновременно режимов подкрепления, награждающих игрока за вариативное поведение и за повышение навыков во всех этих видах деятельности.

Изменение рабочей нагрузки

Еще один аспект подкрепляемого поведения заключается в том, что, будучи усвоенным, оно становится основой для новых видов поведения – как для более сложной работы, так и для более высокой рабочей нагрузки с тем же видом награды.

Такой эффект интуитивно известен учителям и тренерам. Если научить учеников чему-либо простому и последовательно награждать их за такое поведение, то потом можно при прежнем вознаграждении повышать интенсивность заданий (например, требующих большей выносливости) или сделать их строительными блоками для более сложной деятельности (добавляя дополнительные условия).

В играх данное явление тоже распространено. Наиболее типичный пример – это *кривые уровней*, то есть формулы для присвоения очков опыта (XP), соответствующие повышению уровня персонажа. Во многих играх игрок получает очки опыта за выполнение определенных действий – например, за убийство монстров или за исследование зон и сбор предметов. Персонаж может достигать уровня, повышающегося за достижение очередной «вехи» (получения определенного количества XP), после чего становятся доступными новые способности и награды. Очень часто эти «вехи» распределены не равномерно, а требуют все

большого количества ХР.

«Лут-боксы». В недавнее время особую актуальность приобрел вопрос так называемых лут-боксов, представляющих собой «сундучки с сокровищами», которые игрок покупает за реальные деньги или иногда зарабатывает во время игры. Каждый лут-букс содержит некоторое количество случайных наград – обычно сочетание объектов различной ценности с шансом получения редкой или очень ценной награды.

Такие сундучки критиковали за то, что они опасно напоминают азартные игры. Во-первых, лут-боксы – идеальный пример режима подкрепления со случайным соотношением, при котором субъект выполняет «работу» и по случайному принципу получает одну из редких вещей, а это поведение с очень высоким подкреплением и низкой сопротивляемостью к угасанию. Во-вторых, «работа» обычно сводится к тому, чтобы тратить реальные деньги за каждый шанс открыть новый лут-букс, что может привести к чрезмерным расходам и другим проблемам, ассоциируемым с навязчивым поведением.

Многие страны уже исследуют данный вопрос или регулируют продажу лут-боксов из-за их схожести с азартными играми. Как это отразится на использовании режима подкрепления со случайным соотношением, пока еще неясно.

Рассмотрим для примера соотношение очков опыта и уровней в *Diablo III*, в упрощенном виде представленное в таблице 5.2¹⁵. В ней отображена очень типичная, резко возрастающая кривая уровней, похожая на аналогичные кривые других игр.

Уровень персонажа	Количество ХР, необходимое для достижения уровня
1	0
2	280
3	2700
4	4500
5	6600
...	...
10	19 200
...	...
20	57 200
...	...
30	115 200
...	...
40	420 000
...	...
50	2 080 000

Таблица 5.2. Таблица соотношения ХР и уровней для *Diablo III*

Источник : DiabloWiki (2018)

При такой кривой опыта по достижении нового уровня количество ХР, необходимое для достижения следующего этапа, увеличивается. Это оказывает на игрока любопытный эффект – ему приходится либо работать усерднее и чаще для получения следующей награды, либо работать «умнее», чтобы получать больше ХР за то же количество работы (например, создавать лучшее оружие или находить более «прибыльных» врагов). Поэтому в одних случаях участнику хочется играть чаще, а в других увеличившийся объем работы демотивирует его. Это зависит исключительно от сочетания настроек наград и собственной мотивации игрока.

¹⁵ Хотя именно эти конкретные данные были получены благодаря сообществу игроков и в отношении конкретных значений могут содержать неточности, в общем случае для таких игр характерна именно такая кривая уровней.

Смежная тема: геймификация

Геймификация тесно связана с внешними наградами. Это использование механики прогресса в сферах, не имеющих непосредственного отношения к играм. Например, форумы обсуждений вроде Stack Overflow поощряют участников за их вклад очками, значками и со временем, по достижении определенного уровня, даже предоставлением новых возможностей (пользователь должен обладать достаточно высоким уровнем, чтобы задавать новые вопросы).

Поскольку геймификация предоставляет внешние награды, то она определенно может служить «морковкой» для формирования желательного поведения, но с некоторыми оговорками. Во-первых, награда должна быть значимой для участника и представлять для него ценность. Достижение нового уровня и получение новой возможности – более значимое событие, чем достижение нового уровня и просто увеличение очков в профиле пользователя. Во-вторых, активность сама по себе должна быть интересной. Просто получение очков за какое-то действие неинтересно и не кажется значимым. Нудная работа остается нудной, даже если за нее даются очки, а попадание в списки лидеров в таком случае – просто признание того, что ты выполнял скучную работу усерднее других. Сами по себе награды не служат мотивацией.

Эвристические принципы дизайна геймплейных циклов

Предположим, что у нас имеется какая-то идея игры, и мы приблизительно представляем, чем игрок должен в ней заниматься и какой опыт должен получать. Теперь необходимо разработать геймплейные циклы, побуждающие его выполнять такие действия. С чего же начать?

В конце главы 4 «Системы» мы поговорили о *пользовательских историях* как об ориентирах в дизайне систем. Методика, отталкивающаяся от них, очень распространена и полезна, ею можно воспользоваться для дизайна геймплейных циклов.

От пользовательских историй до геймплейных циклов

Вспомним наш пример игры в жанре tower defence. Для начала можно спросить себя: «Что игрок будет делать снова и снова?» Можно ли выявить базовый цикл на основе нарративного описания? Попробуем описать действия игрока следующим образом:

«Игрок строит защитные сооружения, затем порождаются волны крипов, атакующих его базу. Убитые крипы дают монеты, которые можно использовать для починки оборонительных сооружений и постройки улучшенных, прежде чем в атаку направится очередная, более мощная волна врагов».

Это хороший уровень описания: действия игрока заключаются в постройке, защите, а затем в покупке, усовершенствовании или починке. Такой цикл уже сам по себе кажется интересным, на его основе можно разработать хороший базовый цикл, а тестировщики подтвердят, достаточно ли он увлекательный.

Исходя из этого описания, можно представить себе циклы поменьше, например:

«Во время защиты крипы роняют монеты. Игрок должен обращать на них внимание и тут же подбирать, иначе монеты исчезнут с поля и выйдут из игры».

Это описание цикла, ориентированного на быстроту рефлексов и действий. Интересен ли он, доставляет ли удовольствие? Гейм-дизайнер должен решить это на основании своего опыта или игрового тестирования; при необходимости, если этот цикл работает не так, как

задумывалось, нужно будет заменить его на другой.

Схожим образом можно вообразить и более крупные циклы, например:

«С каждым разом волна крипов становится все более мощной и многочисленной. Но игрок может тратить большое количество монет на заклинания, определяющие типы порождаемых крипов. В результате действий появляются более слабые и медленные крипы или крипы с низкой частотой атаки».

Это уже описание потенциального долгосрочного цикла, позволяющего игроку разрабатывать более длительные стратегии. Если, например, игрок построит много атакующих башен, то ему будет выгодно потратить деньги на заклинания, делающие крипов слабее, чтобы ярче проявилась специализация башен. А поскольку стоимость монет высока, то необходимо разработать долгосрочный план и накапливать деньги, вместо того чтобы тратить их на постройку дополнительных башен или починку.

Игровое тестирование циклов

Как только игровой цикл составлен «на бумаге», очень важно как можно быстрее реализовать его на практике и посмотреть, как он работает. Реализовать цикл можно как в *самой игре* (внутриигровые прототипы), так и в *отдельном прототипе* – в зависимости от того, насколько этот цикл взаимодействует с другими системами и можно ли его отделить от них.

Необходимость максимально быстрой реализации проистекает из того, что в реальной жизни геймплейные циклы часто ощущаются не так, как «на бумаге». Нарративное описание легко может породить у читателя иное представление о цикле и завышенные ожидания, не оправдывающиеся на практике. Поэтому очень важно как можно раньше получить реальное представление о геймплее и цикле, прежде чем мы начнем добавлять их в игру. Опытные гейм-дизайнеры иногда могут представить, как будет восприниматься цикл, исходя из своего опыта, но даже в этом случае ничто не заменит реального рабочего прототипа.

Краткий итог

Этим рассуждением о геймплейных циклах мы готовы завершить главу, посвященную опыту игрока. Ключевые ее положения таковы.

- Мы рассматриваем *геймплей* в свете динамического опыта игроков, взаимодействующих с механиками игры и друг с другом, с учетом того что это взаимодействие со временем эволюционирует.

- Основная единица анализа – *геймплейный цикл*, представляющий собой постоянно повторяемые игроком действия (активность). В типичном случае игры состоят из разных активностей с различными вызовами и точками принятия решения. *Быстрый* или *маленький* геймплейный цикл требует частого принятия решений и повышенного внимания по сравнению с *медленным* или *крупным*, в котором решения приходится принимать реже. *Основной цикл* – самый маленький значимый и доставляющий удовольствие в определенный промежуток времени, но обычно это не то же самое, что самый быстрый цикл.

- В играх часто наблюдаются разные *уровни циклов* с различной периодичностью, заставляющие игрока принимать решения на разной временной шкале. Для визуализации этих уровней используются *концентрические схемы* («луковичные диаграммы»).

- Выполнять данные циклы игроков побуждают различные виды мотивации. В широком смысле мотивация бывает внутренней и внешней.

- Состояние *потока* и «потери себя в игре» – важный внутренний мотивирующий фактор, а игры очень хорошо приспособлены для порождения такого чувства, поскольку предоставляют игроку различные вызовы, точки принятия решений и обратную связь.

Другие виды внутренней мотивации – овладение навыками и обучение тому, как преодолевать неопределенность в игре.

- Источником внешней мотивации игрока служат награды, такие как *механики прогресса* (повышение уровней, предоставление наград, повышение счета). Отношения между наградами и поведением хорошо изучены в психологии, а концепция *режима подкрепления* помогает понять, какие внешние награды обуславливают то или иное поведение в геймплейных циклах.

- Поскольку геймплейные циклы возникают в результате взаимодействия игроков с механиками и системами, дизайн этих элементов неизбежно связан между собой. Как и в случае с дизайном систем, можно начать с пользовательских историй, проанализировав и поняв, какие циклы взаимодействия и системы из них следуют.

Геймплейные циклы охватывают разные временные масштабы – от секунд и минут до часов и более. В следующей главе особое внимание будет уделено долгосрочным эффектам и тем структурам, которые проявляются, когда мы рассматриваем игру целиком – от начала до конца, а также повторным прохождениям с многократными сессиями на протяжении продолжительного промежутка времени.

Дополнительная литература

Геймплейные циклы

Тема геймплейных циклов часто излагается в рамках темы систем, но более подробные рассуждения можно найти в книге «Продвинутый гейм-дизайн» Селлерса (*Advanced Game Design*, Sellers, 2017).

В гейм-дизайне хорошо известна концепция геймплейных циклов как отдельного явления. Возможно, самое раннее упоминание их встречается в статье «Формальные абстрактные инструменты дизайна» Черча (*Formal Abstract Design Tools*, Church, 1999), а самым первым «луковичные диаграммы» использовал Уилл Райт в своей лекции «Уроки гейм-дизайна» (Wright, 2003, начиная с отметки 13:30). О связи между циклами и неповторяющимися действиями стоит почитать относительно недавнее рассуждение Кука «Циклы и арки» (*Loops and Arcs*, Cook, 2012).

Мотивация

Тем, кто интересуется широкой темой психологии игрока, особенно рекомендованы две книги, ориентированные как раз на игры – с многочисленными практическими примерами и советами: *Getting Gamers* (Madigan, 2015) и *The Gamer's Brain* (Hodent, 2017).

Рассуждая о мотивации игрока мы мимоходом упомянули, что внутренняя и внешняя мотивация – это два аспекта, которые можно сочетать, но это не бинарные противоположности. Читателям, интересующимся психологической перспективой, хорошим знакомством с ней могут послужить исследования теории самоопределения – например, работа Райана и Деси (Ryan and Deci, 2000).

Что касается именно внутренней мотивации, то сильное влияние на гейм-дизайн продолжает оказывать теория потока. Краткий ее обзор можно найти в работе «Поток» (Flow, Csikszentmihalyi, Abuhamdeh and Nakamura, 2005) или в более научно-популярном варианте – книге «Поток: психология оптимального переживания» Чиксентмихайи (*Flow: The Psychology of Optimal Experience*, Csikszentmihalyi, 1990).

Что касается темы внешней мотивации и наград, неплохим знакомством с ней послужат многочисленные учебники психологии, великолепные образцы которых – бесплатная онлайн-версия книги «Обучение и подкрепление» Уокера (*Learning and Reinforcement*,

Waker, 1975) и уже вышеназванная работа Ходент (Hodent, 2017), во вступлении к которой упоминаются как раз игры. Кроме них, хороший анализ внешних наград с точки зрения азартных игр, потенциально вызывающих привыкание, содержится в книге «Зависимость согласно дизайну» Шулля (*Addiction by Design*, Schüll, 2012).

Наконец, дизайнерская теория «веселья» и получения внутреннего удовольствия от обучения изложена в книге Костера «Теория удовольствия» (*A Theory of Fun*, Koster, 2004). Его же книга «Теория удовольствия: десять лет спустя» (*Theory of Fun: 10 Years Later*, Koster, 2012) представляет собой ретроспективу изложенных в первой книге идей с анализом уроков, которые были усвоены после первоначальной публикации.

Упражнения для индивидуального выполнения

5.1 Геймплейные циклы

Выберите хорошо известную вам игру, компьютерную или физическую.

а. Составьте концентрическую схему («луковичную диаграмму») этой игры. Выделите как можно больше циклов активности с различной периодичностью.

б. Каков самый маленький цикл (цикл с самой короткой периодичностью), в котором игроки снова и снова выполняют одни и те же действия? Каков базовый цикл, то есть самый маленький, от которого можно получать удовольствие даже при отсутствии более крупных? Базовый цикл – это тот же, что и самый маленький? Или это разные циклы?

5.2 Циклы и системы

Проанализируйте еще раз игру из упражнения 5.1. Какие механики или системы включают в себя разные циклы? Имеются ли у разных циклов общие системы?

5.3 Поток

Опишите свой опыт «нахождения в потоке», когда вы погружались в игру на несколько минут или на несколько часов настолько, что «терялись» в ней.

а. Что это была за игра и какие действия в ней показались вам настолько захватывающими?

б. Как игре удалось завладеть вашим вниманием? Перечислите три основных условия вхождения в состояние потока и опишите, как они были отражены в игре.

5.4 Удовольствие от обучения

Опишите две известные вам игры: одну с маленькими циклами, трудными для освоения, другую – в которой труднее было освоить более крупные циклы. Для обеих игр опишите, как менялось получаемое от игры удовольствие, когда вы обучились «легким» циклам. Было ли в оставшихся циклах достаточно вызовов, чтобы получать от них удовольствие?

5.5 Внешние награды

Опишите известную вам игру, которая успешно использует внешние награды согласно определенному режиму подкрепления. За какие действия в ней полагаются награды? Какой при этом используется режим подкрепления? Увеличивается ли сложность заданий по мере прохождения игры?

Упражнения для выполнения в группе

G5.1 Обнаружение и изменение циклов в настольной игре

В этом упражнении мы выявим игровые циклы настольной игры *Settlers of Catan* и постараемся модифицировать их. Предполагается, что у вас есть доступ к данной настольной

игре. Вместо нее преподаватель может предложить другую.

а. Сыграйте в игру *Settlers of Catan* по меньшей мере один раз, чтобы напомнить себе ее геймплей. Затем выделите геймплейные циклы, перечислив различные выполняемые в игре действия (например, торговлю картами ресурсов, постройку поселений, передвижение разбойников и т. д.) и обозначив периодичность этих действий. Каждый участник в начале должен составить отдельный список. Затем все сравнивают свои списки и составляют окончательный вариант для всей группы.

б. Теперь определите самые долгие и менее частотные циклы и удалите их из игры, изменив правила или игровые элементы. Например, можно удалить фишку разбойников и карты солдат, убрать способность строить города и прочие редкие элементы. Протестируйте данный вариант по меньшей мере один раз.

в. Опишите, как эти изменения повлияли на ваши впечатления от геймплея. Ощущается ли видоизмененная игра такой же, как оригинал? Если нет, то как она отличается?

г. Попытайтесь «восстановить» игру, заменив циклы большей периодичности другими по вашему выбору. Для каждого удаленного вами элемента подберите замену, которая будет действовать в той же временной шкале. Протестируйте этот вариант по меньшей мере один раз.

д. Опишите свои ощущения от игры с новыми правилами по сравнению с предыдущим вариантом и изначальной игрой.

Для завершения упражнения опишите подробно:

е. Циклы, которые вы выделили в начале.

ж. Элементы, которые вы удалили. Как это повлияло на геймплей.

з. Элементы, которые вы добавили. Как это повлияло на геймплей.

6 Макроструктура

В предыдущих главах мы рассматривали геймплей и рассуждали о том, что, начиная с некоторых идей об опыте игрока, в дальнейшем реализуем задуманный геймплей с помощью механик и систем в качестве строительных блоков. В процессе этого мы познакомились с некоторыми инструментами *структурирования* геймплея – например, с концепцией происходящих одновременно геймплейных циклов разной периодичности или с идеей циклов обратной связи, расходящихся или сходящихся со временем.

В данной главе мы обсудим структуру геймплея в еще более крупном масштабе. Начнем с обзора общей структуры *всей игры* как арки контента – с того, как опыт игрока меняется с самого начала, как в середине игры усиливаются вызов и сложность, как в конце игра подходит к своему разрешению (если у нее есть конец). А поскольку такого рода структура обычно переплетена с сюжетным вымыслом, сначала рассмотрим нарративную составляющую и обсудим арки контента с точки зрения траекторий напряжения и интереса. Также отдельно разберем нарративную (повествовательную) структуру и подробнее поговорим о специфических схемах – от линейного до нелинейного повествования, показав, как они взаимодействуют с системами (например, система заданий).

Наконец, мы еще больше увеличим масштаб и исследуем структуру опыта вне отдельных игровых сессий, рассмотрев, как игроки на таком временном промежутке играют в метаигру (совершенствуют свои стратегии, меняют состав команд, обсуждают друг с другом процесс, обмениваются идеями и ресурсами) – то есть как они взаимодействуют между играми.

Мотивирующий пример: «Ведьмак»

«Ведьмак» (*The Witcher*) – серия экшен-RPG (ролевых игр), в которых игрок играет

за Геральта из Ривии – ведьмака со сверхчеловеческими способностями, странствующего по миру и в качестве наемника выполняющего различные задания. В игре «Ведьмак-3» Геральт пытается найти и спасти свою приемную дочь Цири, преодолевая по пути различные трудности в разных воюющих между собой королевствах.



Иллюстрация 6.1. Скриншот из игры «Ведьмак-3»

«Ведьмак-3» в высшей степени структурированная и ориентированная на сюжет RPG. Игра начинается с конфликта: приходит весть о том, что Цири угрожает опасность, ее преследуют некие потусторонние силы, а когда игрок пускается на поиски с целью спасения героини, напряжение только усиливается. Эти поиски – основная сюжетная линия игры, но не единственная, а побочные истории добавляют больше подробностей и колорита игровому миру – вместе с вызовами и дополнительными возможностями для игрока.

Как часто бывает в экшен-RPG, личность главного героя уже определена заранее. Предыстория Геральта, его место в мире, отношения с Цири и другими людьми – все это заранее прописано. Игрок вживается в данную историю и *играет роль*, представляя себе, каково это – быть Геральтом, живущим в фантастическом мире. Основной сюжет и побочные истории написаны, спроектированы, реализованы и размещены в мире, где игрок их открывает. Обычно они проявляются в виде сочетания кат-сцен, сюжетных заданий и вызовов. Кроме явно выраженных сюжетных элементов, ролевые игры вроде «Ведьмак а» обычно включают в себя различные сюжетно-ориентированные системы вроде квестов и крафтинга, поддерживающие продвижение игрока по повествовательной линии.

Кроме основного сюжета и побочных историй, игра «Ведьмак-3» отличается еще и тем, что претендует на *открытость мира*. Игроку не обязательно придерживаться сюжетной линии, а в мире существует множество интересных элементов, не связанных с основным сюжетом – различные места, которые можно посещать; персонажи, которым можно помогать; подземелья, которые можно исследовать и т. д. Игрок волен игнорировать сюжет и странствовать по игровому миру, открывая эти элементы, которые сами по себе интересны и доставляют удовольствие. Открытость мира усиливает чувство свободы и осознание того, что ты сам контролируешь свое поведение, отчего игра кажется гораздо более живой и позволяет игроку «создавать свою собственную историю».

Художественный вымысел в игре

В нашем примере мы упомянули игровые вымысел, сюжет и фантазию. В повседневной речи их можно использовать как синонимы, но в гейм-дизайне сложилось так, что ими обозначают весьма разные понятия.

В большинстве игр присутствует *художественный вымысел* об игровом мире и о роли

игрока в игре. Это тематика, сеттинг игры, возможно, предыстория мира, он объясняет мотивацию действий игрока в процессе игры. Художественный вымысел помогает понять, почему мир устроен именно таким образом.

Вымысел может отличаться по сложности и по степени авторского контроля. На одном конце спектра мы имеем игры вроде RPG, в которых существует очень явно выраженная *повествовательная* составляющая (*внедренный нарратив*), придающая игроку весьма определенную личность и роль, помещающая его в конкретную ситуацию, раскрывающая сюжет по мере прохождения игры.

В другой, менее сложной области спектра мы имеем игры, просто задающие какую-то общую *фантазию*, вроде игры *Rome: Total War*, в которой игрок играет роль полководца, или игры *The Sims* – в ней участник является «божественным» наблюдателем за жизнями маленьких компьютерных людей. Эти игры задают роль и сеттинг, но повествовательную линию определяют собственные действия игрока, которые могут очень сильно отличаться от одной сессии к другой.

Наконец, на противоположном конце спектра мы имеем игры, в которых художественный вымысел вовсе отсутствует. Например, такие абстрактные и формальные игры, как *go* или «*Тетрис*», совсем не подразумевающие никакого вымысла, да он им и не нужен. Но в большинстве игр вымысел – серьезная составляющая привлекательности игры, поскольку он помогает игроку понять свою роль в игровом мире, пробуждает интерес к тематике или сеттингу, мотивирует его.

Здесь мы будем рассматривать две широкие категории: игры с общей фантазией с одной стороны и игры со структурированным сюжетом с другой. Но следует иметь в виду, что это бинарно оппозитные понятия – разные игры к вопросу художественного вымысла подходят по-разному.

Фантазия

Элементы художественного вымысла присутствуют во многих играх, но не всегда излагают какую-то определенную историю. Такие игры, как *Civilization*, *SimCity* и *Minecraft*, позволяют игроку действовать по своему усмотрению в рамках игровых правил. Мир в них полностью управляется действиями «игроков» и «искусственного интеллекта» и их последствиями. Вместо того чтобы излагать заранее придуманный сюжет, программа позволяет игрокам самим придумывать свои истории.

Но даже если в таких играх и отсутствует установленный сюжет, в них все равно имеется достаточно мощный элемент: *фантазия* о том, кем является игрок в игровом мире. В *Civilization* это фантазия о правителе мощной империи. В *SimCity* – фантазия о мэре города с многочисленными «симулированными» людьми. В *Minecraft* – фантазия об исследователе, строителе и ремесленнике, овладевающем алхимическими навыками по созданию новых материалов из того, что он находит в мире.

Фантазия о конкретной роли – очень сильный мотивирующий фактор для игрока. Людям очень нравится играть роль кого-то другого, смотреть на мир его глазами и решать новые для себя задачи. Наверное, поэтому игровые фантазии обычно связаны с *мечтами и устремлениями* людей, предлагая им сыграть за того, кем они хотели бы стать – например, за влиятельного правителя или опытного воина, как это бывает во многих коммерческих играх. Но интересными бывают даже не столь высокие и благородные роли – например, в игре *Cart Life* участник играет за продавца уличной еды, едва сводящего концы с концами и воспринимающего повседневную жизнь как вызов.

Фантазия придает *смысл и значение* действиям игрока, окрашивая все, чем он занимается. Это значение настолько важно, что игры с одними и теми же механиками, но с различной фантазией или вовсе без нее, воспринимаются как очень разные. Представим, например, какую-нибудь военную игру, полностью лишенную фантазии. Вместо того чтобы выводить на поле боя отряды и нападать на врагов, игрок передвигает пластиковые фишки

по клеточкам белой доски и сталкивается с другими пластиковыми фишками, исчезающими после столкновения. У такой игры будет идентичный геймплей, но поскольку она лишена фантазии и мотивации военной игры, игрок будет воспринимать ее совершенно иначе.

Фантазия и сюжет – не уникальные элементы, а скорее части некоторого континуума, они могут сосуществовать в одной и той же программе. Например, игра *Grand Theft Auto* предлагает участнику очень увлекательную фантазию о жизни в криминальной городской среде, но в ней есть довольно четкий сюжет. В отличие от нее, в игре *Civilization* есть фантазия, но нет авторского сюжета.

Сюжет

Игровой сюжет – очень известный инструмент. Игрок играет за некоего персонажа, следуя по конкретной, придуманной сценаристами и дизайнерами, линии повествования. В нашем примере из «*Ведьмака-3*» игрок отыгрывает роль *конкретного* Геральта из Ривии, история которого разворачивается на наших глазах.

Сюжеты – это широко распространенный элемент игр, игрокам нравится принимать участие в разворачивающемся повествовании. Излагать их можно разными способами, часто сочетающимися друг с другом.

Явное повествование и экспозиция

- Игроку могут показывать кат-сцены, излагающие конкретные элементы сюжета.
- В игре могут присутствовать книги, справочники или другие средства, излагающие историю мира.

Взаимодействия с NPC

- NPC (неигровые персонажи) в команде игрока или в окружающем мире могут предоставлять множество сведений о сюжете в своих закриптованных (заданных сценарием) репликах.
- NPC заранее определенным образом могут реагировать на действия игрока и на события в игре, тем самым развивая сюжет.

Повествовательные элементы окружения

- Игроки могут выполнять задания, благодаря которым попадают в еще неисследованные части мира или получают дополнительные сведения о мире.
- Дизайн уровней, сама схема мира и его обитатели также могут способствовать изложению сюжета по мере взаимодействия с ними игрока.
- Присутствующие в мире следы прошлых событий (например, комната с красными пятнами и разбросанными в беспорядке вещами) – разновидность повествования, позволяющая игроку догадываться о прошедшем без явного изложения.
- Значимые элементы окружения, такие как звуки и смена музыкального сопровождения, также могут сообщать игроку о том, что случилось в этой области или предупреждать его о грядущих событиях.

Сюжет и агентивность

Сюжет в играх – источник известной проблемы для авторов. Чтобы история разворачивалась так, как он задумал, игра должна ограничивать свободу действий игрока. Но это уменьшает *агентивность* игрока, то есть способность взаимодействовать с игрой так, как ему захочется, влияя на исход.

Агентивность – ключевой аспект игр. Игрокам нравится действовать в игре самостоятельно, наблюдая за тем, как она реагирует на их действия и меняет свое состояние. Благодаря этому игроки узнают, как устроен игровой мир, овладевают навыками и увеличивают свои шансы на успех в будущем или просто экспериментируют из

любопытства¹⁶.

Но хотя в общем случае агентивность в играх – это чрезвычайно полезная концепция, она отрицательным образом сказывается на сюжетном повествовании и на авторском контроле. Игры очень сильно отличаются по степени того, как в них поддерживается агентивность. Можно представить их совокупность как спектр – от полной до минимальной.

На одном конце находятся игры, излагающие какую-то конкретную историю и жестко контролирующую агентивность игрока – например, книги из серии «Выбери себе приключение» (*Choose Your Own Adventure, CYOA*) и похожие на них игры-квесты. Они прекрасно излагают сюжет, но за счет радикального сокращения агентивности. Сюжет уже придуман, действия персонажа прописаны заранее, а агентивность игрока сводится к нескольким *точкам принятия решения*, где нужно выбирать один из немногих вариантов развития сюжета.

На другом конце спектра расположены игры-песочницы, в которых игрок обладает полной свободой экспериментировать и воздействовать на мир с соответствующей реакцией со стороны игры. Такая свобода обладает очень большой притягательностью и большим интересом для игрока, но и достигается она за счет большой цены. Оттого что игроку позволяют делать все, что ему вздумается, страдает сюжет. Писателю придется очень постараться, чтобы изложить историю в мире без ограничений, где игрок может легко проигнорировать сюжет и заниматься чем-то своим или же действовать вопреки замыслу автора.

В коммерческих играх сюжетные элементы редко соответствуют одной из этих крайностей и располагаются где-то посередине. Они ограничивают мир игрока так, что у него имеется *некоторая* агентивность, но при необходимости она может быть ограничена. Взять для примера один из популярных жанров, RPG с открытым миром, в которых игрок может странствовать как ему угодно и делать все что пожелает. Но если игрок захочет принять участие в сюжете или в одной из побочных историй, ему придется выполнить конкретные задания в определенном порядке, запустив тем самым внедренную повествовательную линию. В этих играх также предусматривают защиту от ситуаций, в которых игрок может нарушить сюжет, – например, запрещают нападать на какого-то определенного NPC или грабить его. В таблице 6.1 перечислено несколько жанров с указанием того, как в них контролируется агентивность игрока.

Жанры игр и примеры	Авторский контроль	Агентивность игрока
Квесты, в том числе книжные CYOA (<i>Dragon's Lair</i> , версия 1983)	Наибольший	Наименьшая
Визуальные новеллы (<i>Asa Asotomay</i>)		
История с мини-играми (<i>Du-fin</i>)		
Графические квесты, или «приключения» (<i>Muz</i>)	Значительный	В некоторой степени
Эпопея-приключения (<i>Tomb Raider</i>)		
RPG с открытым миром (<i>«Ведьмак»</i>)	Варьируется	Варьируется
Стратегия в реальном времени (<i>StarCraft</i>)		
Экономические стратегии (<i>Realms of Magic</i> Буксон)	Наименьший	Наибольший
Песочницы (<i>The Sims</i>)		
Киберспортивные (<i>Overwatch</i>)		

¹⁶ Здесь мы придерживаемся идеи агентивности, какой ее впервые описал Черч (Church, 1999), утверждая, что игроки действуют намеренно, когда могут сформулировать свои цели и выполнять осознанные действия ради их достижения. Вместе с тем существуют разные определения этого понятия; более подробное рассуждение можно найти в (*Wardrip-Fruin*, готовится к выходу).

Таблица 6.1. Варианты авторского контроля и агентивности игрока в избранных жанрах.

Последовательность

Ни фантазия, ни сюжет не являются обязательными элементами. В абстрактных играх, таких как *судоку* или *домино*, полностью отсутствует какая бы то ни было фантазия. В *го* или в *шахматах* она незначительна и несущественна. Но эти игры определенно доставляют удовольствие игрокам как абстрактные головоломки с соответствующим геймплеем.

Если же в каком-то виде художественный вымысел присутствует в игре, он должен соответствовать дизайну геймплея. Здесь важно добиться последовательности, поскольку геймплей и вымысел дополняют и усиливают друг друга, что положительным образом сказывается на игровом опыте.

Один из самых простых способов нарушить такую последовательность – ограничить действия игрока противоречащим фантазии образом. Например, если в военной игре фантазия заключается в роли генерала армии, но игрок вынужден управлять каждым солдатом отдельно, то у него вскоре возникнет ощущение, что он играет за сержанта, а не за высокопоставленного командира. Аналогично, если в игре про супергероя персонаж может летать, но на уровне расположены невидимые стены, ограничивающие прогрессию игрока, то возникает ощущение, что сверхспособности практически отсутствуют и вовсе таковыми не являются.

Выбор и агентивность. Предоставление игроку свободы выбора – важная часть геймплея, но сам по себе выбор не совсем то же самое, что агентивность. Игра поддерживает агентивность, если игрок может действовать независимо, формулировать свои цели и поступать согласно им. Но ради достижения целей игрок также должен понять, как его действия влияют на исход в будущем.

В качестве простого примера возьмем какую-нибудь визуальную новеллу, в которой игрок по сюжету должен делать выбор, но с непредсказуемыми последствиями, не понимая, как и когда его результаты проявятся в игре. Хотя повествование от этого становится более захватывающим, с точки зрения геймплея у игрока создается впечатление, что он не обладает контролем и агентивностью в игровом мире, потому что ему не удастся составить ментальную модель того, как устроена игра. Возможно, ситуацию придется улучшить тем, чтобы предоставить более явную обратную связь и сообщать игроку о том, как его действия влияют на игровой мир.

В качестве еще более экстремального примера можно привести игры, в которых произвольные действия вообще не поддерживаются. В такую категорию попадают детские настольные игры вроде «Змей и лестниц» или карточные вроде «Пьяницы», в которых ходы игроков полностью предопределены бросками костей или раскладом карт. Невозможность выбора действий делает такие игры неинтересными для опытных игроков.

Иной вид нарушения предполагает концепция так называемого *лудонарративного диссонанса*, приписываемая Хокингу и подразумевающая противоречие между геймплеем и повествованием. В своем примере (Hocking, 2007) автор описывает игру *BioShock*, в которой игрок играет за одну из двух фракций и постоянно по ходу игры совершает соответствующий моральный выбор. Тем не менее в какой-то момент сюжет принуждает игрока сделать выбор стороны независимо от того, какую роль он отыгрывал до этого, или же полностью прекратить игру. Такой сюжетный поворот принижает все предыдущие усилия игрока, делая их незначительными. В качестве другого примера, описанного Соларски (Solariski, 2017b), можно привести игру *Halo 4* с другим видом диссонанса. Согласно сюжету игроку нужно как можно быстрее победить главного врага Дидакта

(«Наставника»), но такая срочность никак не влияет на геймплей, поэтому игрок может спокойно исследовать игру и повышать свои навыки, вместо того чтобы устремляться к финишной прямой. Механики игры лишают игровой сюжет значимости.

Тем не менее нарушение последовательности иногда бывает *мощным инструментом*, поэтому его используют для того, чтобы опровергнуть ожидания и придать геймплею новизну и увлекательность. Например, с первого взгляда на серию *Crusader Kings* складывается впечатление, что это фантазия о правителе средневекового королевства. Но большую часть времени игрок тратит на мелкомасштабную политику – управление вассалами, удовлетворение потребностей придворных, расширение власти своей семьи благодаря бракам или союзам. В процессе игрок узнает о совсем иной стороне королевской жизни и о том, что значит быть королем (не столько военным командиром, сколько администратором и манипулятором).

Макроструктура и арки контента

Обратим теперь взор на крупномасштабную структуру игры. Разрабатываем мы программу с сюжетом или без, важно предусмотреть общий игровой опыт на протяжении всего времени, которое игрок потратит на игру.

Гейм-дизайнеры обычно пытаются проследить за тем, чтобы опыт игрока в ходе прохождения игры не был монотонным, а немного варьировался. В ориентированной на сюжет игре макроструктура часто описывается в терминах сюжетных линий и арок. В «*Ведьмаке*», например, имеется главная сюжетная арка, начинающаяся в разгар конфликта, постоянно расширяющаяся и увеличивающая свою напряженность по ходу игры. Кроме того, имеются побочные сюжетные линии с дополнительными заданиями и элементами, а также комедийные вставки в качестве отдыха и отвлечения от основной темы. Вместе основной сюжет и побочные истории образуют очень динамичную сюжетную арку, полную эмоциональных подъемов и спусков, сюрпризов и вызовов.

Трехактная модель

Часто об игре говорят, как о развитии некоей *арки*. Эта метафора происходит из театра и литературы и связана с описанием сюжетной линии как *трехактной модели*. Она подразумевает следующие стадии сюжета.

Завязка: знакомство с окружением и персонажами, их мотивацией, ситуацией и основным конфликтом, определяющим траектории персонажей.

Нарастание (конфронтация): развитие основного конфликта, достижение критической точки (кульминации) вследствие действий персонажей.

Развязка(разрешение): конфликт тем или иным образом разрешается, окружение и персонажи переходят в новое состояние, история заканчивается.

Можно попытаться составить график развития сюжета с точки зрения эмоциональной напряженности (интенсивности). Он будет выглядеть примерно так, как показано на иллюстрации 6.2.

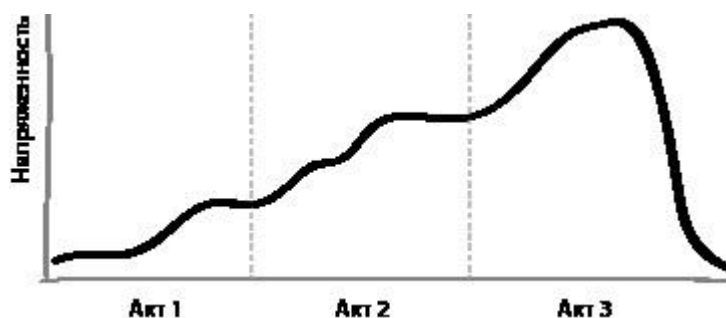


Иллюстрация 6.2. По эмоциональной напряженности восприятие геймплея можно поделить на три акта: завязка, нарастание, развязка

Эта модель известна по крайней мере со времен Аристотеля. Хотя к ней прибегали и ее развивали в своих трудах бесчисленные авторы, благодаря своему чрезвычайному удобству эта простая трехчастная структура сохранилась и до наших дней. Она обладает универсальной для всех людей привлекательностью и помогает развивать сюжет по мере усиления конфликта, разрешение которого происходит в конце повествования.

Любопытно, что такого рода макроструктура встречается и в играх, в которых отсутствует явно выраженная повествовательная составляющая. В главе 4 «Системы» мы рассуждали о соревновательной игре для нескольких игроков «Монополия», обнаружив, что даже в ней есть своя арка – в процессе игры благосостояние игроков растет или снижается. Ее макроструктура не основана на какой-то конкретной истории (сюжете), но к концу эмоциональная напряженность усиливается благодаря конкуренции между участниками. Схожие арки присутствуют во всех других типах игр.

Сюжетная арка и эпизоды

Передать отдельную историю можно в одном фильме или романе, но часто повествование делят на несколько частей, предъявляя их со временем. Этот способ типичен для телесериалов, ситкомов, романов с продолжением, комиксов и других подобных средств коммуникации.

Такие истории иногда называются *эпизодами*, они связаны между собой одним общим нарративом (повествовательной линией). Каждый из коротких эпизодов имеет свою тему и свой сюжет, свою завязку, свое нарастание конфликта и свою развязку. Но вместе они излагают более крупную историю с многочисленными конфликтами и исходами.

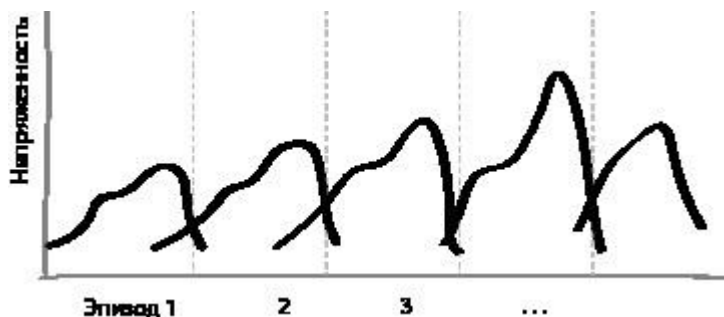


Иллюстрация 6.3. Эпизоды, представляющие собой пересекающиеся между собой арковые структуры

Иногда такое многоэпизодное повествование само по себе является самостоятельным сюжетом с более крупной арочной структурой – взять для примера телеэкранизацию какой-нибудь книги. Но столь же часто эпизоды составляют никогда не заканчивающееся повествование – как в ситкомах или «мыльных операх». Тогда они представляют собой взаимосвязанные между собой арки с регулярным появлением новых и угасанием старых.

Такие эпизодические арки наблюдаются и в играх, особенно ориентированных на сюжет и поделенных на более мелкие элементы. Например, в ролевых играх (серия *The Witcher*, серия *Grand Theft Auto* и т. д.) имеются свои крупномасштабные арки, а иногда и несколько пересекающихся между собой, которые игрок исследует на протяжении многих часов геймплея. В то же время они поделены на ряд более мелких эпизодов (например, по регионам действия или заданиям сюжетной линии), в которых есть свои арки. В данном случае все это занимает гораздо меньше времени – в процессе прохождения игрок заодно движется по более крупной сюжетной арке. Как будет сказано ниже, в разделе про регулирование темпа, эпизоды также помогают варьировать стресс и интенсивность, что

разнообразит игру.

Как в сериалах бывают эпизоды-филлеры, не связанные с общим сюжетом, так и в ролевых играх часто встречаются необязательные побочные задания и встречи с открытым концом, которые могут быть интересными сами по себе и заполнять повествование, но никак не помогают игроку двигаться по сюжету. У игроков появляется больше свободы в игровом мире, потому что они могут исследовать побочные истории и дополнительные сюжетные линии по своему усмотрению и в своем темпе, а затем при желании возвращаться к основному сюжету.

Такую эпизодическую структуру можно разглядеть и в играх с уровнями, от экшен-приключений вроде *Tomb Raider* и шутеров от первого лица вроде *Halo* до казуальных игр вроде *Candy Crush Saga*. Многие типы игр состоят из индивидуальных уровней, у каждого из которых имеются свои небольшие арки с завязками и развязками, но они образуют последовательность с разным темпом прохождения, где серии более легких уровней перемежаются более трудными, демонстрирующими неожиданный вызов, разнообразие и драматизм.

Трехактная модель и несюжетные игры

В играх без сюжета гейм-дизайнеры и игроки также пользуются трехактной моделью для описания своих впечатлений и игрового опыта. Геймплей даже таких игр, как *Civilization* или *шахматы*, нередко описывается по стадиям – начало игры (завязка, дебют), середина (миттельшпиль, «мидгейм») и конец (эндшпиль, «эндгейм»). В обзорах и даже в повседневных разговорах можно встретить такие высказывания, как «на ранней стадии геймплей хорош, но мидгейм слишком сложный», «эндгейм плохо настроен и бесконечно затянут» и т. д.

Такое разделение следует из распространенного наблюдения, что почти в каждой игре ее начальная стадия ощущается не так, как серединная или конечная. В спорте и в соревновательных играх начало гораздо менее напряженное, потому что на данном этапе еще есть возможность исправить ошибки и погрешности по сравнению с конечной стадией, когда исправить ошибки очень трудно. В исследовательских играх, таких как *Minecraft* или *Terraria*, на начальной стадии игрок занимается исследованием и сбором ресурсов, но с течением игры начинает пользоваться ими для постройки фантастических конструкций. В стратегиях в реальном времени на начальном этапе нужно узнать особенности местности, накапливать ресурсы, отстраивать базу и готовиться к неизбежным сражениям с другими игроками и т. д.

Эти стадии можно описать следующим образом.

Начало игры : в основном это выяснение того, что происходит: игроки исследуют игровой мир, местность, готовят почву для собственно игры.

Середина игры : когда все начинается «по-настоящему». Игроки преодолевают препятствия или вовлекаются в конфликт, игра становится более напряженной, приходится выполнять различные задачи, сражаться за ресурсы, вступать в открытую или скрытую конфронтацию или иным образом двигаться к победе или к достижению целей.

Конец игры : сложность или конфликты достигают пика и подталкивают игрока к разрешению игры (достижению победы, выполнению целей, набору максимального количества очков и т. д.).

Эти стадии принимают различные формы в зависимости от жанра игры. Например, будет отличаться их продолжительность. В военной игре завязка может затянуться надолго, а в каком-нибудь физическом виде спорта начало или этап осознания того, где ты находишься и что нужно сделать, может быть очень коротким. Отличается и интенсивность стадий. В стратегиях в реальном времени под конец часто происходят самые массивные сражения,

а «Монополия» печально славится затянутостью и «войной на истощение» с низкой напряженностью.

Данная модель применима как к однопользовательским, так и к многопользовательским играм, поскольку конфликт между игроками можно заменить конфликтом между игроком-человеком и ИИ-персонажами. Схожим образом в играх с неигровыми персонажами или без явно выраженного конфликта интенсивность игрового опыта зависит от необходимости контролировать не слишком податливую игровую симуляцию и овладевать навыками действий или справляться с трудными задачами.

Схемы повествования

В ориентированных на сюжет играх в качестве строительных блоков общей сюжетной арки используются различные схемы повествования. В данном разделе мы рассмотрим распространенные схемы небольшого масштаба, а затем разберем, как из них складываются более крупные повествовательные структуры. В частности, воспользуемся анализом с точки зрения *структур выбора (развилки)*, то есть того, как игра сочетает авторские элементы сюжета с выбором игрока, влияющего на его развитие¹⁷.

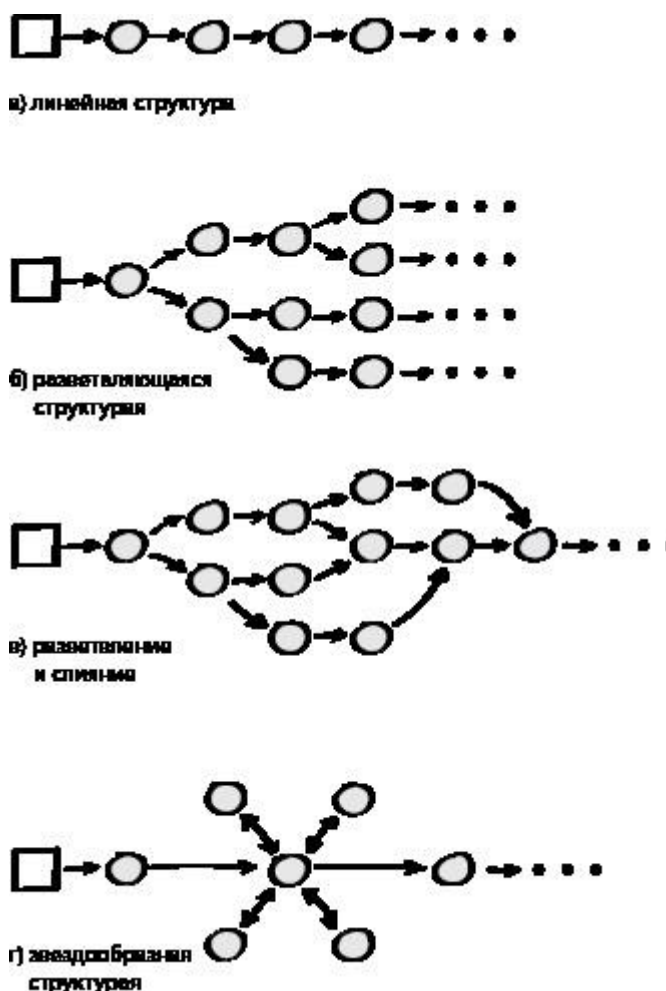


Иллюстрация 6.5. Примеры распространенных схем повествования

¹⁷ В системном анализе данных топологий выбора и их взаимодействия с другими аспектами игрового опыта Мохортер с коллегами (Mawhorter et al., 2014) использует термин *поэтика выбора*. В индустрии для этого явления не выработалось единого термина, но часто используются понятия «нарративная (повествовательная)» или «сюжетная» структура.

Выбор в данном случае – широкий термин. Под ним мы подразумеваем различного рода решения, задания или выполняемые игроком действия, которые могут *изменить ход сюжета*. Выборы и сюжетные элементы связываются друг с другом различными способами. Некоторые из основных схем показаны на иллюстрации 6.5. Они относительно просты, в небольших играх их могут использовать как раз в таком виде, а в больших данные схемы могут образовывать более крупную структуру.

Линейное повествование

В самом простейшем случае сюжет бывает полностью линейным, как показано на иллюстрации 6.5а. В классических квестах вроде *Myst* игроку предъявляют по большей части линейную, не меняющуюся историю. В ней могут быть различные сложные задачи (решить головоломку, пройти уровень), но точки выбора сводятся к ситуации «*успех или поражение*». Сюжет задан заранее, а игрок должен успешно выполнить текущее задание, чтобы пройти к следующей точке сюжета.

Такая схема наблюдается и в других жанрах (вроде экшен-игры с линейной прогрессией уровней), где сюжет – это скорее фоновый элемент, дополняющий геймплей. Например, в таких совершенно разных экшен-играх, как *God of War* (версия 2005 года) и *Katamari Damacy*, в интерлюдиях между сложными и интересными уровнями рассказывается общая история происходящего. Точно так же и стратегии *StarCraft* или *Company of Heroes* включают в себя «режим кампании», где уровни распределены таким образом, чтобы рассказывать историю отдельной военной кампании, а игрок должен обязательно победить на каждом этапе, чтобы пройти к следующему.

Выборы с ветвлением

Вместо линейного сюжета игроку можно предложить точки выбора, *меняющие* сюжет определенным, заранее прописанным образом. В качестве популярной метафоры для этого используется *разветвляющаяся структура*. Сюжет развивается единым образом, пока не достигает точки выбора, а решение игрока определяет один из нескольких дальнейших вариантов развития событий. Например, в какой-нибудь критической ситуации мы можем предложить игроку выбор – сразиться с врагом или избежать сражения, войти в темную пещеру или найти обходной путь и т. д. В зависимости от действий игрока сюжет последует по выбранной траектории (см. иллюстрацию 6.5б).

Возможно, один из самых ранних примеров таких игр – книги из серии «Выбери себе приключение» (*Choose Your Own Adventure, CYOA*). Это были физические книги с пронумерованными параграфами, в каждом из которых рассказывался кусочек истории, а под конец читателю давалось прямое указание («перейди на страницу 34») или предлагался выбор («если ты решил напасть на монстра, перейди на страницу 56, если нет – перейди на страницу 78»). С каждым прочтением при разных выборах получалась иная последовательность параграфов с довольно сильно отличающимся сюжетом.

Но в такой простой ветвящейся структуре имеется одна значительная проблема: чем больше вариантов мы предоставляем игроку, тем больше нужно придумывать контента для всех возможных прохождений, чтобы игрок почувствовал значимость каждого выбора. При прямом следовании этому принципу игра с разветвляющимися выборами будет страдать от комбинаторного взрыва контента (Crawford, 2005), а явная реализация всех возможных историй, соответствующих каждому варианту, будет непрактичной для любой игры (кроме той, где дерево выборов очень мало). На практике такая топология используется лишь для небольших повествований, потому что требует большого количества контента (текстовых фрагментов сюжета, возможно озвучки, анимированных кат-сцен и т. д.). Гораздо хуже, если при однократном прохождении игрок исследует лишь *одну-единственную траекторию* всего пространства, а весь другой тщательно создаваемый контент пройдет мимо него.

Разветвление и слияние

Проблему создания дополнительного контента для разветвляющегося сюжета можно упростить, периодически объединяя разветвления, то есть, по сути, превращая огромное и ветвистое дерево в последовательность небольших и более удобных для разработки поддеревьев. Эту стратегию можно назвать схемой «разветвление и слияние»¹⁸, показанной на иллюстрации 6.5в.

Такой подход критикуют за иллюзорность выбора. Например, если несколько вариантов выбора игрока в середине дерева приводят в конце к одному и тому же исходу, то являются ли они значимыми? Как правило, такие вопросы появляются у автора сюжета, рассматривающего его словно «с высоты птичьего полета». На практике игрок, проходящий игру впервые, не знает об этом, у него нет способов узнать о ее топологии, как и нет причин полагать, что какой-то из выборов менее значим для развития сюжета, чем другой. И только при последующих прохождениях игрок начнет осознавать топологию и понимать, какие последствия имеют его те или иные действия. Но в сюжетных играх может и не быть последующих прохождений, так как игрок удовлетворится первым вариантом и перейдет к следующему эпизоду или к другой игре.

Кроме расширения и слияния опишем еще два варианта выбора, которые могут показаться неэффективными с авторской точки зрения, но часто не обнаруживают своей истинной природы при единственном прохождении (Mawhorter, et al., 2014). При *ложном выборе* казалось бы разные действия ведут к одному и тому же следующему узлу, хотя игрок этого и не знает (примером могут служить различные реплики в диалоговом дереве, вызывающие одну и ту же реакцию неигрового персонажа). *Выбор предпочтений* состоит из вариантов схожих действий, влияющих на фантазию игрока и на восприятие игрового мира (например, разных вариантов одежды аватара или какой-то особенной фразы в диалоге). Такого рода точки выбора, которые можно организовать относительно легко, способны оказать существенное влияние на восприятие игрока при первом прохождении, но с ущербом для последующих, когда игрок построит улучшенную ментальную модель того, как его действия связаны с состоянием игры.

Ветвление с состоянием

Еще один распространенный способ развития сюжета – это добавление к нему понятия «состояние». До сих пор нам приходилось отслеживать лишь общее положение игрока. Но если отслеживать и другие данные, а затем использовать их для определения исхода выбора, то можно создавать сложные повествования при очень сжатом графе выборов.

Представим, что мы предусмотрели в игре диалоги с оскорбительными вариантами высказываний и следующим эффектом: при первом оскорблении разговаривающий с вами персонаж может и проглотить обиду, но в следующий раз его реакция не заставит себя ждать. Такой диалог можно реализовать как простую разветвляющуюся структуру, но поскольку игрок вправе выбрать оскорбление при любом другом разговоре в будущем, в итоге у нас получится неоправданно сложный граф. Вместо этого можно добавить для оскорблений *состояние диалога*, показатель которого увеличивается каждый раз, когда игрок достигает точки выбора и решает кого-то оскорбить, а затем менять сюжет, если этот показатель достигнет определенного значения (см. иллюстрацию 6.6).

Насколько можно представить, добавление модифицируемого и проверяемого в ходе развития сюжета *состояния игры* значительно упрощает представление сложных

¹⁸ Костилян (Costikyan, 2007) использует термин «бусы на нитке», потому что подграфы начинаются с единственной точки выбора, за которой следует более богатая сюжетная структура, затем снова сходящаяся к единственной точке.

отношений между различными сущностями, разделенных в пространстве и во времени. Кроме общего глобального состояния, возможно отслеживать показатели персонажа (здоровье, выносливость и т. д.), межличностные отношения (например, выразить степень дружбы через количественный показатель) и аналогичные моменты, а затем пользоваться ими для определения направления развития сюжета.

Из-за преимуществ данного метода им *очень часто* пользуются в повествовательных играх, включая интерактивную художественную литературу, квесты, визуальные новеллы и прочие схожие жанры. В то же время данный подход значительно увеличивает сложность для автора, которому теперь нужно заботиться не только о локальных последствиях отдельных точек выбора, но и о более глобальных последствиях в различных местах и в разное время. Такого рода «призрачные действия на дальнем расстоянии» потенциально делают процесс поиска и исправления ошибок гораздо труднее.

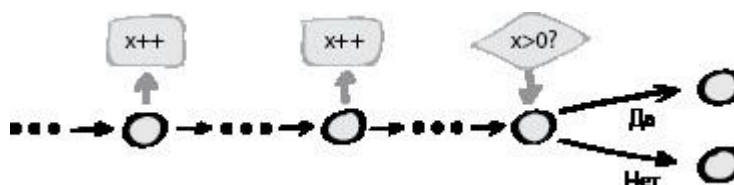


Иллюстрация 6.6. Пример точек выбора, влияющих на состояние мира, и будущая точка выбора, где проверяется его состояние

Звездообразная структура

Вместо того чтобы вести игрока по определенной последовательности точек выбора, бывает проще предоставить ему некоторую гибкость – например, предусмотреть ряд выборов, сделать которые игроку рано или поздно обязательно придется, но он сам будет решать в каком порядке и в какое время. Например, можно предусмотреть ситуацию, в которой игроку нужно поговорить с определенным персонажем, зачистить монстров в определенной зоне и создать определенный артефакт, но сделать это можно в любом порядке.

Обычно такой ситуации соответствует *звездообразная структура* (именуемая также схемой «ступица и спицы»). Игрок несколько раз возвращается и выполняет связанные с ней сюжетные элементы, что напоминает спицы колеса со ступицей (см. иллюстрацию 6.5 г). Игрок может взаимодействовать с любым элементом, специфическим образом изменяя состояние игры, а затем вернуться к ступице («хабу»), перейти к следующему элементу и так далее.

По сравнению с направленным ациклическим графом, о котором мы рассуждали ранее (линейная, разветвляющаяся или сливающаяся схемы), звездообразная схема предоставляет игроку больше свободы и агентивности в выборе того, как и когда идти далее по сюжету. Некоторые из самых ранних примеров данной схемы (например, экшен-платформеры *Super Mario 64* и *Jak and Daxter*) воплощали ее в дизайне уровней – разные уровни, представляющие фрагменты всего игрового мира, были связаны с центральными зонами.

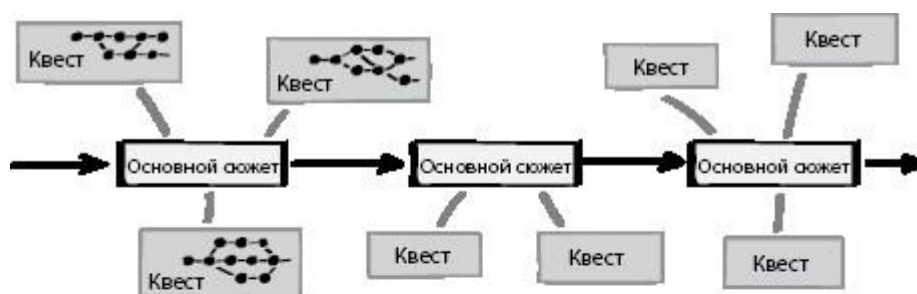
Повествовательная композиция и квесты (задания)

В предыдущем разделе мы описали отдельные топологии, такие как линейный, разветвляющийся, сливающийся и звездообразный графы. В некоторых играх они полностью описывают общую повествовательную структуру. Например, книгу *СУОА* можно смоделировать как один граф с разветвляющимися выборами. Но в более сложных повествованиях мы можем сочетать различные схемы. Например, создать повествовательные модули малого масштаба с индивидуальной топологией, реализованной в виде *заданий*

(называемых также «квестами» или «миссиями»), а затем составить из этих модулей более крупную структуру.

В данном случае «квест» – это короткая, имеющая определенную цель повествовательная последовательность, обычно с линейной или разветвляющейся структурой с определенными начальными условиями, определенными конечными точками, которых нужно достичь, а также с потенциальными специфическими последствиями в случае успеха или неудачи. В ролевых играх, например, игрок может встретить «квестодателя» – неигрового персонажа, который начинает короткий разговор и просит помочь избавиться деревню от призрака. Если игрок принимает задание, то идет на поиски привидения, сражается с ним и в случае победы возвращается за наградой.

Квесты – распространенное и полезное средство изложения сюжета, потому что они поддерживают повествовательную экспозицию (через кат-сцены или диалоги с неигровыми персонажами), а также поощряют игрока принимать участие в разворачивающемся сюжете, используя цели для мотивации и награды в качестве «морковки».



а) линейный сюжет с побочными квестами



б) открытый мир с основным сюжетом и разнообразными квестами

Иллюстрация 6.7. Примеры схем повествования

В качестве простой композиции можно распределить квесты в линейной последовательности зон, каждая из которых представляет собой «ступицу со спицами» (см. иллюстрацию 6.7а) и открывается при завершении всех квестов в предыдущей зоне. Такой стиль используется в играх, подобных *Jak II*, но его можно использовать и с другими повествовательными элементами. Например, в *Gone Home* для перемещения между ступицами игра требует от участника вместо решения головоломок исследовать различные повествовательные элементы окружения.

Создание повествовательной композиции с помощью последовательности небольших модулей – хороший способ расширения сюжета, позволяющий гейм-дизайнеру разрабатывать более крупные повествовательные структуры на основе более мелких, самодостаточных модулей, часто представляющих собой последовательность различных квестов, вносящих разнообразие в общее повествование.

Открытые миры

Мы уже обсуждали открытые миры на примере серии игр «Ведьмак». Но их топология представляет собой еще один вариант композиции. Вместо того чтобы заставлять игрока следовать по цепочке элементов, она предоставляет ему целый набор разнообразных маленьких историй, некоторые из которых связаны между собой и образуют главную сюжетную линию, а другие не связаны с ней (см. иллюстрацию 6.7б). Эти дополнительные сюжетные линии и повествования разбросаны по всей игре и ждут, пока игрок откроет их. Можно утверждать, что топологически отдельные истории образуют густую связанную сеть, поскольку в большинстве случаев в текущем контексте (таком как уровень персонажа или уровень опыта) игрок может переходить от одной из них к любой другой.

С помощью данной модели можно организовать разные сюжетные линии и позволить игроку в любое время переключаться между ними. Игрок даже может покинуть их, исследуя различные участки мира по своему усмотрению, а также заниматься чем угодно в рамках правил игры. Эти внешние элементы могут влиять на прогресс основной сюжетной арки или быть совершенно независимыми, служа фоновой историей, косметическими элементами, возможно «бустами», источником добычи и т. д.

Такой контроль над квестами предоставляет игроку большую свободу и увеличивает агентивность. Он делает открытый мир увлекательным, продолжая при этом вести игрока по нужной сюжетной арке (хотя и с меньшим авторским контролем над последовательностью повествования и темпом).

Открытые миры и дизайн квестов

Открытый мир в своей реализации должен предусматривать различные состояния. Отдельные персонажи имеют отслеживаемое состояние (параметры, инвентарь и т. д.), но сам мир тоже имеет его и помнит, что уже случилось по сюжету (какие квесты завершены, на каких монстров уже поохотились и т. д.). Некоторые из данных состояний видимы для игрока – например, можно просматривать параметры и инвентарь своего персонажа, другие скрыты – такие как список уже посещенных населенных пунктов и убитых монстров, но игра при этом предлагает подходящие под развитие истории квесты.

Однако, как уже было отмечено ранее, добавление отслеживаемых состояний затрудняет тестирование и поиск ошибок в создаваемом контенте, потому что вводит связь между потенциально не имеющими отношение друг к другу частями игры.

В качестве примера представим, что мы работаем над дизайном RPG вроде «Ведьмака-3» и доступ к разным частям игры зависит от того, совершил ли игрок в прошлом какие-то значимые события. Предположим, что одним из ранних квестов было задание «охота на призрака», заключающееся в том, чтобы сразиться с привидением в заброшенной деревне, и оно увеличивает показатель «счетчика призраков» игрока. Гораздо позже, уже вне всякой связи с этим квестом, для охотников на привидений открывается особая локация, доступная только тогда, когда «счетчик призраков» превысит определенное настраиваемое количество.

Теперь представим, что через много месяцев после того, как оба этих квеста были прописаны и протестированы, мы возвращаемся, чтобы поработать над первым квестом, и решаем, что он не подходит для ранней стадии игры. Поэтому вместо охоты на призраков мы заставляем игрока охотиться на троллей и соответствующим образом меняем художественное описание первого квеста. Но такое небольшое изменение может непреднамеренно повредить тайную локацию в поздней стадии игры, потому что у игрока уже нет средств для достижения нужного счетчика призраков. Что еще хуже, мы можем даже не заметить, что нарушили скрытую связь между двумя данными моментами, пока очень тщательно не проследим зависимости между различными элементами, меняющими

показатели.

В общем случае опасно пользоваться простым глобальным состоянием мира как механизмом связи между различными арками контента, но не менее опасно пользоваться глобальными переменными в разработке программного обеспечения для связи между различными модулями. Такой способ позволяет создавать интересные зависимости, его можно довольно успешно реализовывать, но он приводит к потенциальным «багам», которые бывает сложно искать и анализировать. Из-за этого в основанных на квестах играх их части часто изолируют, ограничивая взаимодействия между конкретными элементами, чтобы на них не влияли другие квесты или действия игрока, а любые взаимодействия были ясно видны для дизайнера.

ИИ и повествование . Сюжеты в играх обычно создаются авторами заранее, представляя собой смесь повествовательного изложения, сильно заскриптованных квестов, ситуаций и других авторских элементов. Но в последнее время выходит все больше работ, посвященных технологии *повествовательного искусственного интеллекта* , позволяющей игроку действовать по своему желанию, а игре реагировать на эти действия с соответствующими последствиями для сюжета.

Это открытая проблема в исследованиях ИИ и гейм-дизайна, а также довольно трудная задача, требующая размышления и знаний. В данный момент не существует коммерческих игр, в которых бы полностью была реализована данная концепция, хотя есть эксперименты, допускающие искусственно генерируемое повествование в очень ограниченных условиях. Пожалуй, самый известный пример – игра *Façade* (см. Mattheas and Stern, 2003), доступная онлайн.

Распространенный способ явного установления зависимости между двумя квестами – это некий вручаемый игроку особый «квестовый предмет» . В «Ведьмаке-3 », например, одни из квестовых предметов в начале игры – это «письма о безопасном поведении», открывающие доступ в новую зону игры. Их вручают неигровые персонажи после выполнения особых заданий на ранней стадии.

В нашем примере можно сделать так, что после выполнения квеста «охота на призрака» игроку выдается особый «пропуск», а при проходе в зону на поздней стадии проверяется его наличие. Месяцы спустя, решив изменить дизайн квеста «охота на призрака», мы видим, что он предоставляет игроку особый предмет, и можем по базе данных быстро найти, где он используется. Таким образом, легко автоматизировать аналогичные проверки. Если каждый фрагмент контента предоставляет особые квестовые предметы, то по их коду можно проверить, существует ли хотя бы один квест, в котором требуются эти предметы и наоборот.

Несмотря на сложности, ориентированные на квесты, открытые миры очень популярны – они дают игроку высокую степень автономии, а также контроль над аркой контента и его подачей. Создание контента для открытого мира – непростая задача, требующая значительных авторских усилий и работы по его реализации. Но в проектах, где это имеет смысл, они создают удачное сочетание агентивности и авторства.

Миры-симуляции

На другом конце спектра игр с конкретным повествованием находятся программы, предоставляющие игроку максимальные агентивность и автономию – игры, основанные на симуляциях, обновляющие игровой мир в ответ на действия игрока и искусственного интеллекта согласно определенным правилам. Обычно к этому склоняются такие жанры, как симуляторы, стратегии, экшен-аркады и многопользовательские игры.

В идеале они должны следовать арке с разной степенью интенсивности и темпа, что предоставляет разнообразие и позволяет чередовать напряженные вызовы с отдыхом и созерцанием. Но создавать предсказуемые арки контента для таких игр трудно. Игроки в

игровом мире действуют автономно, так что их действия непредсказуемы. В конце концов игроки создают свои собственные истории и повествования о происходящем.

Без явного сюжета приходится полагаться на настройки геймплейных систем, механик и циклов обратной связи, чтобы макроструктура возникала просто на основе взаимодействия игрока с игрой. Для этих целей особенно полезны системы, создающие долгосрочные циклы геймплея.

Среди примеров систем, генерирующих долгосрочные арки, можно назвать следующие.

- Деревья технологий или системы крафтинга, согласно которым игроки тратят ресурсы на разблокировку новых способностей на протяжении длительного промежутка времени, что позволяет модулировать интенсивность и вызовы. Такой инструмент очень популярен в симуляторах и стратегических играх, включая *Civilization*, *StarCraft*, *Factorio* и многие другие.

- Явное определение привязанных к конкретным достижениям «эпох» геймплея, которые разблокируют новые способности, предоставляют новые требования и меняют баланс игры. В результате может получиться довольно явно выраженная арочная структура – например, четыре эры национального строительства в игре *Tropico 5*.

- Добавление случайных событий, усиливающих сложность и вводящих некоторые элементы сюжета (засухи или пожары), возникающих стихийно и нарушающих долгосрочные планы игрока. Этот инструмент применяется в экономических стратегиях и играх-менеджерах, таких как серия *Caesar*, *Aven Colony*, а также в более казуальных играх вроде *CityVille*.

В общем случае создание долгосрочной структуры геймплея в системно нагруженных играх – задача не из легких, поскольку гейм-дизайнерам приходится влиять на действия игрока и на геймплей только косвенно, посредством настройки данных систем, возможностей и наград, а затем тестировать и проверять, как они работают. В то же время для таких игр бывает важно продумать план макроструктуры, потому что игроки действительно воспринимают игры как состоящие из начальной, серединной и конечной стадий, отличающихся друг от друга.

Регулирование темпа

При разработке структуры игры важно учитывать тот темп, в котором игрок будет воспринимать структуру. Под регулированием темпа обычно подразумевается *варьирование интенсивности игры со временем*.

Главная задача при этом состоит в том, чтобы не допустить в игре необоснованно длинных промежутков высокой интенсивности без отдыха и передышек. Повысить сложность до высокого уровня и держать игрока на ней в течение длительного времени – гарантированный способ вызвать психологическую усталость; это все равно что показывать боевик с постоянными перестрелками, не позволяющими зрителю перевести дыхание. В шутерах дизайнеры уровней давно поняли, что постоянный приток врагов и бесконечная череда вызовов перегружает восприятие игрока, поэтому таких ситуаций следует избегать. Вместо этого на уровнях, как правило, моменты интенсивного геймплея чередуются с тихими зонами, где игроки могут отдохнуть, перегруппироваться, пополнить запасы или просто исследовать мир.

Цель хорошего регулирования темпа – варьирование интенсивности геймплея и нахождение баланса между напряженной игрой и отдыхом. Как именно это будет реализовано в конкретной игре – в большой степени зависит от жанра и ее типа. Например, в экшен-играх от первого лица регулирование темпа достигается за счет дизайна уровней и варьирования количества и типов угроз, чередования безопасных и опасных зон, периодической проверки рефлексов игрока (например, когда ему необходимо защищаться от

волн врагов) и так далее¹⁹. В других играх регулирование темпа может осуществляться иными методами. В пошаговой военной игре интенсивность процесса усиливается благодаря тому, что игрокам приходится полагаться не на рефлекс, а на свои умственные способности и умение стратегически мыслить. В играх без множества уровней, представляющих непрерывную последовательность на протяжении одной сессии (например, военные игры, большие стратегии или песочницы) хорошее регулирование темпа достигается за счет варьирования вызовов со временем. Но это гораздо труднее реализовать в системно-ориентированных играх, потому что темп в нем должен определяться самими системами и циклами, а не авторским контентом.

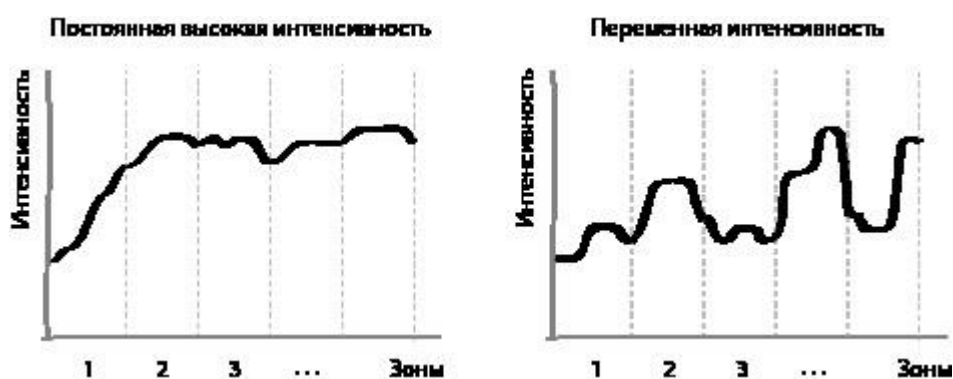


Иллюстрация 6.8. Пример постоянной высокой интенсивности в сравнении с сочетанием ее подъемов и спадов

Существуют различные способы добавления времени для отдыха, сокращения интенсивности и улучшения темпа. Большинство из них специфичны – например, уровни-«передышки», свойственные играм, основанным на уровнях. Но сразу же на ум приходят две тактики, распространенные среди самых разных жанров: *исследование* и *рутинная деятельность* (бесполезная деятельность, деятельность для вида).

Исследование – это то, чем игрок естественным образом занимается сам по себе. Игрокам нравится посещать различные части игрового мира или экспериментировать с механиками. Важно оставлять в игре некоторое место для исследований, чтобы участник мог перевести дух, побродить, а затем вернуться к основному испытанию.

В иных случаях можно заставить игрока перед переходом к следующему испытанию для вида заниматься какой-либо деятельностью. Она отнимает время, хотя гейм-дизайнер мог бы оптимизировать процесс, но дает игроку передышку и возможность снять напряжение. Такой рутинной деятельностью может быть путешествие между различными локациями квеста или продолжительный сбор ресурсов, а также использование других геймплейных систем, например крафтинга или экономики.

При этом исследование и рутинная деятельность порождают со стороны игрока несколько разные реакции. Исследование чаще воспринимается как нечто положительное, потому что игрокам обычно нравится узнавать что-то новое в своем темпе. Рутинная же легко может показаться скучной, бессмысленной и разочаровывающей, если не превратить ее в какое-нибудь интересное занятие между испытаниями.

Метаигра

В предыдущем разделе мы исследовали длительные игры, такие как RPG, прохождения

¹⁹ Некоторые игры пытаются даже автоматизировать данный процесс. Например, в игре *Left4Dead 2* имеется модуль «Режиссер», отслеживающий темп и вставляющий различные вызовы для разнообразия (Serviss, 2013). Но такая реализация была подвергнута критике со стороны игроков за слишком предсказуемую вариативность, а это подчеркивает, насколько сложно бывает обеспечить хороший темп.

которых занимает много часов, а также разновидности макроструктур, поддерживающие данные виды игр. Но что можно сказать о коротких играх, в которых продолжительность каждой сессии выражается минутами? Даже если отдельные игровые сессии гораздо короче и проще, в таких играх тоже может наблюдаться долгосрочная структура, но проявляющаяся на протяжении *многих сессий*.

Например, в коллекционных карточных играх вроде *Hearthstone* или *Magic the Gathering*, каждая сессия очень коротка, а каждый расклад, как и в традиционных карточных играх, разыгрывается в течение нескольких минут. Тактический вызов (как лучше всего распорядиться выпавшими картами) очень привлекает, а ряд быстро следующих друг за другом коротких сессий с очень высоким темпом вызывает желание играть снова и снова.

Но между сессиями возникает очень интересная структура. В отличие от традиционных карточных игр, участники не пользуются картами обычной колоды. У каждого имеется своя составная колода, из которой они извлекают карты, а между сессиями пытаются оптимизировать ее для последующих игр. Игроки стратегически мыслят и анализируют действия своих противников, пытаются вычислить, в каком стиле лучше всего играть, а затем добавляют и удаляют карты, составляя колоду для нового матча. Если взять еще больший временной отрезок, то игроки открывают для себя различные типы карт, которые могут поменять их стратегию, поэтому обмениваются картами с другими игроками или тратят виртуальную валюту (или реальные деньги) на покупку карточных колод и «бустер-паков», благодаря чему у них появляется дополнительный выбор карт.

Такого рода структура является *метаигрой* – игрой поверх игры, приносящей удовольствие деятельностью, основанной на игре, но происходящей вне отдельных игровых сессий или испытаний.

Можно описать метаигру в свете двух популярных категорий:

- *метаигра мастерства* – активность, ориентированная на мастерство и овладение игрой;
- *социальная метаигра* – активность, ориентированная на взаимодействие с другими в контексте игры.

Мы также опишем *модифицирование игр*, происходящее вне отдельных игровых сессий и позволяющее участникам использовать игру как гибкое средство творческой самореализации.

Метаигра мастерства

Иногда метаигра рождается из желаний игрока повысить свои навыки или научиться лучше играть и благодаря его внутренней мотивации. Мы уже упомянули о карточных коллекционных играх, в которых участники играют отдельные матчи, а затем тратят время между ними на улучшение своих колод и составление планов на последующую игру. В качестве другого примера можно привести гоночные игры вроде *Forza Motorsport* или многопользовательские шутеры от первого лица вроде *Counter Strike* или *Overwatch*, в которых напряженные игровые сессии перемежаются передышками, улучшением снаряжения и другой подготовкой к предстоящим сессиям.

Некоторые популярные примеры элементов метаигры следующие.

- Улучшения персонажей или аватара (от стратегического до косметического).
- «Прокачка» персонажа или членов отряда (например, трата очков уровня на усиление параметров).
- Смена инвентаря и получение новых вещей (карт, оружия и т. д.).
- Найм новых членов отряда или смена автомобилей в гоночной игре.
- Поиск новых стратегий и анализ записей предыдущих сессий.

Метаигра возникает в связи с тем, что вдобавок к вызову, с которым игрок сталкивается во время каждой сессии, ему также приходится принимать интересные и значимые решения *между* ними. Эти решения значимы, потому что связаны с системами и механиками, действующими во время игровой сессии. Они либо могут увеличить шансы игрока на победу или продвижение в игре, либо значимы тем, что усиливают фантазию – как в случае косметических улучшений.

Метаигру можно наблюдать и в играх, не ориентированных на победу или состязание, но в них она принимает несколько иную форму. Например, в игре *The Sims* – несоревновательном симуляторе жизни, где игрок в отдельный промежуток времени контролирует отдельное семейство, метаигра может заключаться в том, чтобы заселить весь район интересными персонажами, взаимодействие между которыми будет приводить к более запоминающимся ситуациям и историям.

Социальная метаигра

В других случаях метаигра возникает, потому что игроки получают удовольствие от социальных эффектов игровых сессий. Например, если в игру с короткими сессиями решили поиграть любители состязаний, они могут превратить ее в соревнование – кто наберет максимальное количество очков, кто быстрее закончит, кто первым выполнит необычный ход и т. д. Такие игроки будут раз за разом проходить игровые сессии, чтобы состязаться с друзьями или попасть в таблицу лидеров. Это и есть разновидность социальной метаигры.

Но такая метаигра не обязательно должна быть соревновательной или подразумевать противников. Она может быть и творческой, подразумевающей сотрудничество. В *The Sims* или *Roblox* игроки могут создавать внутриигровые объекты и украшения, загружать их на сайт игры, получать очки рейтинга и отзывы со стороны других участников, которым понравились их творения. Это тоже социальная метаигра, но основанная на признании труда и творчества.

Вот еще несколько примеров такой метаигры.

- Соревнование за статус лучшего игрока (часто при поддержке таблицы лидеров).
- Демонстрация навыков – получение достижений в онлайн-профиле.
- Обмен своим геймплеем с другими людьми – например, публикация видеопрохождений или стрим игры.
- Демонстрация своего личного стиля с помощью редких предметов одежды или коллекционных предметов.
- Демонстрация своих творческих работ, таких как созданные объекты или «скины» («облики») персонажей.
- Наблюдение за чьим-то состязанием или просто за прохождением игры.

Социальная метаигра часто возникает спонтанно, когда игроки играют вместе или обсуждают свой игровой опыт, но ее могут поддерживать и гейм-дизайнеры, дополняя игру следующими механиками или онлайн-инструментами.

- Достижения, демонстрирующие чей-либо прогресс.
- Счет и таблицы лидеров, облегчающие сравнение мастерства между игроками.
- Лиги или кланы в многопользовательских играх, где игроки могут самоорганизовываться в команды.
- Режим потоковой трансляции («стрима»), благодаря которому одни игроки делятся своими прохождениями, а другие наблюдают за ними.
- Сайт, на котором участники могут поделиться созданными в игре предметами.
- Игровые профили, связывающие все это вместе, с помощью которых легко увидеть

чей-то относительный рейтинг, вклад в игру и т. д.

Такого рода метаигра распространена очень широко, а примеры вроде участия в спортивных состязаниях или общении между поклонниками какого-либо вида спорта известны с незапамятных времен. В мире компьютерных или физических игр им соответствует участие в киберспортивных соревнованиях или просмотр и обмен видео на YouTube или Twitch. Но вне зависимости от средств коммуникации метаигра позволяет нам общаться с теми, кто разделяет наши страсти и увлечения.

Модифицирование игр

«Игра поверх игры» иного рода возникает, когда игроки не удовлетворены существующими правилами или их реализацией. Они могут написать свои собственные «домашние» правила существующей компьютерной игры. Могут решить воспользоваться чит-кодами, отредактировать файлы сохранений или создать скрипт для изменения правил данной игры. Могут воспользоваться сторонними инструментами для модификации содержимого игровой памяти, чтобы изменить параметры программы или получить доступ к заблокированному контенту.

Такого рода метаигра широко распространена в компьютерной экосистеме и часто считается источником удовольствия для игроков, которым хочется «поиграть с игрой» в дополнение к собственно самой игре. Игровые чит-коды и «трениры» (программы, модифицирующие поведение игры) давно стали привычным и обыденным явлением в мире игр для персональных компьютеров²⁰. Некоторые разработчики даже издают официальные руководства по составлению модификаций, а иногда выпускают облегчающие такие модификации средства, вокруг которых образуются целые сообщества игроков, делящихся между собой «модами» к игре.

Но такое воздействие на игру крайне нежелательно в случае многопользовательских игр, где оно может послужить средством обмана и получения незаконного преимущества («читерства»), порчи игрового процесса для других игроков («гриферства») и других видов деструктивного поведения. Особенно вредным модифицирование бывает в случае соревновательных и киберспортивных игр (с некоторыми исключениями, когда игроки заранее договариваются об игре с модами). Консольные платформы вроде PlayStation или Xbox обычно не поддерживают игровые модификации, а соревновательные игры для ПК иногда прибегают к техническим инструментам обнаружения «трениров», сетевых прокси и других средств читерства.

Но процесс с игровыми правилами – старый способ экспериментирования с программами, уходящий своими корнями еще к карточным играм с «домашними договоренностями», действующими вместо официальных правил, или к игрокам на восьмидесятибитных консолях, использовавшим РОКЕ-команды для модификации игровой памяти после загрузки. В современном гейм-дизайне поддержка таких модификаций может благотворно сказаться на играх, делая их более интересными для игроков – при условии, что дизайн игры позволяет осуществлять данные модификации.

Положительные стороны метаигры

Метаигра может оказаться положительной по двум основным причинам. Касательно дизайна она улучшает опыт игрока, а относительно бизнеса – сохраняет интерес к игре и продолжает привлекать к ней новых игроков. Последний аргумент важен тем, что активные

²⁰ При этом различные студии разработчиков относятся к такой метаигре очень по-разному. Например, разработчики «Ведьмака-3» официально не поддерживают никакие чит-коды, тогда как разработчики *The Sims 4* проследили за тем, чтобы их было достаточно, даже указав на своем сайте.

участники помогают игре развиваться. Они рассказывают о ней своим друзьям, которые в результате могут задуматься о том, чтобы приобрести новые колоды карт и другие дополнения.

Это одна из причин, по которым метаигра обычно поощряется не только в играх, но и на платформах. Консоли Xbox 460 и PlayStation 3 ввели системы достижений и таблицы лидеров, появившиеся и на мобильных платформах вроде Google Play и Apple iTunes. Позже консоли Xbox One и PlayStation 4 внедрили новые и легкодоступные способы записи и трансляции геймплея или наблюдения за текущими соревновательными играми, поддерживаемые такими платформами трансляций, как Twitch или YouTube. Все это усиливает социальную метаигру, которая легко может быть как соревновательной, так и кооперативной и творческой. Наконец, платформы для ПК вроде Steam компании Valve официально поддерживают модификации, предоставляя средства технической поддержки для более легкого модифицирования. Все это помогает привлекать игроков к играм и к данным платформам.

Краткий итог

В предыдущей главе мы рассматривали игровые циклы и динамическую структуру, возникающие в результате взаимодействия игрока с системами и друг с другом на протяжении продолжительного времени. В данной главе мы поднялись на еще более высокий уровень и рассмотрели интересные долгосрочные макроструктуры такого рода геймплея. Основные выводы таковы.

- Игры обычно предлагают игроку некий *художественный вымысел*. В одних программах это *определенный сюжет*, в других – общая *фантазия*. Чем больше игра ориентирована на сюжет, тем сильнее она ограничивает *агентивность* игрока (способность влиять на будущее) и предоставляет гейм-дизайнеру контроль над развитием сюжета.

- На самом базовом уровне сюжет можно рассматривать, исходя из классической *трехактной модели*, описывающей крупномасштабную *сюжетную арку* или более мелкие *эпизодические структуры* сюжетов отдельных уровней.

- В ориентированных на сюжет играх типичные схемы повествования могут быть линейные, разветвляющиеся, разветвляющиеся и сливающиеся, а также звездообразные. На основе этих и схожих схем можно строить более крупные композиции. Особое внимание мы уделили играм с открытым миром, объединяющим в себе некоторое количество данных повествовательных схем, обычно закодированных в виде квестов. Они служат неплохим средством авторского контроля над сюжетом, а также усиливают автономию и агентивность игрока.

- В ориентированных на сюжет играх особенно важно понятие *темпа*, а также добавление разнообразия и времени для отдыха, что улучшает опыт игрока. Несюжетные игры также выигрывают от удачной настройки темпа, но чем больше у игрока агентивности, тем сложнее его контролировать.

- За пределами отдельной игровой сессии можно исследовать *метаигру* – то, как игроки взаимодействуют с игрой и друг с другом, даже если они в данный момент не играют. Мы обсудили два конкретных типа: метаигру *мастерства*, сосредоточенную на том, чтобы улучшить свое умение для будущих игр, и *социальную* метаигру, сосредоточенную на получении признания своих действий в процессе или удовольствия от игры в компании других. Также мы вкратце затронули тему *модификаций* и креативной игры с правилами и реализациями.

В следующей, последней главе обратим внимание на процесс прототипирования, благодаря которому гейм-дизайнеры превращают данный набор инструментов и приемов в реальные программы, доступные для игры.

Дополнительная литература

В данной главе мы проанализировали повествование с точки зрения его структурных свойств, но другие подходы, позволяющие сделать различные интересные выводы, не менее познавательны. Очень полезными среди различных источников могут оказаться следующие.

- «Искусство интерактивных историй и видеоигр» (*Interactive Stories and Video Game Art*, SolarSKI, 2017a) предлагает обсудить повествование с точки зрения композиции – как дизайнеры используют визуальные элементы, пространство и передвижение для оформления восприятия окружения и разворачивающейся истории со стороны игрока. В книге приводятся многочисленные примеры из телевидения, кинематографа и других средств коммуникации, а также из игр, которые будут полезны любому, кто изучает гейм-дизайн.

- «Сторителлинг в видеоиграх» (*Video Game Storytelling*, Skolnick, 2014) рассуждает о сюжетах и арках контента с точки зрения писателя, сосредотачиваясь на том, как создать высококачественный нарратив. В качестве более общих советов по «ремеслу» будут полезны описания сценариев, например «Сюжет» (*Story*, McKee, 1997).

- «Словарь гейм-дизайна» (*A Game Design Vocabulary*, Anthropy and Clark, 2014) рассматривает сюжет с точки зрения реакции игрока на выбор и того, как игроки интерпретируют его (или его отсутствие) и возникшие последствия.

- Наконец, хороший источник сведений о взаимодействии арок контента и геймплейных циклов – статья «Циклы и арки» Дэниела Кука (*Loops and Arcs*, Cook, 2012). Что касается темы темпа с точки зрения гейм-дизайнера, то отличным руководством здесь послужит статья Дэвиса «Исследование темпа игры» (*Examining Game Pace*, Davies, 2009), в котором данная тема разбирается на примере *Call of Duty 4*.

Упражнения для индивидуального выполнения

6.1 Художественный вымысел

Назовите одну игру с сильным сюжетом и одну игру с яркой фантазией, но без упора на сюжет. Сравните их.

- а. Как сильный сюжет положительно сказывается на первой игре?
- б. Как яркая фантазия положительно сказывается на второй игре?
- в. Можно ли представить первую игру без сюжета, а просто с элементами художественного вымысла? Что будет, если во второй игре появится сюжет? Сработают ли такие варианты? Как можно будет играть в данные игры?

6.2 Трехактная модель

Выберите хорошо известную вам компьютерную игру.

- а. Опишите интенсивность арки от ранней стадии до конечной.
- б. Согласуется ли она с трехактной моделью – либо как игра с явно выраженной сюжетной аркой, либо как игра без сюжета, но с начальной, средней и конечной стадиями? Опишите, в чем эта игра соотносится или не соотносится с данной моделью.
- в. Делится ли игра на более мелкие элементы, такие как уровни или этапы, в которых имеются свои арки со своими пиками и спадами интенсивности? Объясните свое мнение.

6.3 Схемы повествования

Выберите хорошо известную вам компьютерную игру. Найдите в ней и опишите несколько примеров схем повествования – например, квестов или «миссий».

6.4 Метаигра

Выберите хорошо известную вам программу с метаигрой, которая вам особенно нравится.

- а. Опишите подробно, что именно нравится в данной метаигре.
- б. Основана ли данная метаигра на мастерстве, социальных элементах, на чем-то еще или представляет собой сочетание вышеуказанного?
- в. Предложите один из способов изменить данную метаигру к лучшему.

Упражнения для выполнения в группах

Г6.1 Настольные игры и трехактная модель

В этом упражнении мы проанализируем настольную игру *Settlers of Catan* в контексте игровых стадий и трехактной модели. Предполагается, что у вас есть доступ к данной настольной игре. Вместо нее преподаватель может предложить другую.

Сыграйте в игру *Settlers of Catan* в группе и сделайте видео– или аудиозапись игровой сессии (для этого подойдет даже смартфон с низким качеством микрофона). Просмотрите или прослушайте запись и примерно определите моменты, когда начались и закончились три стадии игры.

- *Начало игры* : когда участники в основном занимаются исследованием и расширением своих владений при скромной торговле.
- *Середина игры* : когда участники сталкиваются между собой, соревнуются друг с другом и вступают в конфликты из-за ресурсов или размещения дорог, поселений и городов.
- *Конец игры* : когда участники сосредоточены на выполнении условий победы (стараясь первыми набрать десять очков).

У этих стадий *нет* четко выраженных границ, поэтому постарайтесь как можно лучше по своим ощущениям определить границы начала/середины/конца, а также понять, что ощущается во время переходов.

Для завершения задания подробно опишите каждую стадию по отдельности.

а. Опишите, когда она началась, когда закончилась, сколько времени продолжалась (в минутах).

б. Опишите свои субъективные впечатления от каждой стадии. Какими геймплейными активностями вы занимались? Используйте такие глаголы, как «соревноваться», «общаться», «исследовать», «обучаться» и т. д.

в. Теперь подумайте об игровой сессии в целом. Создалось ли у вас впечатление, что игра следовала трехактной модели с нарастанием напряженности?

г. Опишите, какие активности игроков менялись со временем, как эти перемены способствовали изменению напряженности.

7 Прототипирование и тестирование

В нескольких предыдущих главах мы исследовали множество различных элементов дизайна – от механик, систем и динамических аспектов геймплея до того, как игроки воспринимают геймплей в контексте своей мотивации. Вернемся же к вопросу, поставленному еще в самом начале: как объединить все эти элементы дизайна, чтобы создать игру?

В данной главе мы опишем процесс создания игры посредством итеративного прототипирования и игрового тестирования. Гейм-дизайнеры редко начинают с создания сразу всего дизайна на бумаге с последующей его реализацией. Так работать над игрой трудно, ведь игры – это динамические системы, часто удивляющие своих творцов. Они

состоят из циклов взаимодействия и обратной связи, поэтому бывает очень трудно предсказать, как именно эти циклы будут сочетаться между собой, когда мы предъявим их реальным игрокам.

Вместо этого дизайнеры, как правило, применяют метод исследований и итераций. Согласно ему мы исследуем какую-то часть игры, реализуем ее, тестируем, оцениваем результаты, делаем какие-либо выводы, исправляем и повторяем цикл. Такие реализации помогают развивать дизайн, подтверждать одни решения и отказываться от других. Данный процесс можно назвать *прототипированием* – продолжающимся созданием прототипов дизайна и включением результатов их тестирования в общую концепцию игры. Он носит итеративный, то есть циклически повторяющийся характер, для него характерно приращивание, то есть постепенное добавление новых элементов. Прототипы при этом эволюционируют от небольших и фрагментарных до более крупных и полных структур, затрагивающих все более обширные части дизайна.

В данной главе мы рассмотрим следующее.

- Стадии разработки коммерческой игры и превращения изначальных идей посредством прототипов в продукт.
- Развитие первоначального замысла идеи в конкретную концепцию, готовую для прототипирования.
- Итеративный рост дизайна игры в ходе создания прототипов и их оценка с помощью игрового тестирования.

Под конец рассмотрим, как коммерческие студии движутся от данной стадии к полному коммерческому релизу, почему для студенческих проектных групп бывает полезно не доводить процесс до такого этапа и что им следует делать взамен.

Мотивирующий пример: Project Highrise

В данной главе в качестве примера для иллюстрации различных стадий производственного процесса мы возьмем игру *Project Highrise*.



Иллюстрация 7.1. Скриншот из игры *Project Highrise*

Project Highrise – это экономический симулятор, в котором игрок играет роль архитектора, строителя и управляющего различных небоскребов. Его задачи заключаются в том, чтобы составить дизайн здания, построить его, управлять арендаторами и, что самое важное, следить за тем, чтобы все были довольны и платили арендную плату, а игрок не обанкротился. При этом у различных типов арендаторов бывают очень разные потребности и

ожидания, влияющие на хорошие отношения друг с другом порой очень неожиданным образом. Задача игрока же заключается в том, чтобы разрешать все конфликты и следить за общим состоянием экосистемы здания.

Возможно, игра *Project Highrise* не настолько хорошо известна, как другие из рассмотренных нами игр, но она очень удачно подходит для целей данной главы. Во-первых, это одна из первых игр, созданных нами в студии SomaSim, которую я основал вместе с Мэттью Вильоне. Используя ее в качестве примера, я могу напрямую делиться деталями процесса разработки. Во-вторых, наши команды разработчиков довольно малы по стандартам индустрии и этим похожи на студенческие группы, поэтому данный пример покажется более подходящим для студенческих команд и независимых команд аналогичного размера.

Стадии производства

Процесс превращения идеи в готовую для доставки игру не стандартизирован и *очень сильно* варьируется – в зависимости от размера компании или команды разработчиков, бюджета или типа игры. Но даже в командах разработчиков разной численности процесс разработки можно поделить на несколько стадий.

- *Концепция* : сначала необходимо сформировать общую *концепцию* игры. Над чем именно будет работать команда? Для кого создается программа? На этой стадии мы выясняем характер игры, определяем ее игроков, решаем, что им здесь понравится и т. д.

- *Прототипирование и предпроизводство* : как только концепция сформирована, мы переходим к размышлениям над подробным дизайном. Для этого требуется более конкретный план геймплея и план определяющих его систем, механик и строительных блоков игры, благодаря которым наши идеи будут воплощены в реальной жизни. Для этого можно создавать небольшие самодостаточные *прототипы* отдельных частей игры, от которых потом можно будет легко отказаться, а также создать более крупный *первый игровой прототип* , объединяющий в себе несколько систем и демонстрирующий их совместную работу. Цель данной *предпроизводственной* стадии – придумать дизайн, в котором мы будем уверены и который будет достаточно подробен, чтобы рассчитать график производства и бюджет.

- *Производство* : как только дизайн, график и бюджет в общих чертах определены, мы принимаемся за работу над уровнями, персонажами, квестами и всем остальным, что необходимо в игре. Производство проходит через свои подстадии, многочисленные альфа– и бета-этапы, заканчиваясь официальным релизом, когда игра готова к доставке игрокам.

- *Релиз и пострелиз* : релиз игры – крупнейшая веха в ее разработке, но работа на этом не заканчивается. После выпуска игры команда вступает в стадию *поддержки* , создает патчи, исправляет ошибки и изменяет баланс в ответ на отзывы игроков, создает дополнительный загружаемый контент и т. д. Но недавняя тенденция выпускать игры в виде версий *с ранним доступом* или *открытых бета-версий* , работа над улучшением которых продолжается даже после продажи, затрудняет определение границы между стадиями производства, релиза и пострелиза.

Все эти стадии важны в коммерческой разработке, а тип работ со временем меняется. Пик дизайнерского труда приходится на ранние стадии, когда доступно огромное количество вариантов. Приходится тщательно отсеивать их, сосредотачиваясь на конкретном дизайне, в чем помогают прототипирование и предпроизводство. Цель стадии предпроизводства заключается в том, чтобы устранить все неопределенности к тому времени, когда команда приступит к производству, чтобы сосредоточиться на работе, собственно над игрой, не занимаясь поисками ответов на фундаментальные вопросы дизайна.

В последующих разделах мы сосредоточимся на начальных стадиях, концепте,

прототипировании и предпроизводстве, поскольку они играют центральную роль в дизайне и разработке игры. На последующих стадиях роль гейм-дизайнера снижается, поэтому мы разберем ее лишь вкратце. В конце главы поговорим, как принципы прототипирования и предпроизводства можно использовать в студенческих проектах и семинарах.

Концепция игры

Самая первая стадия работы – разработка концепции игры. Каждая игра начинается с какого-то общего представления, изначального зерна, идеи, но это лишь самое начало. Прежде чем переходить к механикам и системам, нужно иметь довольно ясное представление о том, дизайном чего *именно* мы собираемся заниматься.

Итак, сначала нужно оформить концепцию и подумать о деталях, чтобы понять, над чем придется работать. Стадия концепции обычно включает в себя несколько связанных между собой целей.

- *Понять и определить* идею в достаточных подробностях, чтобы получить представление об общем замысле и понять, что будет нужно создать.
- *Определить коммерческую значимость* идеи – например, понять, кто будет аудиторией, насколько велик рынок, сколько копий можно будет продать и т. д.
- *Объяснить идею* – донести ее до других людей вне команды, особенно до тех, от кого зависит финансирование проекта.

Понимание идеи игры

На первой стадии разработки команде необходимо сообща определить замысел игры, над которой она будет работать. Иногда это бывает легко – у всех может быть четкое представление о том, что именно они хотят сделать. В иных случаях у разных членов команды могут быть свои представления об идеях, целях и замысле – тогда приходится много общаться и договариваться.

На данной стадии гейм-дизайнер сосредотачивается на опыте игрока и на общей идее геймплея. Как говорилось в главе 2 «Опыт игрока», дизайнеру необходимо понять мотивацию игроков различных типов и попытаться определить, кто будет участниками его игры, что они захотят делать в ней и почему.

Можно задавать себе следующие вопросы.

- *Каков художественный вымысел в игре?* Объясните, роль кого будет играть игрок, какое влияние он будет оказывать на игровой мир, как туда попадет.
- *Что представляет собой геймплей?* Объясните, чем игрок будет заниматься в игровом мире, какие он будет выполнять виды деятельности – как краткосрочные, так и долгосрочные.
- *Кто наши игроки?* Объясните мотивацию игрока – не как персонажа игры, а как реального человека, участвующего в игре – какой мотивацией он будет руководствоваться.
- Например, при разработке *Project Hihgrise* мы описали нашу концепцию следующим образом.
 - *Художественный вымысел* . Игрок – архитектор, строитель и управляющий небоскребов. Он создает дизайн высотных зданий и строит их, привлекает арендаторов, управляет ими, следит за тем, чтобы все были довольны, получает прибыль.
 - *Геймплей* . На малой временной шкале самые быстрые циклы включают в себя строительство этажей, привлечение арендаторов, а затем сбор арендной платы и использование накопленных средств для дальнейшего строительства или улучшения имеющейся постройки, добавления новых этажей, предоставления услуг, решения проблем и т. д. по необходимости. Более крупные циклы подразумевают попытки сохранения

экономического баланса, привлечение большего количества престижных арендаторов, которые платят больше денег. Для этого, в свою очередь, приходится накапливать деньги для дорогих и широкомасштабных улучшений и превращения зданий в более престижный вид недвижимости.

• *Типы игроков*. Цель участника – построить сложную систему небоскреба и управлять ею. Она соответствует нескольким типам мотивации: стратегические и тактические вызовы по управлению хаотичной экономикой и получению прибыли, что характерно для экономических симуляторов; исследование и эксперименты с экосистемой, наблюдение за ее поведением и творческое самовыражение – возможность спроектировать небоскреб, дизайн которого отражает стиль и эстетические предпочтения игрока. Мы понимаем, что создаем игру, удовлетворяющую различные типы мотивации в категориях стратегии, вызова, исследования и дизайна.

Обратите внимание на то, что здесь уже упомянуты некоторые подробности – указан жанр (экономический симулятор) и некоторые системы и механики (строительство, денежные циклы, расширение, престиж). Это нормально. Описание не должно быть чисто абстрактным, а просто достаточно кратким, чтобы не увязать в деталях. Наша цель – сделать его настолько конкретным, чтобы все члены команды понимали, в чем заключается игра, но не перегружать подробностями.

Понимание рынка

Если предполагается коммерческая продажа игры (в противоположность студенческим или некоммерческим проектам), то крайне важно предсказать ее поведение на рынке – насколько хорошо она будет продаваться, в каком количестве копий, по какой цене и т. д. Это трудная задача, точные предсказания здесь невозможны. Игры – это эстетические продукты, и реакция игроков может широко варьироваться в зависимости от того, как игра ощущается и выглядит после окончания работы над ней.

Но на данной стадии мы можем составить ориентировочные планы и убедиться, что мы не идем по неверному пути, попытавшись ответить на следующие вопросы.

• *Каков рынок для этой игры?* Если мы разрабатываем экономический симулятор (в отличие от таких ориентированных на массовый рынок игр, как шутеры), то сколько игроков заинтересуются ею? Нишевая ли это игра с ограниченной привлекательностью или же она привлечет более широкую аудиторию?

• *Какова наша конкуренция и насколько она велика?* Новичкам бывает тяжело заинтересовать игроков, если на рынке уже имеется множество игр подобного типа (например, экшен-платформеров), платформа насыщена (как магазины мобильных игр) или большинство игроков сильно увлечены единственной игрой данного направления (например, *World of Warcraft* среди массовых многопользовательских RPG в стилистике фэнтези).

• *Как игра будет держать позиции с учетом конкуренции?* В игре должно иметься нечто, отличающее ее от остальных, а данный аспект должен представлять значимость для аудитории. Это могут быть геймплей, эстетика, художественный вымысел или что-то совершенно иное, но оно должно находить отклик у игроков, которых мы пытаемся привлечь.

• *Насколько велика потенциальная аудитория?* Если мы разрабатываем игру для ОС Windows, то потенциальных игроков будет больше, чем для разработки MacOS или Linux. Схожими соображениями руководствуются, создавая игры для популярных консолей вроде PlayStation или Xbox по сравнению с менее популярными. При расчетах необходимо учитывать размер аудитории платформы и ее потенциал роста.

• *Каких продаж и цен на единицу продукции стоит ожидать?* Следует обратить внимание на продукты-аналоги (вроде игр со схожими геймплеем и стоимостью) и на то,

насколько хорошо они продаются, чтобы получить некоторое представление, насколько хорошо будет продаваться данная игра, если все пойдет хорошо.

Мы проделали это в отношении *Project Highrise* и получили следующие результаты.

- Составили список других игр с концепцией, подобной нашей (схожего жанра или стиля), затем поискали сведения о том, как они демонстрировали себя на рынке.

- Воспользовались рыночными сравнениями для выбора платформы (компьютерные игры для ПК и Mac) и для определения, удастся ли продать достаточно копий, чтобы оправдать затраты. Наконец, рыночные аналоги помогли нам определиться с ценой по сравнению с другими играми данного стиля и планкой качества.

- Рассмотрели различные идеи о том, чем могла бы выделиться наша игра. Мы сосредоточились на жанре, который был недостаточно представлен на тот момент; выбрали довольно своеобразный и сразу узнаваемый стиль в духе середины двадцатого века; решили, что тематика игры должна быть знакомой для игроков предыдущих экономических симуляторов для ПК.

Это лишь некоторые идеи, которые следует иметь в виду на стадии концепции. Существует множество других способов предсказаний рыночного поведения. Гораздо более подробные рыночные исследования описываются в работе Фаттера (Futter, 2017), но они никогда не дадут точных ответов. Цель здесь не столько в том, чтобы предсказать лидирующее положение на рынке (что невозможно), сколько в том, чтобы предотвратить очевидные провалы.

Создание коммерческого предложения

После утверждения и обоснования концепции бывает полезно обобщить результаты в одном небольшом документе – *коммерческом предложении* (заявке), кратко и ясно описывающем особенности игры.

Коммерческое предложение чаще всего используют для обеспечения финансирования. Этот документ предъявляют потенциальным издателям или инвесторам, чтобы они заинтересовались концепцией игры, поэтому он должен быть кратким и понятным. Поскольку его адресаты – это люди извне по отношению к команде разработчиков (например, потенциальные инвесторы), то обычно для сравнения указываются другие игры, чтобы можно было быстро прикинуть, насколько данная концепция похожа или не похожа на имеющееся на рынке.

Единого формата для коммерческого предложения не существует, а у разных издателей или инвесторов могут быть свои ожидания. Тем не менее среди общих элементов коммерческого предложения можно выделить следующие.

- *Краткая презентация* («эlevator питч») – быстрое сообщение секунд на десять о том, что представляет собой игра и чем она интересна. На составление правильной презентации может уйти немалое время, но она очень полезна для того, чтобы аудитория поняла суть концепции.

- *Краткое изложение дизайна* – один или несколько абзацев, кратко излагающих фантазию или сюжет игры и описывающие предполагаемый геймплей.

Некоторым командам нравится сводить краткое изложение дизайна к его *основным аспектам* – нескольким уникальным, центральным для игры особенностям, без которых она немислима. Например, для *Minecraft* в качестве таких основополагающих аспектов можно назвать случайную генерацию мира, добычу ресурсов и их использование в крафтинге, создание фантастических построек и игру с друзьями на одной карте. Они помогают читателям сравнить программу с тем, что уже известно, помогая донести до них центральную идею игры.

- *Рыночное сравнение* – для коммерческих игр это будут результаты анализа рынка –

описание целевых платформ, размер аудитории, ценовая политика и т. д. Здесь также полезны сравнения с другими играми, включая рейтинги, вроде обзоров в Steam или оценки на сайте Metacritic.

В коммерческие предложения для инвесторов часто включают соответствующие бизнес-данные – размер команды, предполагаемые расходы на разработку и т. д. Более подробные сведения о структуре и формате коммерческих предложений можно получить в работах, указанных в разделе «Дополнительная литература».

Коммерческое предложение бывает полезным и для внутренних целей – оно помогает довести концепцию игры до всех членов команды и настроить их на одну волну. Во время разработки у людей будут появляться новые идеи, так что краткий дизайнерский документ сможет напоминать, над чем они работают и почему, помогая определить, какие новшества следует включить в проект, а от каких стоит отказаться.

От концепции до прототипирования: основные принципы Келли

Как узнать, сформулирована ли концепция достаточно четко, чтобы приступить к разработке дизайна? В различных коллективах имеются свои традиции и способы ответа на данный вопрос. Тем не менее особо полезными могут оказаться рекомендации Тайга Келли, перечисленные в его презентации «Семь смертельных вопросов» (Kelly, 2014) и относящиеся к командам разного размера и к различным типам игр. Эти эвристические принципы призваны сделать процесс прототипирования и разработки более четким и ясным, но без необходимости составления большого количества предварительной документации.

Работая консультантом по дизайну, Келли заметил, что во время разработки команды легко отвлекаются и теряют представление о первоначальном замысле по двум причинам: во-первых, потому что о гейм-дизайне трудно говорить четко и недвусмысленно, так что со временем концепции размываются; во-вторых, потому что в процессе разработки и игрового тестирования неизменно появляются новые идеи, которые хоть и бывают грандиозными, но все равно отвлекают. Команды, которые отвлекаются, теряют фокус и забывают о первоначальном замысле, поэтому бывает трудно получить в конце целостную и связную игру.

Чтобы предотвратить такую ситуацию, Келли предлагает команде еще на стадии концепции ответить на семь вопросов. Предполагается, что они заменяют полную и формальную дизайнерскую документацию. Ответить на вопросы легче, они допускают большую гибкость, но отражают ядро замысла и становятся общим основанием, напоминающим всем, что за игру они создают. В дальнейшем данные вопросы можно использовать для отсеивания любых идей, появляющихся во время прототипирования и разработки. С их помощью можно проверить любую новую мысль, а если она не подкрепляет первоначальный замысел, то от нее, возможно, лучше избавиться.

Семь вопросов Келли таковы²¹.

1. *В чем заключается конфликт в игре?* Иными словами, какая в ней наблюдается борьба, какой присутствует вызов, с чем сталкивается игрок как с точки зрения геймплея, так и с точки зрения художественного вымысла? В игре должна быть определенная напряженность – то, что захватывает внимание и воображение игрока и делает процесс интересным.

В играх вроде «*Тетриса*» конфликт создают неизменно падающие с увеличивающейся

²¹ Данные вопросы были переформулированы с использованием словаря, которым мы пользовались в этом тексте, потому что в оригинальной форме «глаголы» и «действия» использовались в них в несколько ином смысле. Приносим особые благодарности Келли за дополнительные разъяснения по электронной почте.

скорости блоки, которые игрок должен упорядочивать. В *Minecraft* игровой мир опасен, игрок должен все самостоятельно строить и уметь выживать. *The Sims* отражают знакомую повседневную жизнь взрослых людей, которым приходится как-то сводить концы с концами.

2. *Какова работа игрока?* Здесь подразумевается не роль игрока в сюжете и не повествовательная работа. Это скорее краткое определение, желательно в одной фразе или даже в одном слове, задач и действий игрока.

В «*Temple*» работа игрока – «упорядочивать фигуры». В *Tomb Raider* игрок – «искатель приключений». В *Rome: Total War* он «полководец». Главное преимущество такого определения состоит в том, что оно помогает гейм-дизайнеру понять, что требуется от игрока. Игрок в *Rome: Total War* понимает, что будет командовать войсками и вряд ли его заставят заниматься микроменеджментом, управляя отдельными воинами и следя за тем, насколько они довольны.

Описание работы также помогает гейм-дизайнеру бороться с искушением перегрузить игрока различными видами работ. Иногда в ходе разработки бывает очень соблазнительно навесить на игрока дополнительные задания и действия, но для игр лучше, если они предлагают участнику только одну работу или небольшой набор связанных между собой задач.

3. *Каковы устройства ввода игрока?* Использует ли он контроллер, клавиатуру с мышью или сенсорный экран? Как игровые действия соотносятся с кнопками, клавишами, джойстиком, зонами сенсорного экрана и т. д?

Выбор устройства ввода очень сильно влияет на выполнение различных задач в игре. У разных устройств разные возможности. Например, сенсорный экран позволяет очень легко подбирать и передвигать объекты на игровом экране, что гораздо труднее делать с помощью контроллера, но при этом контроллер допускает очень сложную навигацию в виртуальном пространстве. При этом запомнить разнообразные сочетания небольшого количества клавиш контроллера для совершения разнообразных действий бывает весьма трудно, поэтому в игре необходимо предусмотреть отличную систему ввода и хорошее обучение.

4. *Можно ли расширять действия игрока?* Этот вопрос касается основных механик. Каковы они, меняются ли со временем? Можно ли их совершенствовать, модифицировать или сочетать друг с другом, чтобы они порождали более сложное поведение?

Например, в военной игре основные действия подразумевают передвижение войск, нападение и защиту. Но когда юниты совершенствуются или их показатели разблокируют новые виды атаки и защиты, это может привести к более сложному геймплею, предусмотренному гейм-дизайнером. Или же в файтинге различные виды атак могут сочетаться и производить «комбо» с новыми интересными характеристиками. Действия, которые можно модифицировать и развивать, явно лучше тех, что остаются неизменными на протяжении всего процесса.

5. *Имеются ли в игре три (или более) типа ресурсов?* Келли обращает внимание, что даже в такой простой спортивной игре, как теннис, уже имеются четыре типа ресурсов. Теннисная ракетка как снаряжение со своими характеристиками; теннисный мяч как «жетон», которым обмениваются игроки; такие статистические показатели, как счет или «сет»; контроль над территорией в том смысле, что положение игрока на корте имеет огромное стратегическое значение.

Среди троек типов ресурсов в известных играх можно назвать «оружие, патроны, здоровье» в *Quake* или «земля, заклинания, очки жизни» в *Magic the Gathering*. Важно, чтобы игры имели не просто варианты одного типа ресурса, а варианты типов ресурса (в играх, не основанных на управлении ресурсами) и варианты типов механик. Наличие различных типов со своими плюсами и минусами делает игру более интересной для игрока, чем просто добавление большего количества элементов одного типа.

6. *Имеется ли в игре основополагающее правило?* Хотя в играх может быть множество разных правил и механик, во многих успешных играх имеется одно или несколько самых характерных, основополагающих правил, уникальных для конкретной игры или жанра, без

которых она была бы уже не та.

Часто они подразумевают уникальные ограничения или компромиссы. В *шахматах* основополагающее правило – правило шаха и уязвимость короля, которое из него следует. В такой игре, как *FarmVille*, это механики выбора участков, высаживания чего-либо и последующего сбора урожая. В игре *Ico* – необходимость защищать следующего за игроком персонажа-спутника. Если удалить данные правила, то игры потеряют свою особенность и станут другими. Наличие такого рода уникального принципа, охватывающего все части геймплея, предоставляет игроку новые сложности, которые он вынужден преодолевать, тем самым выделяя данную игру среди других.

7. *Что вы тестируете?* Игры бросают участникам самые разные вызовы, но в них должен быть один, центральный для данной игры главный вызов или очень мало других, но конкретных и дающих основание для действий. «*Temris*», например, проверяет способность игрока сортировать фигуры разной формы в условиях недостатка времени. Файтинги вроде *Tekken* тестируют способность выполнять сложные движения и атаковать с идеальным таймингом. Градостроительные симуляторы вроде *SimCity* проверяют способность управлять ресурсами и распределять пространство. Игрок должен осознавать, что от него требуется, получая обратную связь о своих действиях, чтобы избежать непонимания и разочарования.

Ответы на эти и аналогичные вопросы помогут команде четко определить замысел игры и подготовиться к его реализации. Они также помогут отсеять и распределить по приоритету новые идеи, неизбежно появляющиеся в ходе прототипирования.

Прототипирование

Как только концепция примет четкую и ясную форму, пора переходить к работе над дизайном деталей. Именно в этот момент мы переходим от высокоуровневого замысла к подробным характеристикам геймплея и связанным с ним системам и механикам.

Эта стадия называется «предпроизводственной», потому что предшествует полноценному производству игры. Цель предпроизводства – составить подробный дизайн, поэкспериментировать с замыслом и его реализацией, устранив разнообразные неопределенности, чтобы вступить в стадию производства с четкой идеей, что необходимо сделать, сколько времени это займет и сколько человек потребуется.

Как говорилось выше, нередко бывает так, что команда разработчиков сначала составляет весь дизайн, а затем разом реализует его, надеясь на то, что игра будет работать. Мы создаем динамические системы, чье поведение трудно предсказать, поэтому некоторые сюрпризы здесь неизбежны. Взаимодействие между системами бывает хаотичным и непредсказуемым, особенно если они состоят из подпитываемых друг другом циклов обратной связи. Но наибольший источник сюрпризов – это игроки. Они любопытны, умны, активны и обычно предпринимают действия, которые гейм-дизайнер никак не мог предвидеть.

Вместо последовательной реализации заранее составленного четкого плана мы предпочитаем рассматривать дизайн и его реализацию как периодический, циклический и итеративный процесс. В определенный промежуток времени мы занимаемся дизайном какой-то части программы, реализуем ее, тестируем и оцениваем, а затем на основании полученных результатов совершенствуем дизайн и повторяем этот цикл.

«Играбельные» прототипы

Есть два подхода к тестированию дизайнерских идей: создание небольших *индивидуальных прототипов* разных частей игры или создание *накопительного прототипа*, вбирающего в себя различные элементы дизайна. При необходимости команда

разработчика может совмещать два этих подхода.

Индивидуальные прототипы представляют собой быстрое воплощение отдельных дизайнерских идей для тестирования отдельных особенностей. Хорошим примером может быть прототип боевой системы файтинга или прототип систем крафтинга в ролевой игре. Они позволяют проверить, доставляет ли удовольствие конкретная система или набор механик независимо от всего остального.

Такого рода прототипы обычно отвечают только на некоторые конкретные вопросы, но от них и не ожидают окончательного решения или того, что они будут включены в конечный продукт. По этой причине их часто создают как *одноразовые* – наспех собранные, без оптимизации, без следования принятым практикам хорошего программирования и с использованием уже имеющихся кода и инструментов (например, при помощи фрагментов кода предыдущих проектов, временных визуальных инструментов и т. д.). Единственное, что при этом важно, – доказать или опровергнуть какие-то конкретные теории о частях дизайна, а затем отказаться от этого прототипа и двигаться дальше.

Иной подход заключается в том, чтобы создать накопительный прототип, состоящий из разных частей игры и тестирующий их общую работу. Такой подход чаще встречается при разработке игр с большим количеством систем – военных играх или экономических симуляторах, поскольку они очень чувствительны к качеству интеграции различных систем, а не только к качеству отдельных составляющих элементов. Этот же подход мы использовали при разработке *Project Highrise* – игры с очень большим количеством систем.

Из-за размеров накопительного прототипа и количества необходимой для его создания работы команда разработчиков обычно делает их не одноразовыми прототипами, а скорее небольшими демоверсиями, со временем превращающимися в настоящую игру. Причины этого носят как экономический, так и технический характер. Разработчики вкладывают в накопительный прототип много времени и усилий, поэтому им не хочется отказываться от него и начинать все с самого начала. К тому же, согласно распространенному среди программистов мнению, более качественный продукт получается, если пытаться исправлять плохую, но функциональную программу, а не переписывать ее с нуля (Spolsky, 2000).

Когда команда решает создать накопительный прототип, в данном случае можно использовать временные фрагменты кода и инструменты, которые впоследствии возможно заменить. Но при этом технические аспекты нужно реализовывать с оглядкой на будущее (придерживаться общепринятых правил программирования, делать код расширяемым, минимизировать потребности в последующих исправлениях и т. д.). В результате такой прототип может превратиться в *«играбельное» демо*, то есть в демонстрацию геймплея, которая при этом не охватывает весь контент или художественно-оформительские аспекты игры.

В любом случае на этой стадии ради скорости итераций лучше отказаться от желания создавать идеально выверенные визуальные элементы. Например, если игра основана на уровнях, то обычно на первых стадиях используют так называемую *технику белой коробки*, то есть уровень без текстур и детального графического оформления. Так легче работать над исправлениями до тех пор, пока не будет разработан нужный пространственный дизайн (SolarSKI, 2017a, 7).

Независимо от того, какой подход или их сочетание выбрала для себя команда, процесс в любом случае схож – мы тестируем различные дизайнерские идеи и наблюдаем, как они взаимодействуют друг с другом во время прохождения игры.

Итеративный процесс

Разработав концепцию, мы начинаем цикл создания прототипов и их оценки. Для того чтобы процесс было легко переделать под конкретные особенности, типичная последовательность шагов может выглядеть следующим образом.

• *Концептуальный дизайн* . На первом шаге мы сосредотачиваемся на некоторых элементах игры, разрабатываем более подробный дизайн, составляем план того, что нужно будет сделать. На основе наблюдений можно добавлять новые системы или изменять существующие элементы и т. д. На данном этапе продумывается дизайн частей и по возможности прорабатываются все детали.

• *Реализация дизайна* . Мы создаем «играбельную» версию новых элементов дизайна. Это может подразумевать их добавление к уже существующему прототипу, который с каждой итерацией становится сложнее и ближе к игровому демо. Но некоторые тесты проводить легче, оценивая отдельные одноразовые прототипы.

• *Игровое тестирование реализации* . Мы ставим себе цели в зависимости от того, что хотим протестировать. Например, работает ли наша система так, как задумывалось? Получают ли игроки от нее удовольствие? Понимают ли игроки, в чем она заключается и как с ней взаимодействовать? Настоящий тест происходит, когда мы предъявляем прототип игрокам, заставляем их взаимодействовать с введенными в него новыми элементами и наблюдаем за тем, что происходит.

Аналоговые и цифровые прототипы. Для ускорения прототипирования иногда возникает желание отойти от создаваемых на компьютерах цифровых прототипов и обратиться к *аналоговым* или *бумажным* вариантам, реализуемым в физических объектах вроде карт, досок для настольных игр, фишек и игровых денег. Создавать бумажные прототипы бывает действительно быстрее цифровых. Гораздо проще распечатать и вырезать бумажные карты или сделать игровое поле из картона, чем реализовывать тот же дизайн при помощи игрового движка. Аналоговые прототипы популярны в классах благодаря скорости их создания, к тому же новичкам с ними легче работать.

Тем не менее в индустрии аналоговые прототипы используются не так часто. Основной недостаток их в том, что они отлично подходят для прототипирования игр с механиками, схожими с карточными или настольными играми, но не очень хорошо подходят для других типов, таких как игры, требующие физической реакции или скрытых механик. Представьте, например, бумажный прототип аркадного файтинга. Он совершенно не помог бы нам ощутить геймплей, определиться с таймингами, выбрать кнопки управления или «комбо» и т. д. В качестве другого примера возьмем игру *SimCity* , которая в очень большой степени полагается на скрытые механики, которыми игрок овладевает со временем и которые создают сложную симуляцию состояния мира. Попытка воссоздать такую симуляцию в таком же масштабе с помощью бумаги и игроков-людей обречена на провал.

Но бумажные прототипы могут послужить отличным инструментом, если цифровые игры напоминают аналоговые. Особенно хорошо они подходят для прототипирования коллекционных карточных игр, например *Hearthstone* или *Gwent* («Гвинт»), потому что в данном случае очень легко самим сделать карты для игры с друзьями, а также для прототипирования тактических и военных игр с самодостаточными явными механиками. Brathwaite and Schreiber, 2009 – отличный источник дополнительных сведений об аналоговом прототипировании.

• *Оценка игрового тестирования* . Мы критически анализируем результаты тестирования и решаем, насколько дизайн и его реализация способствовали достижению поставленных нами целей для данной итерации, как они способствуют достижению общих целей игры.

• *Модификация дизайна* . Смотрим, как результаты повлияли на разработку дизайна и на другие элементы, дизайн которых уже разработан и реализован.

Затем возвращаемся к первому шагу и повторяем все шаги снова и снова!

Такое прототипирование намеренно носит в высшей степени итеративный характер. С каждым циклом мы повторяем одни и те же шаги – планируем, делаем, играем, оцениваем, а

затем принимаем решения относительно последующих действий.

Этот процесс схож с прототипированием в других дисциплинах, таких как промышленный дизайн или дизайн программного обеспечения (Brooks, 2010). В сфере разработки данного направления он описывается моделью, называемой *спиралью Боэма* (Boehm, 2000). На иллюстрации 7.2 показано, как эта спираль могла бы выглядеть, будучи адаптированной к прототипированию игр.

Спираль начинается с разработки некоторых аспектов дизайна игры, которые мы реализуем и тестируем. Если тест проходит успешно, сохраняем данный элемент и реализуем его в такой игре, какова она есть на данный момент, возможно, вместе с некоторыми дополнительными идеями по его дальнейшему улучшению. Затем разрабатываем дизайн другого аспекта, реализуем его, тестируем и т. д. Таким образом, игра с каждым успешным игровым тестом растет, а наш прототип движется в сторону первой качественной «играбельной» сборки.

Несмотря на то что данная модель обрела широкое признание в сфере разработки программного обеспечения, сама по себе она относительно недавнее изобретение. Спираль Боэна пришла на смену другим моделям разработки, в которых различные части программного обеспечения разрабатывались отдельно друг от друга (например, в виде различных функциональных модулей), а затем интегрировались. Но такой подход печально прославился долгими и трудными стадиями интеграции, в течение которых приходилось решать различные проблемы несовместимости и устранять недостатки. Сейчас мы знаем, что трудно предусмотреть все особенности и характеристики модулей настолько подробно, чтобы точно предсказать их поведение при интеграции.

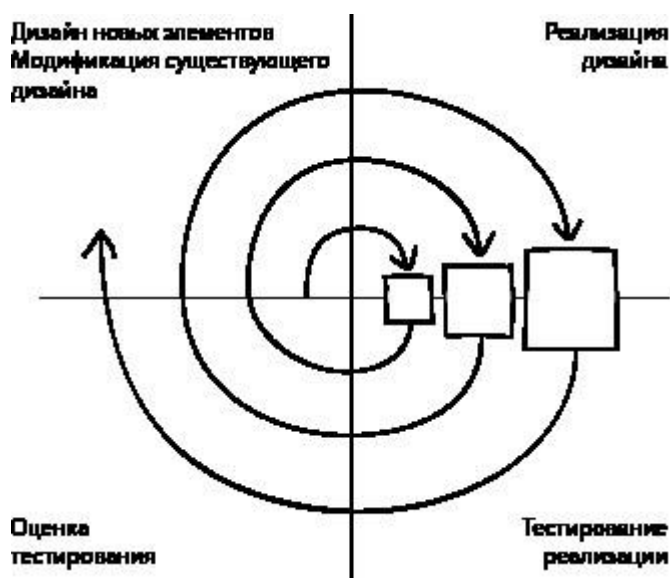


Иллюстрация 7.2. Схема прототипирования игры, построенная на основе спиральной модели Боэма, при которой игровой прототип с каждой итерацией растет и улучшается

В отличие от прежних методов итеративный ставит во главу угла именно интеграцию. Различные части продукта создаются вместе, а не по отдельности. Основное преимущество данного процесса – постепенное сокращение рисков и увеличение надежности дизайна. Каждый раз проходя через цикл, мы не только добавляем что-то новое, но и убеждаемся, что уже созданные нами части прекрасно работают вместе. А если окажется, что это не так, то можно понаблюдать за ними во время игрового тестирования и исправить.

Игровое тестирование

На ранних этапах разработки игру тестируют сами дизайнеры и разработчики. Это легко, потому что их не нужно учить, как играть в свою игру, что особенно важно, когда

интерфейсы и средства управления только разрабатываются, а игра кажется непонятной посторонним. Позже команда может расшириться, а к тестированию будут привлечены родные и знакомые или разработчики из другой команды. Наконец, на поздних этапах производства команда может нанять сторонних тестировщиков, прежде никогда не игравших в эту игру, чтобы взглянуть на нее глазами новых игроков.

Проведение ряда удачных игровых тестов означает попытку взглянуть на игру глазами других. От наблюдателей требуется сохранять объективность по отношению как к игре, так и к игрокам. Самим разработчикам данная стадия может показаться довольно неприятной, потому что трудно оставаться беспристрастным, когда видишь, как люди реагируют на проблемы твоей незаконченной игры. Но это очень важная часть процесса, потому что наблюдение за реакцией игроков позволяет находить *неизвестные* проблемы.

Вот некоторые эвристические принципы игрового тестирования.

- При игровом тестировании игрокам необходимо позволять играть так, как они сочтут нужным, не поправляя их и не говоря, что нужно делать. Сохранять спокойствие бывает сложно, потому что игроки неизбежно допускают ошибки или не понимают, что нужно делать, но обучение – это часть геймплея. Как бы нам ни хотелось поправить их, нужно просто наблюдать. Слегка подтолкнуть можно только в исключительных ситуациях – например, когда игрок застрял или встретился с багом, но таких ситуаций следует избегать.

- Очень полезно бывает попросить игроков проговаривать вслух, что они делают и какие ощущения при этом испытывают. Такие отзывы дают представление о том, как игра воспринимается глазами игрока и какого рода опыт он переживает. Игра может увлечь участников настолько, что они перестанут говорить. Это хороший знак, но все равно нужно предлагать им высказывать свои мысли.

- Очень полезно отмечать слова игроков, но не следует воспринимать их буквально. Вместо этого нужно обращать внимание на действия и эмоции участников, передающие их ощущения. Когда игрок говорит: «Этот враг слишком тяжелый», он не утверждает, что уровень сложности врага действительно высок. Скорее, он говорит о своем разочаровании, потому что не может справиться с врагом. В зависимости от контекста решения бывают разными. Можно усилить оружие игрока, а можно подождать, пока он научится разбираться с такими врагами, повысив свои навыки, и т. д. Иногда правильным вариантом бывает и вовсе ничего не делать.

- Наконец, мы учитываем не только удачи, но и неудачи игроков. Научились ли они тому, чему должны были научиться? Действуют ли ожидаемым образом? Удовлетворили ли вы цели игроков?

Для того чтобы игра полностью поглотила участников, разработчики обычно стараются сделать окружение, в котором те играют, как можно более комфортным и ненавязчивым. В больших студиях разработчиков или в отдельных компаниях по игровому тестированию бывают специально оборудованные помещения, в которых одна часть команды не может общаться с другой напрямую, а разработчикам доступно только наблюдение за реакцией тестировщиков через веб-камеру или через полупрозрачный экран. Небольшие команды разработчиков могут предпочесть менее формальный подход – предложить опробовать игру посетителям игровых фестивалей или даже позволить тестировщикам поиграть дома с помощью специальных программных инструментов, отсылающих данные о прогрессе тестирования.

В результате каждой игровой сессии регулярно появляется огромное количество записей и заметок, описывающих различные ошибки и проблемы – от небольших (вроде предложений по улучшению пользовательского интерфейса, небольших «багов» и т. д.) до более крупных, оказывающих влияние на всю игру (например, замечание о том, что вся система не работает так, как предполагалось).

Все это огромное количество сведений нужно первым делом рассортировать в

зависимости от важности. Рассортируем их по трем категориям.

1. Список «Сделать»: вопросы, представляющие собой очевидные проблемы или требующие важных улучшений. В эту категорию попадает большинство «багов» и отзывы о явных неполадках со стороны игроков. Их следует устранить незамедлительно.

2. Список «Рассмотреть»: вопросы, которые необходимо исследовать – предложения со стороны игроков или идеи, которые появились у разработчиков во время игрового тестирования. При этом неясно, как данные изменения скажутся на игре в дальнейшем. Такие вопросы потребуют дополнительного внимания, чтобы решить, как с ними поступить.

3. Список «Отложить»: соображения, появившиеся во время игрового тестирования, которые не нужно пока подробно рассматривать, либо не повторяющиеся на данный момент проблемы. В качестве примера можно привести негативную реакцию на временное художественное оформление, которое потом все равно будет изменено, или случай, когда игрок не заметил какую-то важную информацию и пришел к неправильному умозаключению. Такие замечания следует отслеживать, но на них не нужно реагировать, пока их не выскажет достаточное количество игроков – в данном случае будет необходимо уделить им дополнительное внимание.

Такая сортировка необходима, потому что не все предложения одинаково важны, а различные идеи требуют разного количества времени или сопряжены с разнообразными последствиями. После сортировки можно распределить все вопросы по приоритету и при необходимости внести их в рабочий график.

Но такие решения не следует принимать во время игрового тестирования. В данном процессе мы просто записываем все замечания, не сортируя их. И только потом, просматривая список, распределяем по категориям и решаем, что делать в следующей итерации прототипирования.

Дизайнерская документация

На стадиях концепции и прототипирования разработчики создают более подробное описание дизайна, достаточно детализированное, чтобы им можно было делиться с другими членами команды.

Эти детализированные описания в итоге образуют *дизайнерский документ*. Данный термин в каком-то смысле можно назвать анахронизмом, отражающим практику прошлого, когда составлялся один большой документ, подробно описывающий различные элементы игры. Но в современных студиях все чаще пользуются дизайнерской документацией в виде посвященной проекту страницы в «вики» или онлайн-инструментами (такими как Google Drive), содержащими различные медиафайлы (не только текстовые документы, но и концепт-арт, скриншоты, видеоклипы и т. д.), которые можно быстро обновлять в ходе итераций дизайна.

В более крупных коллективах возрастает роль коммуникации, которая должна быть хорошо задокументированной и точной, чтобы все имели одинаковое представление о том, над чем они работают, а усилия были скоординированы. В командах из десятков или сотен человек коммуникация затрудняется, поэтому документация помогает проследить, чтобы общую концепцию игры понимали не только дизайнеры, но и специалисты из смежных дисциплин (программисты, маркетологи и т. д.).

Но в небольших коллективах общаться напрямую быстрее и легче, поэтому часто не стоит тратить время на составление подробной документации. Например, над игрой *Project Highrise* работало настолько небольшое количество человек, что никто не испытывал трудностей в общении. Оказалось, что для деловой коммуникации достаточно инструментов Google Docs, которыми все изначально пользовались для прототипирования в ходе разработки общего замысла. Но как только процесс пошел по-настоящему, мы перестали

пользоваться данными документами и переключились на видеочат, электронную почту и списки задач.

Здесь мы не будем углубляться в подробности, но в разделе «Дополнительная литература» указаны источники информации о дизайнерской документации, ее структуре и использовании.

Окончание итераций

В ходе итеративного процесса мы постоянно возвращаемся к своему дизайну, улучшаем его, вносим изменения, тестируем и на основании теста делаем предположения о дальнейшем улучшении. С каждым циклом спирали наши дизайн и «играбельная» сборка обретают все более законченный вид. Но когда же следует остановиться?

К счастью или к сожалению, четкого ответа на данный вопрос не существует. Очень редко бывает, чтобы мы полностью были удовлетворены результатами, достигли именно того, что хотели получить, не желая больше ничего улучшить. В ряде случаев «наилучший» дизайн – понятие условное. Всегда можно внести какие-то изменения или попробовать реализовать какую-либо идею.

Чаще всего мы останавливаемся тогда, когда необходимо остановиться – получив дизайн, который признан достаточно удовлетворительным, прежде чем у нас закончатся время и ресурсы. Но не следует воспринимать это как нечто обязательно плохое. Ограничения заставляют нас сосредоточиться на том, что действительно важно, отбросив несущественные детали. Однажды Леонард Бернстайн сказал: «Чтобы создать нечто грандиозное, необходимы два условия: план и недостаток времени». Ограничения – это неизбежная часть творческого процесса.

Производство и после него

Студенческий проект вполне можно завершить на текущей стадии – когда закончен дизайн первоначальной идеи, завершено предпроизводство и готов первый «играбельный» прототип. Коммерческие же проекты, как правило, переходят на стадию полноценного производства.

Стадия предпроизводства заканчивается, когда команда узнала об игре и дизайне достаточно, чтобы уверенно составить план всех необходимых этапов с указанием нужного количества человек, ресурсов и графика для каждого типа работ.

После принятия плана команда переходит к стадии *производства*, когда основное внимание уделяется созданию всех частей и объединению их в финальный продукт. Обычно тогда же команда разрастается, чтобы соответствовать плану производства. У данной стадии имеется множество подстадий.

На ранних этапах производства целью является создание «*альфа-сборки*», то есть «играбельного» демо с *полностью реализованными системами* (включая все разнообразные системы и циклы геймплея). Это еще не законченная игра, художественное оформление и уровни могут быть не окончательными, но она должна демонстрировать совместную работу систем и механик. На данной стадии игра все еще очень «сырая». Она может быть нестабильной и вылетать, но команда регулярно играет в нее и отправляет сторонним тестировщикам сведения для получения обратной связи. По мере внедрения различных аспектов геймплея и добавления контента обычно получается несколько альфа-сборок.

Некоторые студии по традиции создают *вертикальные срезы*. Они, как и альфа-сборки, содержат разнообразные геймплейные системы и циклы, но вместе с тем имеют небольшую долю контента, отлаженного почти до финального вида. Например, отдельный уровень с одним типом врагов и единственным особым видом оружия, разработанным настолько, что его можно было бы включить в окончательную, готовую для распространения версию. Вертикальные срезы помогают получить представление о том, как будет выглядеть

законченная игра на примере какой-либо части контента. Но такие срезы полезны лишь в некоторых жанрах игр, таких как игры с уровнями, хотя огромное количество рабочего времени тратится на производство тех или иных доработок. В других жанрах, таких как симуляторы и стратегии, вертикальные срезы не настолько полезны, потому что объединить в них все хорошо отлаженные системы аналогично созданию почти окончательной версии игры.

После завершения работы над альфа-версией наступает следующий этап – закончить работу над оставшимся контентом, интегрировать всю обратную связь от игрового тестирования и приступить к созданию *бета-сборок*, которые еще не являются окончательным вариантом игры, но уже дают достаточно полное представление о ней и достаточно пригодны начала игрового тестирования с привлечением внешних тестировщиков. Бета-версия обычно означает, что игра имеет *полный контент*, геймплей и контент реализованы полностью, хотя их элементы не прошли отладку и не были как следует настроены по стандартам готовой к выпуску версии (возможно, отсутствуют какие-либо художественные материалы, недостаточно сообразителен ИИ и т. д.). На данной стадии идет активное игровое тестирование как со стороны специалистов по контролю качества, так и со стороны других профессионалов (возможно, даже силами преданных геймеров, приглашенных принять участие в тестировании), в ходе которого регулярно выявляется целая куча проблем, которые необходимо решать.

Когда все «баги» исправлены, проблемы решены, а шероховатости сглажены, игра вступает в финальный цикл тестирования на явные ошибки и вылеты. На данной стадии консольным играм тоже приходится проходить платформенную сертификацию. В результате этого процесса создается ряд сборок *«кандидатов на релиз»*, которые тестируются так, словно их предстоит доставить покупателям. Как только кандидат проходит все тесты и получает одобрение на релиз, он приобретает статус *«золотого мастера»* или *«производственного релиза»*.

Но даже после релиза игры работа над ней не заканчивается. Необходимо трудиться для осуществления поддержки программы (исправления «багов», добавления функций и улучшений) или выпуска дополнений, поступающих в продажу. Это особенно верно в отношении так называемых игр-сервисов (GaaS Games as a Service) или игр, в которых предполагается, что игроки будут продолжать участвовать в них и покупать различный дополнительный контент на протяжении длительного времени. В данном случае команде разработчиков потребуется продолжать работу над производством столько, сколько поддерживается игра.

Такое разделение стадий предпроизводства и производства может показаться слишком жестким, но это бывает очень часто, потому что в данном случае главная цель – минимизация рисков. Разработчики отчасти стараются контролировать затраты на производство тем, чтобы устранить поводы для внесения изменений в игру на последних этапах дизайна. Чем позже вносятся непредвиденные изменения, тем дороже они обходятся, особенно в крупных коллективах с разными группами специалистов, где дизайн должен быть согласован с разными отделами (художественного оформления, программистов и т. д.). Но у такой жесткости есть своя цена. Гейм-дизайн часто полагается на эксперименты и исследования, но иногда вдохновение просыпается на поздних стадиях процесса, когда многие части уже реализованы, а разработчики уже проверяют их совместную работу. Такое противоречие между гибкостью и желанием минимизировать затраты невозможно разрешить каким-то одним конкретным способом, поэтому оно продолжает преследовать игры на протяжении всего графика производства.

Но в маленьких командах данный процесс можно значительно упростить. В небольших коллективах обеспечить коммуникацию и взаимодействие гораздо легче. Например, при работе над игрой *Project Highrise* мы очень плавно, без заметной точки старта, переходили от стадии предпроизводства к стадии производства. Мы не разрабатывали подробного плана производства, а вместо этого составили примерный график с гибкими датами относительно

альфа, беты и релиза. Наконец, хотя мы и вносили некоторые изменения на поздней стадии, включать их в игру было легко, а они легко ложились в наш график, потому что команда была небольшой и динамичной.

В конце концов, производственный график – это средство коммуникации, используемое для того, чтобы все работало согласованно и стремились к общим целям. Небольшие коллективы, где официальная коммуникация невелика, а члены команды обладают универсальными навыками, могут позволить себе большую гибкость в составлении графика.

Идеи для студенческого прототипирования

До сих пор мы рассуждали о процессе производства коммерческих продуктов. Теперь обратим внимание на студенческие проекты и выскажем несколько полезных рекомендаций. Студенческие проекты не испытывают такого давления, как коммерческие, от них не требуют успешных показателей, но и здесь есть как свои ограничения, так и свои возможности.

Пожалуйста, воспринимайте все нижесказанное только как советы и предложения. Они высказываются, чтобы облегчить работу для студентов и учащихся, но поскольку некоторые из них могут не соответствовать замыслу каких-либо конкретных программ или курсов, то вы вправе следовать им или изменять по своему усмотрению.

Расширение стадии прототипирования

Длительный и многоэтапный процесс производства характерен для коммерческих студий, но для учебной работы он слишком громоздок. К тому же студенты обычно работают в условиях гораздо более ограниченных сроков. Им выделяют четверть или семестр на создание какой-либо игры, а то и двух-трех, хотя при этом нужно уделять время и другим занятиям. К тому же студенты одновременно обучаются как дизайну, так и разработке, получая большую пользу от экспериментов.

По своему опыту преподавания разработки игр в формате одной или двух четвертей, я предпочитаю сосредотачиваться на стадии предпроизводства. В таком формате типичный график может выглядеть следующим образом.

- Короткая стадия концепции.
- Первая веха: проработать замысел дизайна и составить заявку.
- Длительный период итеративного дизайна и прототипирования.
- Финальная веха: «играбельное» демо, вобравшее в себя весь дизайн.

Похоже, что для студентов с ограниченным графиком оптимально тратить время на эксперименты с дизайном и прототипами, чтобы в конце получилось «играбельное» демо, даже если ему недостает контента. Но в случае с крупными проектами (например, в качестве «дипломных работ», процесс создания которых продолжается год) бывает полезнее придерживаться модели коммерческого процесса.

Разные задачи прототипирования

Хотя в тексте довольно подробно описывается процесс работы над дизайном новой игры от начала до конца, настолько полный процесс не всегда подходит для всех типов проектов. Например, студентам-новичкам бывает комфортнее для начала попытаться изменить дизайн какой-то существующей игры и реализовать в ней свою идею, а не начинать процесс с нуля.

Ниже перечислены предложения для различных типов проектов, которые можно

адаптировать под опыт разных команд и требования курса.

- *Переоформление* или *клон* игры – заимствование механик существующей игры и создание ее новой реализации с новым художественным оформлением. Например, можно сделать клон классической игры вроде *Rac-Man* со всеми ее механиками, но с небольшими изменениями, а также с другими схемами уровней и спрайтами персонажей. Переоформление – подходящая отправная точка для начинающих программистов и дизайнеров, для которых переделка оформления – уже достаточно сложная задача.

- *Вариант* игры – это процесс, заимствующий механики существующего и придающий им новые особенности с сохранением жанра и стиля. Например, вариантом *Rac-Man* может быть игра с какими-либо новыми механиками, их удалением, добавлением механик других игр и т. д. Такой подход представляет собой определенный дизайнерский интерес, но остается в довольно строгих рамках и хорошо подходит для студенческого проекта.

- *Жанровая* игра – процесс, принадлежащий к какому-либо определенному жанру с типичными для него системами и геймплеем, базирующийся на них, но с разными отсылками на конкретные игры с добавлением новых, уникальных для этой игры систем, сеттинга и т. д. По сравнению с созданием варианта, это более сложная дизайнерская задача, потому что изменения охватывают уровень систем и геймплейных циклов.

- *Экспериментальная* игра – попытка создать совершенно новый геймплей, в котором присутствуют элементы игрового опыта или сочетание существующих жанров, выполненное неожиданным образом. Цель данного задания – создать нечто, не имеющее прецедентов. Можно также взять какую-нибудь жанровую игру и изменить ожидания от нее – переделать основные системы и создать на ее основе нечто новое. Такой подход очень интересен, но результаты могут быть совершенно непредсказуемыми, как непредсказуемо и время, которое займет прототипирование.

Это лишь некоторые идеи относительно проектов. Основные ограничения связаны с требованиями курса, с целями проектов и с тем, сколько времени и усилий готовы уделить данным проектам команды.

Составление портфолио разработчика

Студенческие проекты по созданию игр часто полезны как элементы портфолио во время собеседований о приеме на работу в сфере игровой индустрии или в других дисциплинах. Поэтому при работе над проектом студентам стоит задумываться о том, как он послужит демонстрацией их навыков в будущем.

Прежде всего во время работы над студенческим проектом не нужно беспокоиться о коммерческом успехе. Это предоставляет студентам большую свободу для самовыражения, позволяя им делать то, что они не смогли бы осуществить, работая над коммерческим продуктом. Эксперименты с геймплеем или интересными, но неоправданными с коммерческой точки зрения технологиями (например, в области виртуальной реальности) либо стремление продемонстрировать какие-то особенные навыки (например, умение создавать визуальные эффекты) – отличный способ создать пример своей работы для портфолио.

Во-вторых, студенты могут выложить свою игру на общедоступный независимый игровой портал, такой как *Itch.io*. Такие порталы выполняют двойную функцию – служат как хранилищем игр и витриной проектов, так и местом для взаимного общения игроков и разработчиков. Это хороший способ привлечь к себе общественное внимание и получить отзывы и советы со стороны сообщества людей, интересующихся разработкой игр.

Наконец, стоит подумать о том, чтобы продемонстрировать свою игру на студенческих фестивалях. Некоторые из фестивалей, допускающих участие студентов, такие как *IndieCade*

или *Independent Games Festival* , обладают огромным престижем, но и конкуренция на них очень высока. Студентам следует тщательно взвесить все варианты и выбрать фестиваль, соответствующий их навыкам и предполагаемому качеству игры. Кроме того, различные фестивали придерживаются разных стилей и могут отдавать предпочтение каким-то конкретным жанрам или отдельному типу геймплея, что также следует принимать во внимание.

Краткий итог

В данной главе мы рассмотрели процесс, в ходе которого команды разработчиков на основе концепции и дизайна создают готовые для предоставления игрокам игры.

- Сначала мы перечислили типичные для игровых студий *стадии разработки* и рассмотрели роль концепции, за формированием которой следует стадия предпроизводства и прототипирования, переходящая в стадии производства, релиза и пострелизных работ.

- Мы в подробностях разобрали *стадию концепции* и ее цели. Хорошая концепция должна помогать команде понимать и осознавать дизайн игры, а также давать представление о ее рыночной ценности. Такую концепцию можно выразить в коммерческом предложении или заявке, то есть в документе, с помощью которого идеи доносят до других пользователей как внутри компании, так и вовне, например, описывая игру инвесторам.

- Ориентируясь на небольшие команды мы описали метод создания краткого списка вопросов, на которые следует ответить перед тем, как объявить стадию концепта законченной. По утверждению Келли (Kelly, 2014), на ранней стадии эти вопросы могут стать заменой подробного дизайнерского документа.

- После стадии концепции мы обратили внимание на *стадию предпроизводства* (а именно прототипирования идей о дизайне игры) и представили концепцию отдельных и *накопительных прототипов* . Мы исследовали итеративный процесс, в ходе которого дизайнерские идеи превращаются в прототипы, проходящие игровое тестирование и используемые для улучшения дизайна. Более подробно был рассмотрен процесс *игрового тестирования* , представляющий собой цикличные переходы между дизайном, реализацией, тестированием и получением более качественного результата. Наконец, мы описали переход от стадии прототипирования к стадии предпроизводства и поговорили о ее роли и пострелизной стадии в разработке коммерческих продуктов.

- В финале мы предложили некоторые идеи проектов и решили, что для них будет лучше пропустить стадию производства, чтобы лучше сосредоточиться на тех аспектах разработки, которые будут особенно полезны для студентов.

Дополнительная литература

Теме стадий производства посвящено огромное количество источников. Подробное описание планирования, прототипирования и игрового тестирования приводится в следующих книгах:

- «Продвинутый гейм-дизайн» (*Advanced Game Design* , Sellers, 2017), главы 11–12
- «Мастерская гейм-дизайна» (*Game Design Workshop* , Fullerton, 2008), главы 6–9
- «Гейм-дизайн» (*The Art of Game Design* , Schell, 2008), главы 23–25

Что касается темы игрового бизнеса в целом, то довольно доступным введением в нее можно назвать книгу «Руководство по геймдев-бизнесу» (*GameDev Business Handbook* , Futter, 2017). Книга «Пирамида гейм-дизайна» (*The Pyramid of Game Design*, Lovell, 2018) более подробно описывает модели free-to-play. А поскольку ситуация в данной сфере очень быстро меняется, заинтересованные читатели могут найти много всего интересного на таких

тематических сайтах, как Gamasutra и GamesIndustry.Biz.

Конкретные темы заявок (коммерческих предложений) и внешней коммуникации затрагивают некоторые из указанных ниже ресурсов.

- «Руководство по геймдев-бизнесу» (*GameDev Business Handbook*, Futter, 2017), глава 6

- «Мастерская гейм-дизайна» (*Game Design Workshop*, Fullerton, 2008), главы 13–14

- «Гейм-дизайн» (*The Art of Game Design*, Schell, 2008), главы 27–29

Наконец, для читателей, интересующихся историей итеративного дизайна и разработки, особенно в контексте программного обеспечения, отличными источниками послужат «Загадочный человек-месяц» (*The Mythical Man-Month*, Brooks, 1975) и «Дизайн дизайна» (*The Design of Design*, Brooks, 2010).

Упражнения для выполнения в группах

Примечание : данные упражнения предназначены для студентов или команд, приступающих к работе над новым проектом.

Г7.1. Заявка проекта

Составьте одностраничный документ-презентацию вашего проекта. Он должен включать в себя следующие компоненты.

- Краткая «речь для лифта» в одном или нескольких очень коротких и емких предложениях.

- Краткое описание геймплея в несколько абзацев.

- Описание основных аспектов дизайна.

- Ссылки на любые игры, послужившие источником вдохновения для отдельных механик или систем.

Г7.2. Эвристические принципы гейм-дизайна

Ответьте на все семь вопросов Келли относительно вашей игры. Помогли ли ответы на них каким-то образом уточнить ваш дизайн? Если да, то как?

Г7.3 Итеративная разработка

Представьте свой проект, проходящий через спираль итеративной разработки. Как вы продумываете дизайн определенных аспектов, реализуете их, тестируете, анализируете результаты, на основании которых пересматриваете дизайн и повторяете цикл. Какие элементы вы хотите реализовать первыми и почему? Какими будут последующие элементы, основанные на первых?

Заключение

На протяжении всего текста мы рассматривали множество самых разных элементов дизайна, механик, систем, геймплейных циклов и разновидностей игрового опыта, а в последней главе поговорили о том, как из всего этого создается игра.

Куда же нам двигаться дальше?

Теперь, вооружившись всеми этими сведениями, мы можем сделать следующий шаг – попробовать усвоенное на практике. Самый лучший способ научиться гейм-дизайну – постараться реализовать свои собственные дизайнерские идеи и понаблюдать за тем, как они работают в игре с участием настоящих игроков.

Дальше вас ожидает самый волнующий и захватывающий этап – создание своей

собственной игры.
Удачи!

ИСТОЧНИКИ

- Abt, Clark C, et al. 1965. *Counter-Insurgency Game Design Feasibility and Evaluation Study* . Washington, DC: Advanced Research Projects Agency. https://archive.org/details/DTIC_AD0475846.
- Adams, Ernest, and Joris Dormans. 2012. *Game Mechanics: Advanced Game Design* . Berkeley, CA: New Riders.
- Adams, Harold W. 1970. "The Design Dilemma: Games for Whom?" *Proceedings of the 9th Symposium. National Gaming Council* . 136–143. https://archive.org/details/ERIC_ED049109.
- Anthropy, Anna, and Naomi Clark. 2014. *A Game Design Vocabulary: Exploring the Foundational Principles Behind Good Game Design* . Upper Saddle River, NJ: Addison-Wesley.
- Avalon Hill, Inc. 1971. "Avalon Hill Philosophy." *The General Magazine* , July – Aug: 2–3. <https://archive.org/details/GeneralMagazineVol8i2>.
- Bartle, Richard. 1996. "Hearts, Clubs, Diamonds, Spades: Players Who Suit MUDs." Technical Report, Muse Ltd., Colchester. Доступно по состоянию на 27 октября 2018 года. <http://mud.co.uk/richard/hcds.htm>.
- Barwood, Hal, and Noah Falstein. n. d. *The 400 Project* . Accessed September 17, 2018. <http://www.finitearts.com/Pages/400page.html>.
- Bjork, Staffan, and Jussi Holopainen. 2004. *Patterns in Game Design* . Needham, MA: Charles River Media.
- Boehm, Barry. 2000. "Spiral Development: Experience, Principles, and Refinements." Report CMU/SEI-2000-SR-008, Software Engineering Institute, Carnegie Mellon, Pittsburgh. Accessed November 14, 2018. <http://www.dtic.mil/dtic/tr/fulltext/u2/a382590.pdf>.
- Brathwaite, Brenda, and Ian Schreiber. 2009. *Challenges for Game Designers* . Boston, MA: Cengage Learning.
- Brooks, Frederick P., Jr. 1975. *The Mythical Man-Month: Essays on Software Engineering* . Boston, MA: Addison-Wesley.
- Brooks, Frederick P., Jr. 2010. *The Design of Design: Essays from a Computer Scientist* . Upper Saddle River, NJ: Addison-Wesley.
- Caillois, Roger. 1962. "The Definition of Play: The Classification of Games." В книге *The Game Design Reader* (2006), by Katie Salen and Eric Zimmerman. Cambridge, MA: MIT Press.
- Church, Doug. 1999. "Formal Abstract Design Tools." Блог. *Gamasutra* . July 16, 1999. http://www.gamasutra.com/view/feature/131764/formal_abstract_design_tools.php.
- Cook, Daniel. 2012. "Loops and Arcs." April 29, 2012. <http://www.lostgarden.com/2012/04/loops-and-arcs.html>.
- CORE Team. 2017. *The Economy* . Oxford: Oxford University Press.
- Costikyan, Greg. 2002. "I Have No Words & I Must Design: Toward a Critical Vocabulary for Games." В публикации *Proceedings of Computer Games and Digital Cultures Conference* , edited by Frans Mäyrä. Tampere, FL: Tampere University Press. <http://www.costik.com/nowords2002.pdf>.
- Costikyan, Greg. 2007. "Games, Storytelling, and Breaking the String." В издании *Second Person* , edited by Pat Harrigan and Noah Wardrip-Fruin. Cambridge, MA.
- Costikyan, Greg. 2013. *Uncertainty in Games* . Cambridge, MA: MIT Press.
- Crawford, Chris. 2003. *Chris Crawford on Game Design* . Indianapolis, IN: New Riders.
- Crawford, Chris. 2005. *Chris Crawford on Interactive Storytelling* . Indianapolis, IN: New Riders.
- Csikszentmihalyi, Mihaly. 1990. *Flow: The Psychology of Optimal Experience* . New York: Harper & Row.
- Csikszentmihalyi, Mihaly, Sami Abuhamedh, and Jeanne Nakamura. 2005. "Flow." В

издании *Handbook of Competence and Motivation* , edited by A. J. Elliot and C. S. Dweck, 598–608. New York: Guilford Press.

Davies, Mark. 2009. “Examining Game Pace: How Single-Player Levels Tick.” Блог. *Gamasutra* . May 12, 2009. https://www.gamasutra.com/view/feature/132415/examining_game_pace_how_.php.

Deen, Menno. 2015. “G.A.M.E., Games Autonomy Motivation & Education: How Autonomy-supportive Game Design May Improve Motivation to Learn.” Ph.D. diss., Technische Universiteit Eindhoven, Eindhoven.

DiabloWiki. 2018. “Experience Level Chart.” Accessed October 23, 2018. https://www.diablowiki.net/Experience_level_chart.

Dunnigan, James F. 2000. *Wargames Handbook, Third Edition: How to Play and Design Commercial and Professional Wargames* . Lincoln, NE: Writers Club Press.

Elias, George Skaff, Richard Garfield, and K. Robert Gutschera. 2012. *Characteristics of Games* . Cambridge, MA: MIT Press.

Fenlon, Wes. 2016. “How Cats Get Drunk in Dwarf Fortress, and Why Its Creators Haven’t Figured Out Time Travel (Yet).” *PC Gamer* . March 31, 2016. <http://www.pcgamer.com/how-cats-get-drunk-in-dwarf-fortress-and-why-its-creators-havent-figured-out-time-travel-yet/>.

Forrester, Jay Wright. 1961. *Industrial Dynamics* . Cambridge, MA: MIT Press.

Fullerton, Tracy. 2008. *Game Design Workshop: A Playcentric Approach to Creating Innovative Games, 2nd Edition* . Burlington, MA: Morgan Kaufmann.

Futter, Michael. 2017. *The GameDev Business Handbook: How to Build the Business You’ll Build Games With* . London, UK: Bithell Games.

Garriott, Richard. 2017. “War Stories. Ultima Online: The Virtual Ecology.” Видео, 7:27. *Ars Technica* . December 21, 2017. <http://video.arstechnica.com/watch/war-stories-ultima-online-the-virtual-ecology>.

Greenlaw, Paul S., Lowell W. Herron, and Richard H. Rawdon. 1962. *Business Simulation in Industrial and University Education* . Englewood Cliffs, NJ: Prentice Hall. <https://archive.org/details/businesssimulati00gree>.

Harrigan, Pat, and Noah Wardrip-Fruin. 2007. *Second Person: Role-playing and Story in Games and Playable Media* . Cambridge, MA: MIT Press.

Harrigan, Pat, and Noah Wardrip-Fruin. 2009. *Third Person: Authoring and Exploring Vast Narratives* . Cambridge, MA: MIT Press.

Hocking, Clint. 2007. “Ludonarrative Dissonance in *BioShock* .” Блог. *Click Nothing* . Accessed October 7, 2007. http://clicknothing.typepad.com/click_nothing/2007/10/ludonarrative-d.html.

Hodent, Celia. 2017. *The Gamer’s Brain: How Neuroscience Can Impact Video Game Design* . Boca Raton, FL: CRC Press.

Hunicke, Robin, Marc LeBlanc, and Robert Zubek. 2004. “MDA: A Formal Approach to Game Design and Game Research.” Edited by Dan Fu, Stottler Henke, and Jeff Orkin. *Challenges in Game Artificial Intelligence: Papers from the 2004 Workshop* . Menlo Park, CA: AAAI Press.

John, Oliver P., and Sanjay Srivastava. 1999. “The Big Five Trait Taxonomy: History, Measurement, and Theoretical Perspectives.” In *Handbook of Personality: Theory and Research* , edited by Lawrence A. Pervin and Oliver P. John, 102–138. New York: Guilford Press.

John, Oliver P., and Sanjay Srivastava. n. d. “The Big Five Trait Taxonomy: History, Measurement, and Theoretical Perspectives.” Preprint. Доступ по состоянию на 15 сентября 2018 года. <https://pages.uoregon.edu/sanjay/pubs/bigfive.pdf>.

Juul, Jesper. 2010. “The Video Games of Video Games: Prejudices against Social Games verbatim copies of Prejudices against Video Games.” Блог. *The Ludologist* . September 1. <https://www.jesperjuul.net/ludologist/2010/08/31/the-video-games-of-video-games/>.

Kelly, Tadhg. 2014. “Raw Game Design: You Have a Game Idea, What Comes Next?” *Game Developers Conference Next 2014* . Los Angeles, CA.

- Koster, Raph. 2004. *A Theory of Fun in Game Design* . Scottsdale, AZ: Paraglyph Press.
- Koster, Raph. 2005. "A Grammar of Gameplay." *Game Developers Conference 2005* . San Francisco, CA. <https://www.raphkoster.com/gaming/atof/grammarofgameplay.pdf>.
- Koster, Raph. 2012. "A Theory of Fun: 10 Years Later." *Game Developers Conference Online 2012* . Austin, TX. https://www.raphkoster.com/gaming/gdco12/Koster_Raph_Theory_Fun_10.pdf.
- Lazzaro, Nicole. 2004. "Why We Play Games: Four Keys to More Emotion in Player Experiences." *Game Developers Conference 2004* . San Francisco, CA.
- Lehdonvirta, Vili, and Edward Castronova. 2014. *Virtual Economies: Design and Analysis* . Cambridge, MA: MIT Press.
- Llamasoft Ltd. 1985. "Batalyx." https://archive.org/details/Batalyx_1985_Llamasoft.
- Lovell, Nicholas. 2018. *The Pyramid of Game Design: Designing, Producing and Launching Service Games* . Boca Raton, FL: CRC Press.
- Madigan, Jamie. 2015. *Getting Gamers: The Psychology of Video Games and Their Impact on the People Who Play Them* . Lanham, MD: Rowman & Littlefield.
- Mateas, Michael, and Andrew Stern. 2003. "Façade: An Experiment in Building a FullyRealized Interactive Drama." *Game Developers Conference 2003* . San Francisco, CA.
- Mawhorter, Peter, Michael Mateas, Noah Wardrip-Fruin, and Arnav Jhala. 2014. "Towards a Theory of Choice Poetics." *Foundations of Digital Games Conference* .
- McEntee, Chris. 2012. "Rayman Origins." *Game Developer Magazine*. no. 10/2012. 26–31.
- McKee, Robert. 1997. *Story: Style, Structure, Substance, and the Principles of Screenwriting* . New York: Harper Collins.
- Meier, Sid. 2012. "Interesting Decisions." *Game Developers Conference 2012* . San Francisco, CA.
- Norman, Donald A. 1988. *The Design of Everyday Things* . New York: Basic Books.
- Pecorella, Anthony. 2016. "Quest for Progress: The Math and Design of Idle Games." *Game Developers Conference Europe 2016* . Cologne, Germany.
- Reynolds, Craig. n. d. "Boids: Background and Update." Доступ по состоянию на 7 сентября 2018 года. <https://www.red3d.com/cwr/boids/>.
- Ryan, Richard M., and Edward L. Deci. 2000. "Self Determination Theory and the Facilitation of Intrinsic Motivation, Social Development, and Well-Being." *American Psychologist* 55 (1): 68–78.
- Ryan, Richard M., C. Scott Rigby, and Andrew Przybylski. 2006. "The Motivational Pull of Video Games: A Self-Determination Theory Approach." *Motivation and Emotion* 30 (4): 344–360.
- Salen, Katie, and Eric Zimmerman. 2004. *Rules of Play* . Cambridge, MA: MIT Press.
- Schell, Jesse. 2008. *The Art of Game Design: A Book of Lenses* . Boca Raton, FL: CRC Press.
- Schüll, Natasha Dow. 2012. *Addiction by Design: Machine Gambling in Las Vegas* . Princeton, NJ: Princeton University Press.
- Selinker, Mike, ed. 2011. *The Kobold Guide to Board Game Design* . Kirkland, WA: Open Design.
- Sellers, Michael. 2017. *Advanced Game Design: A Systems Approach* . Indianapolis, IN: Addison-Wesley.
- Serviss, Ben. 2013. "The Discomfort Zone: The Hidden Potential of Valve's AI Director." Блог. *Gamasutra* . February 7, 2013. https://www.gamasutra.com/blogs/BenServiss/20130207/186193/The_Discomfort_Zone_The_Hidden_Potential_of_Valves_AI_Director.php.
- Sicart, Miguel. 2008. "Defining Game Mechanics." *Game Studies* 8 (2). <http://gamestudies.org/0802/articles/sicart>.
- Skolnick, Evan. 2014. *Video Game Storytelling: What Every Developer Needs to Know About Narrative Techniques* . Berkeley, CA: Watson-Guptill.
- Solarski, Chris. 2017a. *Interactive Stories and Video Game Art: A Storytelling Framework for*

Game Design . Boca Raton, FL: CRC Press.

Solarski, Chris. 2017b. "The Unreliable Gamemaster: Player Motivation in StoryDriven Games." Блог. *Gamasutra* . September 9, 2017. https://www.gamasutra.com/blogs/KrzysztofSolarski/20170209/291164/The_Unreliable_Gamemaster_Player_Motivation_in_StoryDriven_Games.php.

Spolsky, Joel. 2000. "Things You Should Never Do, Part I." Блог. Joel on Software. April 6, 2000. <https://www.joelonsoftware.com/2000/04/06/things-you-should-never-do-part-i/>.

Swink, Steve. 2009. *Game Feel: A Game Designer's Guide to Virtual Sensation* . Boston: Morgan Kaufmann.

Tozour, Paul. 2013. "Postmortem: Intelligence Engine Design Systems' City Conquest." Блог. *Gamasutra* . February 6, 2013. https://www.gamasutra.com/view/feature/186088/postmortem_intelligence_engine_.php.

Upton, Brian. 2015. *The Aesthetic of Play* . Cambridge, MA: MIT Press.

Walker, Stephen F. 1975. *Learning and Reinforcement* . London: Methuen & Go Ltd.

Ward, Trent. 2003. "Ultima Online Preview." *Gamespot* . December 2, 2003. <https://www.gamespot.com/articles/ultima-online-preview/1100-2559974/>.

Wardrip-Fruin, Noah, and Pat Harrigan, ed. 2004. *First Person: New Media as Story, Performance, and Game* . Cambridge, MA: MIT Press.

Wardrip-Fruin, Noah. Готовится к выходу. *How Pac-Man Eats* . Cambridge, MA: MIT Press.

Westfall, Anne, Jon Freeman, and Paul Rieche III. 1983. *Archon Manual* . <https://archive.org/details/ArchonEAManual>.

Wiener, Norbert. 1961. *Cybernetics: or Control and Communication in the Animal and the Machine, 2nd Edition* . Cambridge, MA: MIT Press.

Wittgenstein, Ludwig. 1959. *Philosophical Investigations* . Перевод G. E. M. Anscombe. New York: Macmillan.

Wolf, Maryanne. 2018. "What Does Immersing Yourself in a Book Do to Your Brain?" *Literary Hub* . August 8, 2018. <https://lithub.com/what-does-immersing-yourself-in-a-book-do-to-your-brain/>.

Wright, Will. 2003. *Lessons from Game Design* . Видео на YouTube, 1:42:10. Записано в Музее компьютерной истории. Mountain View, CA, November 20. <https://www.youtube.com/watch?v=CdgQyq3hEPo>.

Yee, Nick. 2016. "Gamer motivation profile: model and findings." *Game Developers Conference 2016* . San Francisco, CA. <https://quanticfoundry.com/wp-content/uploads/2016/03/Gamer-Motivation-Profile-GDC-2016-Slides.pdf>.