OD POMYSŁU
DO GOTOWEJ GRY

PROJEKTOWANIE GIER przy użyciu środowiska UNITY I JĘZYKA C#

Wydanie II

Helion

Jeremy Gibson **BOND**

Tytuł oryginału: Introduction to Game Design, Prototyping, and Development: From Concept to Playable Game with Unity and C# (2nd Edition)

Tłumaczenie: Jacek Janusz

ISBN: 978-83-283-4252-1

Authorized translation from the English language edition, entitled:
INTRODUCTION TO GAME DESIGN, PROTOTYPING, AND DEVELOPMENT:
FROM CONCEPT TO PLAYABLE GAME WITH UNITY AND C#, Second Edition;
ISBN 0134659864; by Jeremy Gibson Bond; published by Pearson Education, Inc, publishing as Addison-Wesley Professional. Copyright © 2018 by Pearson Education, Inc.

Helion SA
ul. Kościuszki 1c, 44-100 Gliwice
tel. 32 231 22 19, 32 230 98 63
e-mail: *helion@helion.pl*
WWW: *http://helion.pl* (księgarnia internetowa, katalog książek)

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
http://helion.pl/user/opinie/progie
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Pliki z przykładami omawianymi w książce można znaleźć pod adresem:
ftp://ftp.helion.pl/przyklady/progie.zip

Printed in Poland.

SPIS TREŚCI

Część II Prototypowanie cyfrowe

ROZDZIAŁ 5.

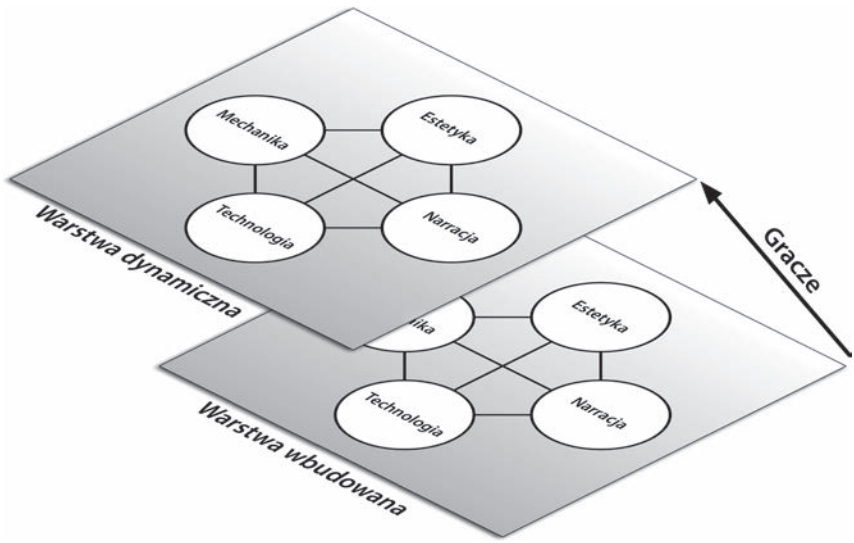
WARSTWA DYNAMICZNA

Gdy gracze rozpoczynają grę, zmienia ona swoje położenie w tetradzie warstwowej z warstwy wbudowanej na dynamiczną. Pojawiają się w niej: granie, strategia i sensowne wybory graczy.

W tym rozdziale omówiono warstwę dynamiczną, różne cechy emergencji oraz sposoby, dzięki którym projektanci mogą przewidywać postać grania dynamicznego wyłaniającego się z wbudowanych decyzji projektowych.

Rola gracza

Pewien znajomy projektant powiedział mi kiedyś, że gra nie jest grą, dopóki nie będzie przez kogoś grana. Chociaż może to brzmieć początkowo jak powtórka powiedzenia „jeśli drzewo upadło w lesie i nie było nikogo, kto mógłby to słyszeć, to czy wydało ono dźwięk?", ta definicja jest w rzeczywistości dużo ważniejsza w przypadku mediów interaktywnych niż jakichkolwiek innych. Film może nadal istnieć, a sztuka być pokazywana na scenie, nawet jeśli nie ma nikogo, kto mógłby je oglądać[1]. Telewizja można być transmitowana poprzez fale radiowe i nadal nią będzie, nawet jeżeli nikt nie będzie miał włączonego odbiornika. Gry jednak po prostu nie istnieją bez graczy, ponieważ przez ich działania przekształcają się one ze zbioru elementów wbudowanych w doświadczenie dynamiczne (patrz rysunek 5.1).



Rysunek 5.1. Gracze przenoszą grę z warstwy wbudowanej do warstwy dynamicznej

Jak wszędzie, także tutaj oczywiście zdarzają się skrajne przypadki. *Core War* to gra hakerska, w której gracze próbują pisać wirusy komputerowe. Są one następnie rozpowszechniane i próbują przejąć procesor komputera, pozbywając się wirusów stworzonych przez innych graczy. Gracze przesyłają swoje wirusy i czekają, aż będą one walczyć ze sobą o pamięć i przetrwanie. W corocznym turnieju *RoboCup* różne zespoły robotów rywalizują ze sobą w piłce nożnej bez ingerowania ze strony programistów podczas rozgrywki. W klasycznej grze karcianej o nazwie *Wojna* gracze nie podejmują żadnej decyzji poza wyborem, jaką z dwóch talii użyje się na początku gry, która następnie toczy się całkowicie w oparciu o przypadkowość początkowego tasowania.

[1] Niektóre filmy, takie jak *The Rocky Horror Picture Show*, mają społeczność fanów i wysoki poziom kultowości w dużym stopniu dzięki przedstawieniom, w których bierze udział publiczność. Reakcje publiczności podczas oglądania tych filmów rzeczywiście zmieniają wrażenia odbioru u innych widzów. Jednak publiczność nie ma wpływu na sam film. Dynamika w grach polega na zdolności medium do reagowania na zachowanie gracza.

Chociaż w każdej z powyżej przedstawionych trzech sytuacji gracz nie bierze udziału poprzez dokonywanie wyborów podczas faktycznego grania w grę, na jej przebieg wciąż wpływają jego decyzje podjęte przed oficjalnym rozpoczęciem rozgrywki, a gracze z pewnością są zainteresowani i oczekują na wynik gry. Ponadto wszystkie te przypadki wymagają od graczy skonfigurowania gry i dokonania wyborów, które zdecydują o jej wyniku.

Mimo że gracze mają ogromny wpływ na grę i rozgrywkę (z oddziaływaniem na elementy tetrady włącznie), znajdują się poza tetradą i są „silnikiem" sprawiającym, że gra działa. Gracze powodują, że gry urzeczywistniają się i pozwalają im stać się doświadczeniem, które twórcy gier zakodowali w warstwie wbudowanej. Będąc projektantami, polegamy na graczach, którzy pomagają nam, by gra stała się tym, co dla niej zaplanowaliśmy. Kilka aspektów rozgrywki w warstwie dynamicznej znajduje się całkowicie poza kontrolą projektantów, w tym także to, czy gracz faktycznie stara się przestrzegać zasad, czy troszczy się o wygraną, jakie jest fizyczne środowisko, w którym jest prowadzona gra, jak przedstawiają się emocjonalne stany graczy itd. Ponieważ gracze są tak ważni, projektanci muszą ich traktować z szacunkiem i dbać o to, aby wbudowane elementy gry — zwłaszcza reguły — były dla nich zrozumiałe. Spowoduje to, że gracze będą mogli je przekształcić w doświadczenia, które zostały dla nich zaplanowane.

Emergencja

Najważniejszą koncepcją omawianą w tym rozdziale jest **emergencja**, której fundamentem jest to, że nawet bardzo proste reguły mogą powodować złożone zachowania dynamiczne. Zastanów się nad grą *Bartok*, w którą grałeś i z którą eksperymentowałeś w rozdziale 1. zatytułowanym „Myśląc jak projektant". Chociaż gra *Bartok* miała bardzo niewiele zasad, wyłoniło się z niej skomplikowane granie. Kiedy zacząłeś zmieniać zasady i dodawać własne, zobaczyłeś także, że nawet proste, pozornie nieszkodliwe modyfikacje reguł mogą prowadzić do potencjalnie dużych zmian zarówno w odbiorze, jak i graniu w grę.

Warstwa dynamiczna tetrady warstwowej zawiera wspólne elementy dotyczące gracza i gry dla wszystkich czterech jej składników: mechaniki, estetyki, dramatyzmu i technologii.

Nieoczekiwana emergencja mechaniczna

Mój przyjaciel Scott Rogers, autor dwóch wspaniałych książek o projektowaniu gier[2], powiedział mi kiedyś, że nie wierzy w emergencję. Po krótkiej dyskusji doszliśmy do wniosku, że co prawda wierzył w emergencję, ale nie w to, iż projektanci gier mają prawo jej używać jako pretekstu do nieodpowiedzialnego projektowania. Zgadzamy się, że będąc twórcą systemów w grze, jesteś odpowiedzialny za granie, które wyłania się z nich. Oczywiście bardzo trudno jest przewidzieć, jakie możliwości wynikną z reguł, które zdefiniujesz, dlatego tak ważne jest testowanie. Rozwijaj swoje gry, grając często już we wczesnej fazie ich rozwoju. Zwracaj szczególną uwagę na to, aby dostrzec niestandardowe zdarzenia, które wystąpią tylko podczas jednej gry testowej. Po tym, jak gra zostanie wydana, sama liczba osób grających spowoduje, że te nietypowe przypadki pojawią się znacznie częściej, niż można by się tego było spodziewać.

[2] Scott Rogers, *Level up!: The Guide to Great Video Game Design*, Wiley, Chichester 2010, oraz Scott Rogers, *Dotknij i przeciągnij. Projektowanie gier na ekrany dotykowe*, Helion, Gliwice 2013.

Oczywiście zdarza się to wszystkim projektantom: spójrz choćby na niektóre z kart, które zostały uznane za nielegalne w grze *Magic: The Gathering*. Jak mówi jednak Scott, ważne jest, by projektanci wzięli odpowiedzialność za te problemy i ich rozwiązanie.

Mechanika dynamiczna

Mechanika dynamiczna to dynamiczna warstwa elementów, które odróżniają gry i media interaktywne od innych mediów. Są to składowe, które sprawiają, że gry są grami. Mechanika dynamiczna obejmuje procedury, sensowne granie, strategię, reguły lokalne, zamiary gracza i wynik. Podobnie jak ma to miejsce w przypadku mechaniki wbudowanej, wiele z nich jest rozwinięciem elementów opisanych w książce Tracy Fullerton *Game Design Workshop*[3].
Na mechanikę dynamiczną, którą omówimy, składają się:

- **procedury**: działania, które podejmują gracze,
- **sensowne granie**: nadawanie wagi decyzjom graczy,
- **strategia**: plany opracowywane przez graczy,
- **reguły lokalne**: proste modyfikacje gry wprowadzane przez graczy,
- **zamiary gracza**: motywacje i cele graczy,
- **wynik**: wynik (lub wyniki) grania w grę.

Procedury

Mechanika w warstwie wbudowanej zawiera **reguły**, czyli przekazywane graczom przez projektanta instrukcje, w jaki sposób należy grać w grę. **Procedury** są dynamicznymi działaniami podejmowanymi przez graczy w odpowiedzi na te reguły. Jest to jednoznaczne z tym, że procedury wynikają z reguł. Rozważ poniższą opcjonalną regułę z gry *Bartok*, przedstawioną w rozdziale 1.:

- **Reguła 3.** Gracz musi podać informację „Ostatnia karta!", gdy ma tylko jedną kartę. Jeśli ktoś inny ogłosi wcześniej tę samą wiadomość, gracz musi dobrać dwie karty (co spowoduje, że jego całkowita liczba kart wzrośnie do trzech).

Ta zasada w sposób bezpośredni instruuje aktywnego gracza, aby postępował zgodnie z procedurą informowania, gdy pozostanie mu tylko jedna karta. Jednak ta zasada pośrednio implikuje kolejną procedurę: to, że inni gracze powinni obserwować aktywnego gracza, aby mogli go uprzedzić, jeśli zapomni przekazać informację. Przed tą regułą nie było żadnego prawdziwego powodu, dla którego gracz zwracałby uwagę na grę podczas trwania kolejki innej osoby. Ta prosta zmiana zasad zmieniła procedury grania w grę zarówno dla aktywnych, jak i nieaktywnych graczy.

[3] Tracy Fullerton, Christopher Swain i Steven Hoffman, *Game Design Workshop: A Playcentric Approach to Creating Innovative Games*, Morgan Kaufmann Publishers, Burlington 2008, rozdziały 3. i 5.

Sensowne granie

W książce *Rules of Play* Katie Salen i Eric Zimmerman definiują **sensowne granie** jako takie, które jest *dostrzegalne* przez gracza i *zintegrowane* w większą grę[4].

- **Dostrzegalne**: działanie jest dostrzegalne, jeśli gracz może stwierdzić, że akcja została podjęta. Na przykład naciśnięcie przycisku wzywającego windę jest dostrzegalne, ponieważ przycisk zaczyna świecić. Jeśli kiedykolwiek próbowałeś przywołać windę, gdy żaróweczka wewnątrz przycisku była uszkodzona, wiesz, jak frustrująca może być próba działania, gdy jednocześnie nie możesz stwierdzić, czy gra je odpowiednio zinterpretowała.
- **Zintegrowane**: działanie jest zintegrowane, jeśli gracz może powiedzieć, że jest ono związane z wynikiem gry. Na przykład gdy naciśniesz przycisk przywołania windy, czynność ta zostanie zintegrowana, ponieważ wiesz, że spowoduje ona zatrzymanie się windy na Twoim piętrze. W grze *Super Mario Bros.* decyzja o tym, czy należy zniszczyć pojedynczego wroga, czy po prostu go unikać, zwykle nie ma większego znaczenia, ponieważ to indywidualne działanie nie jest zintegrowane z ogólnym wynikiem gry. W tej grze nigdy nie dowiesz się o liczbie pokonanych wrogów. Wymaga ona jedynie ukończenia każdego poziomu, zanim skończy się czas, a ostatecznie zakończenia gry w ramach przysługujących wskrzeszeń. W serii gier *Kirby* autorstwa HAL Laboratories postać gracza Kirby zyskuje specjalne zdolności podczas pokonywania przeciwników, więc decyzja, którego z wrogów należy pokonać, jest bezpośrednio połączona z nabywaniem umiejętności, dzięki czemu jest bardziej sensowna.

Jeśli działania gracza w grze nie mają znaczenia, może on szybko stracić zainteresowanie nią. Koncepcja sensownego grania autorstwa Salen i Zimmermana przypomina projektantom, aby stale zastanawiali się nad sposobem myślenia gracza i analizowali, czy z jego punktu widzenia interakcje w grach są przezroczyste lub nieprzejrzyste.

Strategia

Kiedy gra pozwala na sensowne działania, gracze zwykle tworzą strategie, które pomagają im ją wygrać. **Strategia** to skalkulowany zestaw działań, które pomagają graczowi osiągnąć cel. Jednak ten cel może być dowolnym wyborem gracza i niekoniecznie musi być jednoznaczny z wygraniem gry. Na przykład podczas grania z małym dzieckiem lub z kimś o niższym poziomie umiejętności celem gracza może być upewnienie się, że druga osoba lubi grać w grę i uczy się czegoś, czasami kosztem gracza wygrywającego.

Strategia optymalna

Gdy gra jest bardzo prosta i zawiera niewiele możliwych akcji, jest prawdopodobne, że gracz opracuje **optymalną strategię** gry. Jeśli obaj gracze grają racjonalnie, mając na celu zwycięstwo, optymalna strategia jest możliwą do zrealizowania strategią z najwyższym prawdopodobieństwem wygranej.

[4] Katie Salen i Eric Zimmerman, *Rules of Play: Game Design Fundamentals*, MIT Press, Cambridge 2003.

Większość gier jest jednak zbyt skomplikowana, aby naprawdę posiadać optymalną strategię, lecz niektóre, takie jak *Kółko i krzyżyk* są na tyle proste, że ją mają. W rzeczywistości gra *Kółko i krzyżyk* jest tak prosta, że nawet kurczaki zostały (prawdopodobnie) przeszkolone, w jaki sposób w nią grać, i wymuszają remis lub wygraną prawie za każdym razem[5].

Strategia optymalna jest częściej niewyraźnym pomysłem na coś, co prawdopodobnie poprawi szanse gracza na wygraną. Na przykład w grze planszowej *W górę rzeki* autorstwa Manfreda Ludwiga gracze próbują przetransportować trzy łodzie w górę rzeki, aby zacumować przy górnej krawędzi planszy. Przybycie do portu jest warte 12 punktów za pierwszą łódź, następnie 11 punktów za drugą, a wreszcie jedynie 1 punkt za dwunastą łódź. W każdej rundzie (czyli za każdym razem, gdy wszyscy gracze wykonali jedną turę) rzeka przesuwa się o 1 pole w tył, a każda łódź spadająca na końcu rzeki (z wodospadu) zostaje utracona. W każdej turze gracz rzuca pojedynczą sześcioboczną kostką i wybiera, którą łódź należy przemieścić. Ponieważ średnia wartość rzutów kostką sześcioboczną wynosi 3,5, a gracz musi wybrać jedną spośród swoich trzech łodzi, aby przemieścić się w każdej turze, każda łódź porusza się co trzy kolejki średnio o 3,5 pola. Plansza porusza się jednak do tyłu o 3 pola co trzy rundy, tak więc każda łódź uzyskuje średnią prędkość przemieszczenia do przodu równą 0,5 pola na trzy kolejki (lub 0,1666, czyli $^1/_6$ pola w pojedynczej rundzie)[6].

W tej grze optymalna strategia polega na tym, aby gracz nigdy nie przemieszczał jednej ze swoich łodzi i po prostu pozwolił jej spaść z wodospadu. Następnie każda łódź będzie poruszać się średnio o 3,5 pola co dwie rundy zamiast trzech. Gdy plansza cofnie się o 2 pola w dwóch rundach, każda z łodzi będzie miała przeciętną prędkość 1,5 pola co dwie tury (lub 0,75 pola w pojedynczej turze), co jest znacznie lepsze niż 0,1666 dla gracza, który spróbuje zachować wszystkie swoje łodzie. Uzyskujemy więc większą szansę na dotarcie do portu na pierwszym i drugim miejscu, zdobywając 23 punkty (12 + 11). W grze dwuosobowej ta strategia nie zadziałałaby, ponieważ drugi gracz uzyskałby 10, 9 i 8, czyli razem 27 punktów, lecz w przypadku trzech lub czterech osób staje się ona najbardziej zbliżona do optymalnej strategii istniejącej dla gry *W górę rzeki*. Jednak wybory innych graczy, losowe wyniki kości i dodatkowe czynniki sprawiają, że strategia nie zawsze zapewni wygraną; po prostu spowoduje, że wygrana stanie się bardziej prawdopodobna.

Projektowanie dla strategii

Jako projektant powinieneś brać pod uwagę kilka spraw, aby strategia miała znaczenie w Twojej grze. Na razie najważniejszą rzeczą, o której należy pamiętać, jest to, że przedstawienie graczowi wielu możliwych sposobów na wygraną wymaga od niego podejmowania trudnych decyzji strategicznych podczas gry. Ponadto gdy niektóre z tych celów są ze sobą w konflikcie, podczas gdy inne się uzupełniają (to znaczy niektóre wymagania dotyczące obu celów są takie same), w rzeczywistości może to powodować, że poszczególni gracze zaczną grać pewne role w miarę postępu gry. Kiedy gracz widzi, że zaczyna realizować jeden z celów, wybierze inne uzupełniające się cele, aby je także zrealizować, a to doprowadzi go do podejmowania taktycznych decyzji, które spełnią rolę, dla której te cele zostały zaprojektowane. Jeśli te cele spowodują, że podejmie on konkretny rodzaj działania, możliwe, że jego relacje z innymi graczami ulegną zmianie.

[5] Kia Gregory, *Chinatown Fair Is Back, Without Chickens Playing Tick-Tack-Toe*, „New York Times", 10 czerwca 2012; *http://www.nytimes.com/2012/06/11/nyregion/chinatown-fair-returns-but-without-chicken-playing-tick-tack-toe.html*.

[6] W tym przykładzie pomijam dodatkowe reguły gry ze względu na prostotę.

Przykładem powyższego rozumowania jest gra *Osadnicy z Catanu* zaprojektowana przez Klausa Teubera. W tej grze gracze zdobywają zasoby poprzez losowe rzuty kostką lub handel. Niektóre z pięciu zasobów są przydatne we wczesnej fazie gry, podczas gdy inne są przydatne na końcu. Trzy mniej użyteczne na początku to owce, pszenica i ruda, jednakże razem można je wymienić na kartę rozwojową. Najpopularniejszą kartą rozwojową jest karta żołnierza, która może przenieść żeton złodzieja w dowolne miejsce, umożliwiając okradzenie innego gracza. Dlatego gracz posiadający na początku gry nadmiar rudy, pszenicy i owiec często kupuje karty rozwojowe, a ponieważ grając największą liczbą kart żołnierzy, można zdobyć punkty zwycięstwa, kombinacja tych zasobów i potencjalny cel może tak wpłynąć na gracza, by zaczął częściej okradać innych graczy, a także silnie zachęcać go do odgrywania w grze roli awanturnika.

Reguły lokalne

Reguły lokalne pojawiają się, gdy gracze sami zaczynają modyfikować reguły gry. Jak widzieliśmy w przykładzie gry *Bartok*, nawet prosta zmiana reguły może mieć drastyczny wpływ na grę. Na przykład większość graczy w grze *Monopoly* ustanawia reguły lokalne, które pomijają licytację nieruchomości (co normalnie by się stało, gdyby gracz pojawił się na nieruchomości nieposiadanej przez nikogo i nie chciał jej kupić), a także dodaje do pola o nazwie „parkowanie bezpłatne" zestaw wszystkich kar, który zostanie wybrany przez każdego gracza pojawiającego się na tym polu. Pominięcie reguły aukcji usuwa niemal całą potencjalną strategię istniejącą od początku gry (przekształcając ją w bardzo powolny system losowej dystrybucji nieruchomości), natomiast wprowadzenie drugiej reguły likwiduje pewien determinizm (ponieważ może przynieść korzyść każdemu graczowi, bez względu na to, czy prowadzi, czy też jest na ostatnim miejscu). Chociaż pierwsza reguła lokalna z powyższego przykładu powoduje, że gra staje się nieco gorsza, większość reguł lokalnych ma na celu podwyższenie poziomu zabawy[7]. We wszystkich przypadkach reguły lokalne są przykładem działań graczy, którzy zaczynają przejmować grę na własność, co powoduje, że staje się ona trochę bardziej ich „dzieckiem", a trochę mniej dziełem projektanta. Fantastyczne w regułach lokalnych jest to, że dla wielu osób są to ich pierwsze eksperymenty z grami.

Zamiary gracza: typy Bartle'a, oszuści i psuje

Rzeczą, nad którą sprawujesz niewielką kontrolę lub nie masz jej wcale, są zamiary graczy. Chociaż większość graczy będzie grać w grę w sposób racjonalny, tak aby wygrać, być może będziesz musiał walczyć z oszustami i psujami. Nawet gracze grający zgodnie z zasadami dzielą się na cztery odmienne typy osobowości zdefiniowane przez Richarda Bartle'a, jednego z projektantów pierwszej gry MUD (*Multi-User Dungeon*, tekstowego przodka współczesnych gier sieciowych RPG).

Cztery zdefiniowane przez niego typy istnieją od czasu stworzenia przez niego gry MUD; są one obecne we wszystkich grach sieciowych przeznaczonych dla wielu graczy. Artykuł Bartle'a z 1996 roku „Hearts, Clubs, Diamonds, Spades: Players Who Suit MUDs[8]" zawiera fantastyczne

[7] Jeśli kiedykolwiek będziesz grać w grę *Lunch Money* autorstwa Atlas Games, pozwól graczom atakować przeciwników, leczyć się i odrzucać karty, których nie chcą w danej turze (zamiast wybierać tylko jedną z tych trzech opcji). To sprawi, że gra stanie się o wiele bardziej szalona!

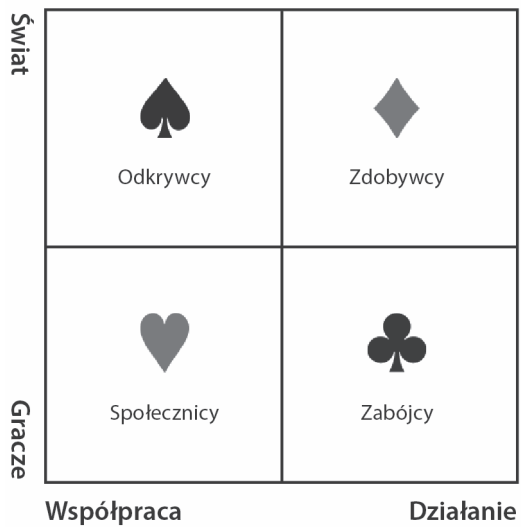
[8] Richard Bartle, „Hearts, Clubs, Diamonds, Spades: Players Who Suit Muds", *http://www.mud.co.uk/richard/hcds.htm*.

informacje o tym, jak tego typu gracze wchodzą w interakcje ze sobą i samą grą, a także porady, jak w pozytywny sposób można rozwijać społeczność graczy.

Cztery typy Bartle'a (symbolizowane przez cztery kolory kart) są następujące:

- **Zdobywca** (♦ **karo**): stara się uzyskać jak najlepszy wynik w grze. Chce dominować w grze.
- **Odkrywca** (♠ **pik**): szuka wszystkich ukrytych miejsc w grze. Chce zrozumieć grę.
- **Społecznik** (♥ **kier**): chce grać z przyjaciółmi. Chce zrozumieć innych graczy.
- **Zabójca** (♣ **trefl**): chce sprowokować innych graczy w grze. Chce dominować nad innymi graczami.

Powyższe typy można przedstawić na siatce o rozmiarach 2×2 (zdefiniowanej również w artykule Bartle'a). Została ona zaprezentowana na rysunku 5.2.



Rysunek 5.2. Cztery rodzaje graczy, które według Richarda Bartle'a pasują do gier typu MUD

Oczywiście istnieją też inne teorie motywacji i typów graczy[9], ale ta zaprezentowana przez Bartle'a jest najbardziej rozpoznawalna i rozumiana w branży gier.

Dwa inne typy graczy, z którymi możesz mieć do czynienia, to oszuści i psuje:

- **Oszuści**: dbają o zwycięstwo, ale nie przejmują się uczciwą grą. Oszust nagina lub łamie zasady, aby wygrać.
- **Psuje**: nie przejmują się wygraną ani grą. Psuj często łamie reguły gry, by zniszczyć wrażenia innych graczy.

[9] Zobacz artykuł Nicka Yeesa „Motivations of Play in MMORPGs: Results from a Factor Analytic Approach", *http://www.nickyee.com/daedalus/motivations.pdf*. Innym świetnym źródłem informacji jest artykuł Scotta Rigby'ego i Richarda Ryana „The Player Experience of Need Satisfaction (PENS)", *http://immersyve.com/white-paper-the-player-experience-of-need-satisfaction-pens-2007/*.

Żadnego z powyższych typów graczy nie chciałbyś na pewno spotkać w swojej grze, jednakże powinieneś zrozumieć ich motywacje. Na przykład, jeśli oszust czuje, że ma szansę na wygraną w sposób zgodny z prawem, może nie mieć ochoty na oszukiwanie. Psuje są o wiele trudniejsi do pokonania, ponieważ nie dbają o grę ani wygraną. Jednakże rzadko będziesz musiał sobie z nimi radzić w cyfrowych grach jednoosobowych, ponieważ nie mieliby powodu, aby grać, jeśli nie byliby zainteresowani zwycięstwem. Jednak nawet wielcy gracze mogą czasami stać się psujami, gdy napotkają okropną mechanikę gry... często tuż przed jej opuszczeniem.

Wynik

Wynik jest rezultatem grania w grę. Wszystkie gry zawierają wynik. Wiele tradycyjnych gier to **gry o sumie zerowej**, co oznacza, że jeden gracz wygrywa, a drugi przegrywa. Jednak nie jest to jedyny rodzaj wyniku, jakim może się zakończyć gra. W rzeczywistości każdy pojedynczy moment w grze ma swój własny wynik. Większość gier ma kilka różnych poziomów wyników:

- **Wynik natychmiastowy**: każda pojedyncza akcja kończy się wynikiem. Kiedy gracz atakuje wroga, wynikiem tego jest albo chybienie, albo trafienie i wynikające z tego obrażenia przeciwnika. Kiedy gracz kupuje nieruchomość w grze *Monopoly*, wynik jest taki, że gracz ma mniej dostępnych pieniędzy, ale obecnie posiada potencjał, aby w przyszłości zarobić ich więcej.
- **Wynik misji**: wiele gier wysyła gracza na misje lub wyprawy i oferuje nagrodę za ukończenie tego typu zadań. Misje i wyprawy często zawierają konstruowane wokół nich narracje [na przykład w grze *Spider-Man 2* (wyprodukowanej przez firmę Treyarch w 2004 roku) mała dziewczynka straciła swój balonik, więc Spider-Man musi go odzyskać], a wynik zadania oznacza również koniec niewielkiej narracji, która je otacza.
- **Wynik skumulowany**: wynik ten pojawia się, gdy gracz przez jakiś czas dąży do osiągnięcia celu i ostatecznie go osiąga. Jednym z najczęstszych przykładów jest zmiana poziomów w grze z tzw. punktami doświadczenia. Gracz zdobywa tylko kilka punktów doświadczenia, a kiedy ich liczba osiąga określony próg, postać gracza przechodzi na nowy poziom, który poprawia jej statystyki lub umiejętności. Główna różnica między tym wynikiem a wynikiem misji polega na tym, że wynik kumulacji zwykle nie zawiera narracji, a gracz często osiąga go w sposób pasywny, aktywnie wykonując coś innego (na przykład gracz w grze *Dungeons & Dragons 4th Edition* bierze aktywny udział w sesji gry, a sumując zdobyte punkty doświadczenia pod koniec rozgrywki, zauważa, że przekroczyły one wartość 10 000, dzięki czemu pozwalają mu osiągnąć poziom 7.)[10].
- **Wynik końcowy**: większość gier kończy się wynikiem, np. jeden gracz wygrywa w szachy (przy czym drugi przegrywa) albo kończy grę *Final Fantasy VII* i ratuje świat przed Sephirothem itd. W niektórych grach osiągnięcie wyniku końcowego nie oznacza ich zakończenia (na przykład nawet gdy gracz w grze *Skyrim* zakończy zadanie główne, może nadal pozostać w jej świecie i doświadczać innych zadań[11]). Co ciekawe, śmierć postaci gracza rzadko jest wynikiem końcowym.

[10] Rob Heinsoo, Andy Collins i James Wyatt, *Dungeons & Dragons Player's Handbook: Arcane, Divine, and Martial Heroes: Roleplaying Game Core Rules*, Wizards of the Coast, Renton 2008.

[11] Gra *Fallout 3* początkowo kończyła się, gdy gracz osiągnął wynik końcowy w postaci ukończenia głównego wątku, ale następnie pojawiły się dodatki do pobrania, które pozwalały graczom kontynuować grę nawet po tym wydarzeniu.

W kilku grach, w których śmierć jest wynikiem końcowym (na przykład w grze *Rogue* wyprodukowanej przez AI Design w 1980 roku, w której śmierć postaci powoduje, że gracz traci wszystko, co dotychczas zdobył), indywidualne sesje grania są zazwyczaj stosunkowo krótkie, aby gracz nie odczuwał ogromnej straty po śmierci postaci. Jednak w większości gier śmierć jest jedynie chwilową porażką, a umieszczone w nich punkty kontrolne zazwyczaj gwarantują, że gracz nigdy nie straci więcej niż pięć minut swojego postępu.

Estetyka dynamiczna

Podobnie jak ma to miejsce w przypadku mechaniki dynamicznej, również estetyka dynamiczna pojawia się podczas rozgrywki. Dzieli się ona na dwie różne kategorie podstawowe:

- **Estetyka proceduralna**: estetyka, która jest programowo generowana przez komputerowy kod gry (lub przez zastosowanie mechaniki w grze papierowej). Należą do nich muzyka i sztuka proceduralna, które wyłaniają się bezpośrednio z estetyki i technologii wbudowanej.
- **Estetyka środowiskowa**: to estetyka środowiska, w którym jest rozgrywana gra. W dużej mierze ten rodzaj estetyki jest poza kontrolą twórców gier.

Estetyka proceduralna

Estetyka proceduralna, którą powszechnie zauważamy w grach komputerowych, powstaje programowo, łącząc technologię i estetykę wbudowaną[12]. Nazywa się ją proceduralną, ponieważ wynika z procedur (zwanych również funkcjami), które zostały zapisane jako kod programowania. Spadający wodospad obiektów, który stworzysz w rozdziale19. „Witaj, świecie — Twój pierwszy program", można uznać za sztukę proceduralną, ponieważ jest ciekawym efektem wizualnym wyłaniającym się z kodu programowania języka C#. Dwie najbardziej powszechne formy estetyki proceduralnej w profesjonalnych grach to muzyka i sztuka wizualna.

Muzyka proceduralna

Muzyka proceduralna stała się bardzo popularna w nowoczesnych grach wideo i jest obecnie tworzona za pomocą trzech różnych technik:

- **Resekwencjonowanie poziome** (ang. *horizontal re-sequencing* — HRS): HRS zmienia kolejność kilku wstępnie zdefiniowanych sekcji muzyki zgodnie z emocjonalnym wpływem, jaki projektanci chcą wywrzeć w bieżącym momencie w grze. Przykładem HRS jest system iMUSE (Interactive MUsic Streaming Engine) firmy LucasArts, który był używany w serii gier *X-Wing*, a także w wielu grach przygodowych tego producenta. W grze *X-Wing* wcześniej skomponowaną muzykę tworzą fragmenty partytury Johna Williama do filmów *Gwiezdne wojny*. Korzystając z iMUSE, projektanci mogą odtwarzać spokojną muzykę, gdy gracz lata sobie w kosmosie; złowieszczą, gdy siły wroga mają zamiar go zaatakować; fanfary zwycięstwa, ilekroć niszczy wrogi statek lub osiąga cel itd.

[12] Przykładem sztuki proceduralnej w grach planszowych może być mapa stworzona przez stopniowe układanie płytek w grze *Carcassonne* (Klaus-Jürgen Wrede, 2000), jednakże sztuka proceduralna gier cyfrowych jest znacznie bardziej rozpowszechniona.

Istnieją również dłuższe sekcje muzyczne, które mają pętle i zapewniają tworzenie jednolitego nastroju, jak również bardzo krótkie **frazy muzyczne** (o długości jednego lub dwóch taktów), które służą do maskowania przejścia z jednego nastroju w drugi. Jest to obecnie najbardziej rozpowszechniony rodzaj technologii muzyki proceduralnej i historycznie sięga gry *Super Mario Bros.* (Nintendo, 1985), w której odtwarzana była krótka fraza, aby następnie przełączyć się na szybszą wersję podkładu muzycznego, gdy graczowi pozostało mniej niż 99 sekund do ukończenia bieżącego poziomu.

- **Synchronizacja pionowa** (ang. *vertical re-orchestration* — VRO): VRO obejmuje nagrania różnych ścieżek pojedynczego utworu, które można indywidualnie włączać lub wyłączać. Ta metoda jest bardzo często używana w grach rytmicznych, takich jak *PaRappa the Rapper* i *Frequency*. Cztery różne utwory muzyczne dostępne w grze *PaRappa* reprezentują cztery różne poziomy sukcesu gracza. Miara sukcesu jest oceniana co kilka taktów, a jeśli gracz pogorszy lub poprawi swój stan, muzyka w tle przełączy się odpowiednio na ścieżkę gorzej lub lepiej brzmiącą, aby to odzwierciedlić. W grze *Frequency* i jej kontynuacji *Amplitude* gracz steruje statkiem podróżującym tunelem, którego ściany reprezentują różne ścieżki w studyjnym nagraniu utworu. Kiedy gracz odniesie sukces w grze rytmicznej na określonej ścianie, zostaje włączona odpowiednia ścieżka muzyczna[13]. Technika VRO jest niemal wszechobecna w grach rytmicznych (z wyjątkami w postaci fantastycznej japońskiej gry rytmicznej *Osu Tatake Ouendan!* oraz jej zachodniego następcy *Elite Beat Agents*), a także stała się powszechna w innych grach, by udostępniać graczowi informacje zwrotne na temat zdrowia jego postaci, prędkości pojazdu itd.
- **Komponowanie proceduralne** (ang. *procedural composition* — PCO): technika PCO jest najrzadszą formą muzyki proceduralnej, ponieważ w trakcie realizacji wymaga najwięcej czasu i umiejętności. Zamiast zmieniać różne ścieżki muzyczne lub włączać i wyłączać wcześniej przygotowane utwory, program komputerowy tworzy muzykę z poszczególnych nut na podstawie zaprogramowanych reguł kompozycji, tempa itd. Jednym z najwcześniejszych komercyjnych eksperymentów w tej dziedzinie był program *C.P.U. Bach* Sida Meiera i Jeffa Brigsa z 1994 roku, stworzony dla konsoli 3DO. W grze *C.P.U. Bach* słuchacz (lub gracz) był w stanie wybierać różne instrumenty i parametry, zaś aplikacja miała tworzyć podobną w brzmieniu do utworów Bacha kompozycję muzyczną opartą na regułach proceduralnych.

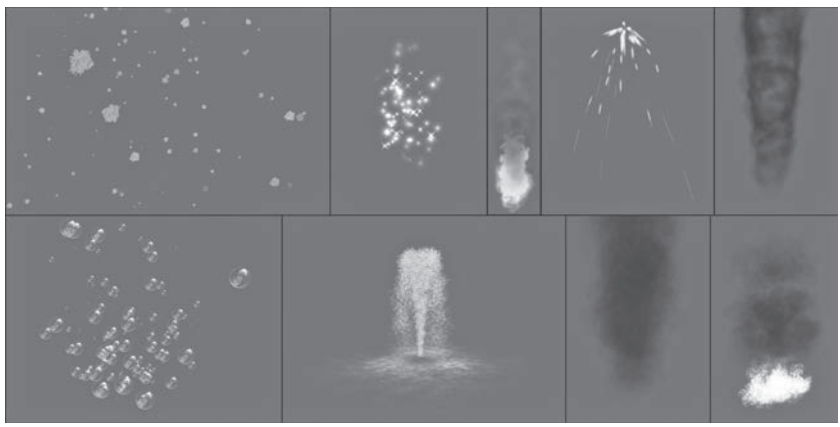
Kolejnym fantastycznym przykładem komponowania proceduralnego jest muzyka stworzona przez kompozytora i projektanta gier Vincenta Diamante dla gry *Flower* autorstwa firmy thatgamecompany (2009). Diamante stworzył w tej grze obie wstępnie skomponowane sekcje muzyki oraz zasady kompozycji proceduralnej. Podczas trwania rozgrywki, gdy gracz lata nad kwiatami rosnącymi na łące i otwiera je, zbliżając się do nich, w tle jest zazwyczaj odtwarzana muzyka (czasami zmieniana w zależności od sytuacji za pomocą techniki HRS). Każdy otwarty kwiat, kwitnąc, generuje pojedynczą nutę, która jest wybierana przez system PCO w taki sposób, by harmonijnie łączyć się ze wstępnie skomponowaną muzyką i tworzyć melodię wraz z innymi nutami kwiatowymi. Niezależnie od tego, kiedy gracz przeleci nad kwiatem, system wybierze nutę, która będzie pasować do bieżącego pejzażu dźwiękowego, w wyniku czego lot nad kilkoma kwiatami umożliwi proceduralne wygenerowanie przyjemnych melodii.

[13] Gra *Amplitude* zawiera również tryb, w którym gracze mogą wybrać, jakie ścieżki powinny być włączane w dowolnym momencie utworu. Dzięki temu VRO może służyć do stworzenia własnego remiksu utworów zawartych w grze.

Proceduralna sztuka wizualna

Proceduralna sztuka wizualna powstaje, gdy programowanie kodu w sposób dynamiczny tworzy grafikę w grze. Prawdopodobnie znasz już kilka form wizualizacji proceduralnych:

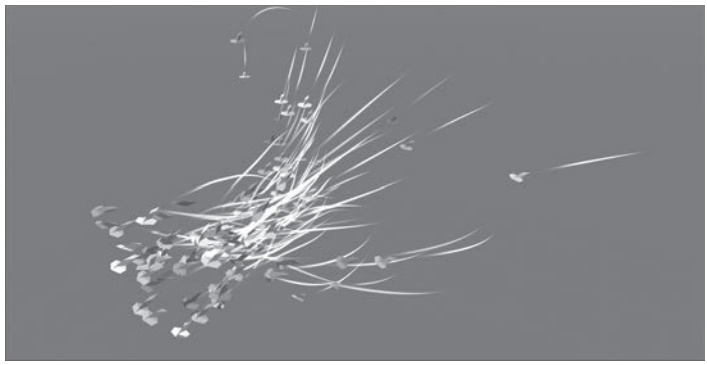
- **Systemy cząstek**: będąc najbardziej rozpowszechnioną wizualizacją proceduralną, systemy cząstek istnieją w większości gier opracowywanych w ciągu ostatnich dziesięciu lat. Chmura pyłu, która wznosi się, gdy Mario ląduje po skoku w grze *Super Mario Galaxy*, efekty ognia w grze *Uncharted 3* i iskry, które pojawiają się, gdy samochody zderzają się ze sobą w grze *Burnout*, to różne wersje efektów generowanych przez systemy cząstek. Środowisko Unity zawiera bardzo szybki i solidny silnik systemów cząstek (patrz rysunek 5.3).



Rysunek 5.3. Różne efekty graficzne oparte na systemie cząstek w środowisku Unity

- **Animacja proceduralna**: animacja proceduralna obejmuje szeroki zakres zagadnień, począwszy od algorytmów poruszania się grup stworzeń, aż po genialny mechanizm proceduralnej animacji postaci w grze *Spore* autorstwa Willa Wrighta, który generuje ruchy chodzenia, biegania, ataku i innych czynności dla każdego stworzenia, które mógłby zaprojektować gracz. Podczas normalnej animacji poruszane nią postacie zawsze podążają dokładnie za ścieżkami zaprojektowanymi przez animatora. W animacji proceduralnej poruszające się stworzenia przestrzegają reguł proceduralnych, które wyłaniają się w postaci złożonego ruchu i zachowania. W rozdziale 27. zatytułowanym „Myślenie zorientowane obiektowo" uzyskasz pewne doświadczenie związane z zachowaniem się grup znanych pod nazwą „boidy"[14] (patrz rysunek 5.4).
- **Środowiska proceduralne**: najbardziej oczywistym przykładem środowiska proceduralnego występującego w grach jest świat *Minecraft* autorstwa firmy Mojang z 2011 roku. Za każdym razem, gdy gracz rozpoczyna nową grę, na podstawie pojedynczego ziarna (*początkowej liczby generatora pseudolosowego*) w sposób losowy jest tworzony dla niego do eksploracji cały świat o rozmiarach miliardów kilometrów kwadratowych. Ponieważ cyfrowe generatory liczb losowych nigdy nie są w rzeczywistości losowe, oznacza to, że każdy, kto zacznie grę od tego samego ziarna, otrzyma ten sam świat.

[14] Wspólnie przemieszczające się, programowo generowane obiekty posiadające te same właściwości — *przyp. tłum.*



Rysunek 5.4. Boidy jako przykład animacji proceduralnej z rozdziału 27.

Estetyka środowiskowa

Inny ważny rodzaj estetyki dynamicznej dotyczy rzeczywistego środowiska fizycznego, w którym rozgrywana jest gra. Mimo że jest ona w dużej mierze poza kontrolą projektanta gry, nadal do zadań projektanta należy zrozumienie, co mogłoby powstać dzięki estetyce środowiskowej, i zebranie jak najwięcej wiedzy z tego obszaru.

Wizualne środowisko gry

Gracze będą grać w gry, stosując różne ustawienia i odmienne urządzenia, więc jako projektant musisz zdawać sobie sprawę z problemów, jakie może to powodować. W szczególności należy uwzględnić dwa elementy:

- **Jasność otoczenia**: większość twórców gier pracuje w środowiskach, w których poziom światła jest dokładnie kontrolowany, aby obrazy na ekranie były jak najbardziej czytelne. Gracze nie zawsze wchodzą w interakcje z grami w środowiskach z doskonałym oświetleniem. Jeśli gracz używa komputera znajdującego się na zewnątrz budynku, gra przy użyciu projektora lub w miejscu z niedoskonałą kontrolą oświetlenia, może mieć duże trudności z wyraźnym rozróżnianiem scen, które mają niski poziom jasności (na przykład w przypadku sceny przedstawiającej ciemną jaskinię). Pamiętaj, aby upewnić się, że Twoja estetyka wizualna zapewni uzyskanie dużego kontrastu między elementami jasnymi i ciemnymi lub pozwoli graczowi dopasować korekcję gamma albo poziom jasności wyświetlanych elementów wizualnych. Jest to szczególnie ważne w przypadku projektowania gier dla telefonów lub innych urządzeń mobilnych, ponieważ można w nie łatwo grać na zewnątrz w bezpośrednim świetle słonecznym.
- **Rozdzielczość ekranu**: jeśli projektujesz dla urządzenia z określoną rozdzielczością ekranu, takiego jak konkretny model iPada lub konsola przenośna (na przykład Nintendo DS), nie będziesz miał problemów. Jeśli jednak projektujesz grę dla komputera lub tradycyjnej konsoli do gier, masz bardzo niewielką kontrolę nad rozdzielczością lub jakością ekranu używanego przez graczy, szczególnie jeśli dotyczy to gier konsolowych. Nie możesz zakładać, że gracz będzie miał ekran o rozdzielczości 1080p lub nawet 720p. Wszystkie współczesne konsole wyprodukowane przed urządzeniami PS4 i Xbox One wciąż mogą udostępniać standardowy kompozytowy sygnał wideo, który został opracowany w latach 50. dla telewizji o zwykłej rozdzielczości. Jeśli chcesz, by gra była z nim kompatybilna, musisz użyć znacznie większego rozmiaru czcionki, aby tekst w grze był czytelny.

Nawet gry o najwyższych budżetach, takie jak serie *Mass Effect* i *Assassin's Creed*, nie były dopasowane w przeszłości do tego typu rozdzielczości, uniemożliwiając odczytanie ważnych treści na telewizorach wyprodukowanych ponad 10 lat przed ich wydaniem. Nigdy nie wiadomo, czy ktoś nie będzie próbować grać w grę na starszym sprzęcie, ale jest możliwe wykrycie tego zamiaru, a następnie odpowiednia zmiana rozmiaru czcionki.

Dźwiękowe środowisko gry

Podobnie jak w przypadku wizualnego środowiska gry, równie rzadko masz kontrolę nad otoczeniem dźwiękowym, w którym Twoja gra jest grana. Chociaż podjęcie odpowiednich działań jest bardzo ważne dla gier mobilnych, należy pamiętać o tym w każdym przypadku. Oto następujące zagadnienia do rozważenia:

- **Hałaśliwe otoczenie**: dowolna liczba zdarzeń może mieć miejsce w tym samym czasie, w którym Twoja gra jest uruchomiona, więc musisz upewnić się, że gracz będzie nadal mógł grać, nawet jeśli dźwięk będzie zakłócany lub w ogóle go nie będzie słychać. Musisz także się upewnić, że sama gra nie tworzy środowiska tak hałaśliwego, że gracz przestaje odbierać znaczące informacje. Ogólnie rzecz biorąc, ważne dialogi i instrukcje głosowe powinny być najgłośniejszymi dźwiękami w grze, a reszta tła powinna być nieco spokojniejsza. Powinieneś więc unikać subtelnych, cichych sygnałów dźwiękowych dla wszystkiego, co w grze ma znaczenie.
- **Poziom głośności kontrolowany przez gracza**: gracz może wyciszyć grę. Jest to szczególnie ważne w grach mobilnych, w których nigdy nie można liczyć na to, że gracz będzie ich słuchał. W przypadku każdej gry upewnij się, że istnieje jakaś alternatywa dla dźwięku. Jeśli udostępniasz ważne dialogi, pozwól graczowi włączyć napisy. Jeśli zaplanowałeś sygnały dźwiękowe, które przekazują pewne informacje, nie zapomnij także o odpowiednich sygnałach wizualnych.

Rozważania dotyczące graczy

Kolejnym ważnym zagadnieniem, jakie należy wziąć pod uwagę w środowisku, w którym gra jest grana, jest sam gracz. Nie wszyscy gracze mają ten sam poziom wrażliwości swoich pięciu zmysłów. Gracz, który jest niesłyszący, powinien móc grać z niewielkimi problemami, szczególnie jeśli zastosujesz się do porad z poprzednich akapitów. Istnieją jednak dwa inne przypadki, tym ważniejsze, że wielu projektantów je pomija:

- **Ślepota barw (daltonizm)**: w Stanach Zjednoczonych aż 8% mężczyzn i 0,5% kobiet z północnoeuropejskim pochodzeniem cierpi na pewien rodzaj zaburzeń w odbieraniu barw[15]. Istnieje kilka różnych rodzajów anomalii w postrzeganiu kolorów, z których najczęstsza powoduje brak możliwości odróżniania podobnych odcieni czerwieni i zieleni. Ponieważ ślepota barw jest tak powszechna, powinieneś być w stanie znaleźć kogoś znajomego mającego ten dyskomfort, kogo możesz poprosić o sprawdzenie Twojej gry, aby upewnić się, że nie istnieje w niej żadna ważna informacja przekazywana poprzez kolor w sposób, który uniemożliwia jej zauważenie. Innym świetnym sposobem przetestowania gry jest pobranie aplikacji na smartfona, która może modyfikować to, co widzisz przez kamerę, aby symulować różne rodzaje ślepoty barw[16].

[15] Ślepota barw jest znacznie częstsza u mężczyzn niż u kobiet; *https://pl.wikipedia.org/wiki/%C5%9Alepota_barw*.

[16] Chodzi tu na przykład o program *Chromatic Vision Simulator* stworzony przez Kazunori Asadę na iOS i Androida oraz *Color DeBlind* na iOS zaprojektowany przez electron software.

- **Padaczka i migrena**: migreny i napady padaczkowe mogą być spowodowane migotaniem szybko włączających się świateł, przy czym dzieci z epilepsją są szczególnie podatne na napady padaczkowe wywołane światłem. Wyświetlenie odcinka telewizyjnego programu *Pokémon* w Japonii w 1997 roku spowodowało jednoczesne pojawienie się problemów u setek widzów z powodu migotania obrazów w jednej ze scen[17]. Prawie wszystkie gry są udostępniane z ostrzeżeniem, że mogą powodować napady padaczkowe, ale faktyczne występowanie tych ataków jest obecnie bardzo rzadkie, ponieważ projektanci zaczęli brać pod uwagę wpływ, jaki ich gry mogą mieć na graczy, i w dużej mierze pozbyli się sytuacji, w których pojawiają się szybko migające światła.

Narracja dynamiczna

Istnieje kilka sposobów podejścia do narracji z perspektywy dynamicznej. Przykładem tego może być doświadczenie graczy i ich mistrza gry podczas tradycyjnej rozgrywki w grze fabularnej typu „pióro i papier". Chociaż już od ponad 30 lat istnieją eksperymenty związane z tworzeniem prawdziwie interaktywnych narracji, wciąż jednak nie osiągnęły one poziomu interakcji dobrze prowadzonej gry *Dungeons & Dragons*. Powodem, dla którego w grze *Dungeons & Dragons* można tworzyć tak fantastyczne narracje dynamiczne, jest to, że mistrz lochów (nazwa mistrza gry w *Dungeons & Dragons*) nieustannie analizuje potrzeby, lęki oraz rozwijające się umiejętności postaci swoich graczy i tworzy wokół nich historię. Jak wspomniano wcześniej w tej książce, jeśli na początkowych poziomach gracze natkną się na wroga, który (z powodu losowych rzutów kostką) jest bardzo trudny do pokonania, mistrz lochów może zdecydować, by w ostatniej chwili uciekł, a następnie powrócił jako mściciel, z którym mogą walczyć później. Osoba będąca mistrzem lochów może dostosowywać grę i fabułę do graczy w sposób, który jest bardzo trudny do powtórzenia za pomocą oprogramowania komputerowego.

Pierwotna forma narracji dynamicznej

Janet Murray, profesor Georgia Institute of Technology, opublikowała w1997 roku książkę *Hamlet on the Holodeck*[18], w której analizowała wczesną historię narracji interaktywnej w odniesieniu do historii innych form mediów narracyjnych. W swojej książce Murray bada pierwotną formę innych mediów, która dotyczy okresu pomiędzy początkowym stworzeniem a ich dojrzałą formą. Na przykład na początkowym etapie rozwoju filmu reżyserzy próbowali tworzyć 10-minutowe wersje *Hamleta* i *Króla Leara* (ze względu na 10-minutową długość pojedynczej rolki filmu 16 mm), a pierwotna telewizja była w dużej mierze tylko wizualnymi wersjami popularnych programów radiowych. Porównując wiele przykładów z różnych mediów, Murray prezentuje rozwój interaktywnej fikcji cyfrowej i jej stan w formie pierwotnej. W książce analizowane są wczesne gry przygodowe stworzone przez firmę Infocom, takie jak seria *Zork* i *Planetfall*, na podstawie których autorka zwraca uwagę na dwa bardzo atrakcyjne aspekty sprawiające, że interaktywna fikcja jest wyjątkowym zjawiskiem.

[17] Sheryl WuDunn, *TV Cartoon's Flashes Send 700 Japanese Into Seizures*, „New York Times", 18 grudnia 1997.

[18] Janet Horowitz Murray, *Hamlet on the Holodeck*, Free Press, New York 1997.

Interaktywna fikcja dzieje się przy udziale gracza

W przeciwieństwie do niemal każdej innej formy narracji interaktywna fikcja dzieje się bezpośrednio przy udziale gracza. Poniżej zaprezentowano sytuację mającą miejsce na początku gry *Zork* firmy Infocom[19] [treść wiersza za znakiem większości (np. `> otwórz klapę zapadni`) jest poleceniem wprowadzanym przez gracza].

```
…Po przesunięciu dywanu pojawia się zakurzona klapa zapadni.

> otwórz klapę zapadni
Drzwi otwierają się, niechętnie ukazując chybotliwe schody prowadzące w ciemność.

> idź w dół
Jest ciemno. Prawdopodobnie zostaniesz zjedzony przez grue.

> zaświeć lampę
Lampa świeci. Znajdujesz się w ciemnej i wilgotnej piwnicy z wąskim przejściem
prowadzącym na wschód i kanałem na południe. Na zachodzie znajduje się dół stromej
metalowej pochylni, po której nie można się wspinać.
Drzwi się zamykają, a ty słyszysz, że ktoś je blokuje.
```

Kluczowym elementem jest to, że faktycznie *słyszysz* kogoś, kto blokuje drzwi. To *Ty* jesteś teraz uwięziony. Interaktywna fikcja jest jedynym medium narracyjnym, w którym gracz (czytelnik) jest postacią podejmującą działania i odczuwającym konsekwencje narracji.

Relacje są tworzone dzięki wspólnym doświadczeniom

Innym fascynującym aspektem fikcji interaktywnej jest to, że pozwala graczowi rozwijać relacje z innymi postaciami poprzez ich wspólne doświadczenia. Świetnym tego przykładem jest gra *Planetfall*[20], kolejna tekstowa przygodówka firmy Infocom. Po zniszczeniu statku kosmicznego, w którym był zamiataczem, gracz musi sam dawać sobie radę w pierwszej części gry. W końcu napotyka maszynę do produkcji wojowniczych robotów, ale kiedy ją uruchamia, działa ona nieprawidłowo i produkuje dziecinnego, w większości bezużytecznego robota o nazwie Floyd. Floyd podąża za graczem, uzupełniając grę przede wszystkim o wątek humorystyczny. Dużo później w grze gracz musi odzyskać sprzęt z laboratorium biologicznego, które jest pełne zarówno promieniowania, jak i złośliwych kosmitów. Floyd po prostu mówi: „Floyd iść wziąć!" i wchodzi do laboratorium, aby odzyskać przedmiot. Robot wkrótce wraca, ale przecieka olejem i ledwo się porusza. Umiera w ramionach gracza, śpiewając mu „Balladę o gwiezdnych wydobywcach". Wielu graczy informowało projektanta gry *Planetfall*, Stevena Meretzky'ego, że płakali, gdy Floyd umarł. Janet Murray przedstawia to jako jeden z pierwszych przykładów namacalnego emocjonalnego związku między graczem a postacią w grze.

[19] Gra *Zork* została stworzona w Massachusetts Institute of Technology w latach 1977 – 1979 przez Tima Andersona, Marca Blanka, Bruce'a Danielsa i Dave'a Leblinga. W 1979 roku utworzyli oni firmę Infocom i udostępnili grę w postaci produktu komercyjnego.

[20] Gra *Planetfall* została zaprojektowana przez Steve'a Meretzky'ego i opublikowana przez Infocom w 1983 roku.

Narracja pojawiająca się

Prawdziwa dynamiczna narracja pojawia się, gdy gracze i mechanika gry przyczyniają się do rozwoju historii. Kilka lat temu grałem z przyjaciółmi w grę *Dungeons & Dragons 3.5*. Mistrz gry postawił nas w trudnym położeniu. Odzyskaliśmy artefakt od sił zła z innego wymiaru i byliśmy ścigani przez wielkiego balroga[21]. Zaczęliśmy uciekać wąską jaskinią na naszym latającym dywanie ku portalowi przenoszącemu do naszego wymiaru. Wróg szybko zbliżał się do nas, a nasza broń miała niewielką siłę rażenia. Przypomniałem sobie jednak, jak mało używałem posiadanej przeze mnie Różdżki Splendoru. Mogłem ją wykorzystywać raz w tygodniu w celu tworzenia „ogromnego namiotu z jedwabiu o średnicy 20 metrów, wewnątrz którego znajduje się wyposażenie i jedzenie na przyjęcie dla 100 osób"[22]. Często używaliśmy tej zdolności, by urządzać przyjęcie po zakończeniu misji, lecz tym razem stworzyłem namiot w tunelu bezpośrednio za nami. Ponieważ jaskinia miała tylko 10 metrów szerokości, balrog uderzył w namiot i zaplątał się, pozwalając nam uciec bez utraty życia.

Takie nieoczekiwane historie wynikają z połączenia sytuacji stworzonej przez mistrza gry, jej zasad i kreatywności poszczególnych graczy. Napotkałem wiele podobnych sytuacji dzięki fabularnym rozgrywkom, w których brałem udział (zarówno jako gracz, jak i mistrz gry). Masz kilka możliwości, aby zachęcić do wykorzystywania tego typu współpracy w grach fabularnych, które prowadzisz. Aby uzyskać więcej informacji na temat gier RPG i prowadzenia dobrej rozgrywki, zapoznaj się z podrozdziałem „Gry RPG" znajdującym się w dodatku B „Przydatne koncepcje".

Technologia dynamiczna

Ponieważ inne obszerne fragmenty tej książki są poświęcone zarówno technologiom cyfrowym, jak i fizycznym, w tym rozdziale znajdziesz bardzo niewiele na ten temat. Podstawową zasadą, o której musisz wiedzieć w tym momencie, jest to, że kod, który napiszesz (Twoja wbudowana technologia), będzie systemem działającym w trakcie doświadczania gry przez gracza. Podobnie jak dla wszystkich systemów dynamicznych, również w tym przypadku pojawia się emergencja, a to oznacza, że istnieje wiele możliwości niespodziewanych zdarzeń, zarówno przyjemnych, jak i trudnych. Technologia dynamiczna obejmuje wszystkie działania środowiska wykonawczego kodu, a także sposób, w jaki wpływają one na gracza, poczynając od systemu do symulacji praw fizyki, a kończąc na sztucznej inteligencji i tym wszystkim, co jest zaimplementowane w programie.

Aby odnaleźć informacje na temat dynamicznego zachowania się elementów technologii gier papierowych, takich jak kostki, obrotowe wskaźniki, karty i inne randomizery, zajrzyj do rozdziału 11. zatytułowanego „Matematyka i równoważenie gry". Informacje na temat technologii gier cyfrowych można znaleźć w dwóch ostatnich częściach książki, a także w dodatku B „Przydatne koncepcje".

[21] Balrog to olbrzymi skrzydlaty demon ognia i dymu, z którym zmierzył się Gandalf w scenie „nie przejdziesz" filmu *Drużyna Pierścienia* opartego na prozie J.R.R. Tolkiena.

[22] Systemowa dokumentacja referencyjna dla gry *Dungeons & Dragons 3.5e*; opis działania Różdżki Splendoru znajduje się pod adresem *http://www.d20srd.org/srd/magicItems/rods.htm#splendor*.

Podsumowanie

Mechanika dynamiczna, estetyka, narracja i technologia wyłaniają się z grania gry przez graczy. Choć elementy, które się pojawią, mogą być trudne do przewidzenia, projektanci powinni testować i rozumieć całą otoczkę dotyczącą emergencji.

Następny rozdział związany z tetradą warstwową dotyczy warstwy kulturowej — tej, która znajduje się poza rozgrywką. W warstwie kulturowej gracze zyskują większą kontrolę nad grami niż ich twórcy, i jest ona jedynym elementem tetrady doświadczanym przez te części społeczeństwa, które nigdy nie grają w daną grę.

SKOROWIDZ

C

D

E

F

G

H

I

J

K

L

Ł

M

N

O

P

Q

R

S

Ś

T

U

V

W

X

Z

Jaki pomysł dziś wdrożysz?

Każdy, kto chce pisać gry, poza odpowiednią wiedzą teoretyczną i znakomitymi pomysłami powinien posiadać praktyczne umiejętności korzystania z nowoczesnych narzędzi służących do tego celu. W czasach gdy napisanie i pokazanie światu nowej gry jest poważnym projektem angażującym wielu profesjonalistów z różnych branż, projektant doświadczeń interaktywnych musi podejmować wiele istotnych decyzji na dość wczesnych etapach jej rozwoju. Ważna jest również umiejętność prototypowania i przekazywania pozostałym członkom zespołu swoich koncepcji projektowych. To wszystko sprawia, że prowadzenie projektu, którego celem jest napisanie dobrej gry, staje się zadaniem trudnym i pełnym wyzwań.

Ta książka jest przeznaczona dla osób, które chcą projektować i programować gry. Przedstawiono tu kilka praktycznych teorii projektowania gier oraz praktyk pomocnych w rozwijaniu i udoskonalaniu pomysłu na projekt. Znalazło się tu również sporo wskazówek dotyczących programowania gier jako takiego. Istotnym elementem książki jest opis procesu opracowywania prototypów dla różnych gatunków gier. Każdy z omówionych przykładów uwzględnia szybkie metody przechodzenia od koncepcji do działającego prototypu cyfrowego. W publikacji wykorzystano silnik gier Unity i język programowania C#. To sprawia, że maksymalnie ułatwia ona nabycie wiedzy i umiejętności, które doceni każdy profesjonalny projektant gier!

W tej książce między innymi:

- czterowarstwowa struktura programowa i iteracyjny proces projektowania
- programowanie w języku C#
- zasady testowania gier i rozwiązywania problemów projektowych
- praca w środowisku Unity na zaawansowanym poziomie
- przykłady prototypów gier i materiały szkoleniowe

JEREMY GIBSON BOND jest projektantem gier i wykładowcą akademickim. Od 2013 roku zasiada w Katedrze Edukacji i Postępu na niezależnym festiwalu gier i zjeździe IndieCade. Przemawiał także kilka razy na konferencji Game Developers. Jest osobą o ogromnym doświadczeniu w zakresie nauczania projektowania gier i tworzenia oprogramowania. Współpracował z takimi firmami jak Human Code, Frog Design, Walt Disney Imagineering, Maxis i Electronic Arts/Pogo.com.

ISBN 978-83-283-4252-1

Cena: 149,00 zł

Pearson
Addison-Wesley