UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Balanceamento Dinâmico de *Quests* em MMORPGs

Trabalho de Graduação

Felipe Kühner Câmara dos Santos

Orientador: *Geber Lisboa Ramalho*

Recife, dezembro de 2011

UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Felipe Kühner Câmara dos Santos

Balanceamento Dinâmico de *Quests* em MMORPGs

Trabalho apresentado ao programa de Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

Orientador: *Geber Lisboa Ramalho*

Recife, dezembro de 2011

UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Felipe Kühner Câmara dos Santos

Balanceamento Dinâmico de *Quests* em MMORPGs

Trabalho apresentado ao programa de Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

Defesa: 16 de dezembro de 2011.

_____

Prof.: Geber Lisboa Ramalho
(orientador)
(UFPE)

Recife, dezembro de 2011

A Marcelo, Josely, Carolina e Rodrigo.

Agradecimentos

Agradeço primeiramente a meus pais, por todo o apoio e educação que me foi dado durante todos esses anos de vida, e por todos os incentivos em lutar por tudo aquilo que almejo.

A meu orientador, Geber Ramalho, por estar sempre de porta aberta, pelos conselhos e indicações do caminho a seguir, e por sempre exigir o melhor de mim.

Aos meus amigos, Nelson, Paulo, João e Eduardo; companheiros de quase cinco anos de curso, agradeço pelo apoio nas horas de dificuldade, dedicação, e pelos momentos de descontração.

Deixo aqui um agradecimento especial a todos os amigos e colegas do Centro de Informática pelo tempo que passamos juntos, e pelo espírito competitivo nos projetos desenvolvidos.

Por fim, agradeço a todos os professores do CIn pela dedicação e excelência demonstradas ao longo do curso e pelo entusiasmo em ajudar a superar dificuldades.

“A imaginação é mais importante que a ciência.”

Albert Einstein

Resumo

Técnicas de balanceamento dinâmico podem ser aplicadas em quase todos os gêneros de jogos, mas um gênero bastante interessante é o dos jogos online massivos (*Massively Multiplayer Online Game – MMOG*), cuja importância comercial experimentou um crescimento brusco nos últimos cinco anos, marcado por jogos de amplo escopo como World of Warcraft, e jogos casuais em redes sociais.

Um caso particular é o dos jogos online massivos de interpretação de personagens (*Massively Multiplayer Online Role Playing Game - MMORPG*), dos quais *quests* são os elementos fundamentais, representando tarefas a serem realizadas pelo jogador em troca de recompensas.

O presente trabalho propõe um modelo paramétrico de representação de *quests* adaptado para o problema de balanceamento dinâmico em MMORPGs, com uma hierarquia de classificação de *quests* inclusiva, e o desenvolvimento de uma ferramenta de edição e gerenciamento de *quests* no formato de representação especificado.

Abstract

Dynamic balancing techniques can be applied in almost all genres of games, but a very interesting genre for use is that of massively multiplayer online games (MMOG), whose commercial importance has experienced a growth spurt in the last five years, marked by a broad range of games such as World of Warcraft, and casual games on social networks.

A particular instance of MMOGs is that of the massively multiplayer online role playing games (MMORPG), of which quests are the fundamental elements, representing tasks to be performed by the player in exchange for rewards.

This paper proposes a parametric model representation of quests tailored to the problem of dynamic balancing in MMORPGs, with a hierarchical classification of quests, and the development of a tool for editing and managing quests in the specified representation format.

Sumário

1. Introdução 1

1.1. Motivação 1
1.2. Objetivo 2
1.3. Estrutura do Trabalho 2

2. Balanceamento de MMORPGs 3

2.1. *Massively Multiplayer Online Games* 3
2.1.1. MMOGs vs jogos off-line 4
2.2. *Massively Multiplayer Online Role-Playing Games* 5
2.2.1. Progressão 5
2.2.2. Quests 7
2.3. Balanceamento Dinâmico de Dificuldade em Jogos 7
2.3.1. Aplicações 10
2.4. Balanceamento Dinâmico de *Quests* 12
2.5. Metodologia de Projeto 12

3. Tipos e Variantes de Quests 15

3.1. Estado da Arte 15
3.2. Quest de Extermínio 18
3.3. Quest de Saque 20
3.4. Quest de Coleta (Cenário) 22
3.5. Quest de Assassinato 24
3.6. Quest de Escolta 26
3.7. Quest de Exploração 29
3.8. Quest de Interação (NPC) 30
3.9. Quest de Interação (Cenário) 31
3.10. Quest de Interação (Item) 33
3.11. Quest de Entrega 34
3.12. Quest de Defesa 35
3.13. Quest Composta 36
3.14. Quest Única 38
3.15. Síntese do Capítulo 38

4. Conflitos de Balanceamento 40

4.1. Conflitos de NPC 40

4.1.1. Conflitos Intra-Quest 40

4.1.2. Conflitos Inter-Quest 42

4.2. Conflitos de Item 44

4.2.1. Conflitos Intra-Quest 44

4.2.2. Conflitos Inter-Quest 45

4.3. Conflitos de Localização 45

4.3.1. Conflitos Intra-Quest 45

4.3.2. Conflitos Inter-Quest 45

4.4. Conflitos de Objetos de Cenário 46

4.5. Síntese do Capítulo 46

5. DDA Quest Designer 47

5.1. Conceitos Gerais 47

5.2. Requisitos de Sistema 47

5.2.1. Requisitos Não-Funcionais 47

5.2.2. Requisitos Funcionais 48

5.3. Projeto 49

5.3.1. Arquitetura 50

5.3.2. Tecnologias 51

5.4. Interface do Usuário 51

5.4.1. O Editor de Quests 52

5.4.2. O Gerenciador de Quests 53

5.5. Síntese do Capítulo 55

6. Considerações Finais 56

6.1. Contribuições 56

6.2. Dificuldades Encontradas 56

6.3. Trabalhos Futuros 57

7. Referências 58

Índice de Figuras

Figura 2 – Classificação Hierárquica de Quests 18
Figura 2 - Unidade Atômica de Quest 38
Figura 3 - Conflito de NPC: solução inicial 41
Figura 4 - Conflito de NPC: solução heurística 41
Figura 5 - Conflito de NPC: solução proporcional 42
Figura 6 - Editor de Quests: Boss Quest 52
Figura 7 - Editor de Quests: Composite Quest 53
Figura 8 - Filtro de Quests 54
Figura 9 - Nova Quest 54
Figura 10 - Funções de Gerenciamento 54
Figura 11 - Gerenciador de Quests 54

Índice de Quadros

Quadro 3.2.1 - Quest de Extermínio: Caso de Estudo 18
Quadro 3.2.2 – Quest de Extermínio: Parametrização 19
Quadro 3.3.1 - Quest de Saque: Caso de Estudo 21
Quadro 3.3.2 - Quest de Saque: Parametrização 22
Quadro 3.4.1 – Quest de Coleta (Cenário): Caso de Estudo 23
Quadro 3.4.2 – Quest de Coleta (Cenário): Parametrização 24
Quadro 3.5.1 – Quest de Assassinato: Caso de Estudo 25
Quadro 3.5.2 – Quest de Assassinato: Parametrização 26
Quadro 3.6.1 – Quest de Escolta: Caso de Estudo 27
Quadro 3.6.2 – Quest de Escolta: Parametrização 28
Quadro 3.7.1 – Quest de Exploração: Caso de Estudo 29
Quadro 3.7.2 – Quest de Exploração: Parametrização 30
Quadro 3.8.1 – Quest de Interação (NPC): Caso de Estudo 30
Quadro 3.8.2 – Quest de Interação (NPC): Parametrização 31
Quadro 3.9.1 – Quest de Interação (Cenário): Caso de Estudo 32
Quadro 3.9.2 – Quest de Interação (Cenário): Parametrização 33
Quadro 3.10.1 – Quest de Interação (Item): Parametrização 34
Quadro 3.11.1 – Quest de Entrega: Parametrização 34
Quadro 3.12.1 – Quest de Defesa: Caso de Estudo 35
Quadro 3.12.2 – Quest de Defesa: Parametrização 36
Quadro 3.13.1 - Quest Composta: Caso de Estudo 37
Quadro 3.13.2 - Quest Composta: Parametrização 37
Quadro 3.15.1 - Parâmetros de Quest por Categoria 39
Quadro 4.1.1 - Conflito de NPC: Reuso de Blanceamento 43
Quadro 5.2.1 - Requisitos Não-Funcionais 48
Quadro 5.2.2 - Requisitos Funcionais 49
Quadro 5.3.1 - Padrões de Projeto 51

1. Introdução

Técnicas de aprendizagem de máquina têm sido amplamente utilizadas em domínios competitivos, com enfoque na busca de estratégias ideais para maximizar uma função de avaliação. Jogos de computador podem ser vistos como ambientes competitivos com o objetivo, não de alcançar um comportamento ótimo, mas um comportamento cuja função de avaliação permaneça próxima à dos jogadores.

A área de balanceamento de jogos visa garantir um bom nível de desafio aos jogadores, evitando extremos nos quais o jogador se sinta frustrado devido à grande dificuldade do jogo, ou entediado, em consequência de um baixo grau de dificuldade.

1.1. Motivação

Técnicas de balanceamento dinâmico podem ser aplicadas em quase todos os gêneros de jogos, mas um gênero bastante discutido é o dos os jogos online massivos (*Massively Multiplayer Online Game - MMOG*). Devido ao tamanho destes jogos e à sua natureza, em grande parte baseada em atributos e parâmetros (do inglês, *stats*), na qual mesmo pequenos desequilíbrios no balanceamento podem ter impacto negativo sobre a experiência do jogador, o ajuste constante dos níveis de dificuldade tem alto potencial de sucesso [1].

O desafio em se tratando de MMOGs, uma vez que o ambiente, assim como os personagens não jogadores, é compartilhado entre todos os jogadores, consiste em minimizar o impacto do balanceamento individual no jogo como um todo. Afinal, aumentar a dificuldade para um dos jogadores não deve influenciar o nível de dificuldade dos outros.

Um caso em particular é o dos jogos online massivos de interpretação de personagens (*Massively Multiplayer Online Role Playing Game - MMORPG*), gênero cada vez mais presente no mercado de jogos e liderado por *World of Warcraft* da produtora *Blizzard,* com mais de 12 milhões de jogadores cadastrados. O grande número de jogadores simultâneos, aliado ao enfoque em parâmetros ambientais (que dificulta o balanceamento comportamental) faz dos MMORPGs o cenário ideal para o desenvolvimento de técnicas de balanceamento dinâmico [2].

1.2. Objetivo

O presente trabalho possui como objetivo elaborar uma solução para o problema de balanceamento de *quests* em MMORPGs, jogos online massivos que combinam as características de jogos de interpretação com o ambiente persistente e compartilhado dos MMOGs.

1.3. Estrutura do Trabalho

Os demais capítulos deste trabalho estão organizados da seguinte forma:

- **Capítulo 2 -** Aborda os conceitos de *Massively Multiplayer Online Games*, com ênfase nas distinções para com os tradicionais jogos off-line, e o estado da arte em balanceamento dinâmico de jogos. Por fim, a metodologia empregada no projeto é apresentada em detalhes.
- **Capítulo 3 -** Apresenta os modelos de classificação de *quests* existentes, seu contexto e problemática, e propõe uma nova classificação paramétrica adaptada às necessidades dos mecanismos de balanceamento;
- **Capítulo 4 -** Expõe as limitações da classificação proposta no Capítulo 3, e apresenta soluções e mecanismos mitigadores para tratar os problemas levantados por conflitos de balanceamento;
- **Capítulo 5 -** Apresenta a solução *DDA Quest Designer* para a edição e gerenciamento de *quests* no formato delimitado no Capítulo 3, com ênfase em seu desenvolvimento e limitações.
- **Capítulo 6 -** Encerra este trabalho apresentando as conclusões, contribuições, dificuldades encontradas e propostas de trabalhos futuros.

2. Balanceamento de MMORPGs

Neste capítulo é feita uma descrição geral dos MMORPGs, destacando conceitos chave como colaboração multiusuário [3], esquema de evolução de avatares (personificação do jogador dentro do jogo), interação jogador vs. jogador (*Player vs. Player* - PvP), formação de comunidades e persistência do mundo virtual. As diferenças para com o modelo tradicional de MMOGs são ressaltadas, e o conceito de *quest* como elemento fundamental do gênero é apresentado.

Nas seções subsequentes, é feita uma definição do campo de ajuste dinâmico de dificuldade em jogos, marcada pela apresentação dos estudos que compreendem o estado da arte na área, e de suas aplicabilidades para o balanceamento dinâmico de *quests* em MMORPGs.

Finalmente, considerações relevantes são apresentadas e discutidas.

2.1. *Massively Multiplayer Online Games*

Massively Multiplayer Online Games (MMOGs) compõem um modelo de entretenimento originado dos tradicionais jogos off-line que surgiu como forma de permitir uma interação do usuário com o jogo de forma mais realista e gratificante [4]. Conceitos como colaboração entre avatares (representando, na prática, uma colaboração entre pessoas físicas), e interações PvP, em que usuários competem uns com os outros, demonstram a mudança de foco do jogo em si para os usuários e caracterizam o sistema como facilitador de comunicação e provedor do ambiente virtual.

Essa colaboração entre usuários é caracterizada por compra, venda, e troca de itens (objetos virtuais) entre avatares, a participação conjunta em eventos e *quests* (termo que representa missões a serem completadas) e formação de comunidades de diverso porte, dentre as quais *party* (grupo pequeno com, em média, seis personagens e caráter temporário) e *guild* (grupo grande, chegando a mais de 300 personagens e caráter duradouro) merecem especial destaque. O ambiente de um MMOG favorece e incentiva a criação deste tipo de comunidade por meio de um sistema de recompensas [5].

Os primeiros MMOGs, conhecidos como *Multi-User Dungeons* (MUDs), utilizavam a aplicação telnet disponível com a maioria dos Sistemas Operacionais da época para enviar comandos textuais representando ações do avatar a um servidor remoto, que respondia com o efeito que estas obtiveram sobre o mundo virtual [6]. Comandos como *eat*, *move*, *say*, *atack* e *sleep* eram apenas alguns de uma grande lista de possíveis ações disponíveis ao usuário.

É uma característica própria de MMOGs a persistência do mundo virtual, que está em constante evolução independente da participação de um usuário específico. Dessa forma, um usuário que não acesse o sistema por certo tempo, ao retornar encontrará o ambiente diferente de quando por último o acessou. Essa mutabilidade só é possível devido ao grande número de jogadores que utiliza o MMOG com frequência. Como exemplo, um dos mais populares MMOGs atuais, *World of Warcraft*, possui mais de 8.5 milhões de jogadores no mundo todo, com em média 200 mil jogadores utilizando o sistema simultaneamente [6].

Existe ainda outro ramo de MMOGs voltados para usuários casuais, que não possuem muito tempo disponível para jogar, ou horários erráticos. Esse tipo de MMOG possui como principal característica o estímulo ou recompensa por frequência de acesso, e não mais o tempo de jogo. Muitos MMOGs, a exemplo de *World of Warcraft*, permitem que o avatar do usuário evolua mesmo quando este não está ativamente jogando.

2.1.1. *MMOGs* vs jogos off-line

A principal característica que distingue os MMOGs dos jogos off-line é o número de jogadores simultâneos. No segundo, mesmo se tratando de um jogo multiplayer (com suporte a vários jogadores), o número de jogadores geralmente não ultrapassa 16, enquanto no primeiro este número pode chegar a mais de 200 mil [7].

Outra distinção importante diz respeito ao enfoque de cada uma destas formas de entretenimento. Jogos off-line procuram construir um ambiente em que o usuário se sinta imerso por meio de gráficos avançados e poderosos mecanismos de inteligência artificial, enquanto MMOGs geralmente possuem gráficos simples e inteligência artificial fraca, preocupando-se mais com a interação entre os usuários. Como disse Will Wright, criador da popular série de jogos *The Sims*, típico exemplo

de jogo off-line, o que ele busca em seus jogos é a ilusão de comunidades [4], enquanto os MMOGs se propõem a cultivar e expandir comunidades reais.

Para os propósitos deste projeto, a persistência do mundo virtual e a interação simultânea de múltiplos jogadores com elementos do jogo são os fatores que influenciam e limitam o emprego de mecanismos de balanceamento dinâmico para o ajuste de dificuldade.

2.2. *Massively Multiplayer Online Role-Playing Games*

Um subgênero dos MMOGs, os jogos massivos de interpretação de personagem (*Massively Multiplayer Online Game* - MMORPG) unem as características de interação e formação de comunidades do gênero com elementos de evolução de personagem típicos dos jogos de interpretação de papeis (*Role-Playing* Games - RPG). Como em todos os RPGs, os jogadores assumem o papel de um personagem (em um contexto de temas variados, sendo o mais comum a fantasia medieval), assumindo controle direto das de suas ações.

Apesar de muitos dos MMORPGs modernos divergirem do padrão definido para o gênero, todos possuem uma série de fatores em comum: mecanismos de progressão de personagem, sistemas de interação social multijogadores, e customização de personagem.

Essas características, em especial a evolução e customização de personagens, proveem o objetivo implícito do jogo, fato exacerbado pela constante adição de novas opções de customização pelos desenvolvedores.

2.2.1. Progressão

O principal objetivo em quase todos os MMORPGs ou, diga-se, a principal motivação dos jogadores nesses jogos, é o desenvolvimento e evolução de seu personagem. Para tanto, é típico encontrar em MMORPGs mecanismos e sistemas de progressão de diversos tipos, sendo o mais comum o uso de pontos de experiência [8]. Ao completar ações e interagir com o ambiente do jogo, o jogador recebe pontos que podem ser acumulados e usados para avançar o "nível" de seu personagem ao atingir um pré-requisito arbitrário. Ao avançar ao próximo nível, o personagem do jogador adquire novas habilidades e aumentos em seus atributos

básicos, efetivamente tornando-o mais poderoso no contexto do jogo. Tradicionalmente, pontos de experiência são obtidos ao matar monstros, ao utilizar habilidades de criação de itens, ao completar *quests*, ou ao participar de eventos promovidos pelo jogo.

Apesar de ser o principal mecanismo de evolução de personagem, o sistema de pontos de experiência é muitas vezes complementado com sistemas auxiliares de evolução de habilidades, de conceito similar. Os pontos de habilidade podem ser obtidos da mesma forma que os pontos de experiência, em decorrência do acumulo de pontos de experiência, ou obtidos de forma única a cada jogo [7]. A conexão entre os dois sistemas é dependente de sua implementação, mas limitações garantindo evolução constante e concorrente pelos dois mecanismos é uma prática comum a muitos jogos.

Um sistema menos evidente de evolução, mas que possui grande impacto no poder dos personagens dentro do jogo é o sistema de troca de equipamentos. Trata-se de uma característica do gênero permitir que os jogadores escolham as armas e armaduras utilizadas por seus personagens, que podem ter características de ataque e defesa diferentes. A escolha de equipamento é controlada pelo custo de aquisição - quanto mais poderoso um equipamento, mais difícil é obtê-lo, que diminui em correlação com o sistema de experiência [8]. Desta forma, as armas e armaduras com melhores benefícios são mais fáceis de serem obtidas por personagens mais avançados. Apesar disto, é comum a prática de diretamente restringir o uso de equipamentos segundo o nível de experiência do personagem para garantir um melhor balanceamento.

Finalmente, o ganho de experiência e familiaridade do jogador em si para com os controles e ações do jogo pode ser o fator evolutivo empregado. O jogo *EVE Online* é um exemplo típico deste último, sendo um híbrido de MMORPG com simuladores espaciais no qual a habilidade de gerenciamento de recursos e controle de personagens determinam o sucesso e evolução do jogador [9]. É comum também o emprego ocasional deste mecanismo no ajuste de dificuldade de *quests*, colocando o jogador em uma posição na qual não possua familiaridade com os controles de seu personagem. O jogo *World of Warcraft* é conhecido por suas *quests* de veículos [10], nas quais os jogadores devem atingir um objetivo específico

estando dentro de um veiculo com controles únicos, seja montado em um animal, controlando um avião, ou controlando outro personagem.

2.2.2. *Quests*

Quests (ou missões) são os elementos base do gênero MMORPG, representando tarefas a serem realizadas pelo jogador em troca de recompensas e experiência para evolução de seu personagem.

Um MMORPG tradicional apresenta uma série de *quests* com nível de dificuldade estático e crescente, buscando abranger todos os possíveis perfis de jogadores. Na prática, o jogador acaba enfrentando períodos nos quais já efetuou as *quests* abaixo de seu nível, mas não possui experiência suficiente para concluir com sucesso as restantes.

Esse estado intermediário, no qual o jogador permanece sem objetivo e é forçado a evoluir por conta própria de modo a prosseguir com o enredo do jogo, é conhecido como *Grind* (do inglês moer, ou ranger) [11], e é o principal responsável pela perda de entusiasmo e desmotivação, que levam o jogador a parar de jogar.

Balanceamento de MMORPGs é, portanto, sinônimo de balanceamento de *Quests*, sendo o enfoque do presente trabalho. Na seção a seguir é feita a revisão de literatura das pesquisas existentes sobre balanceamento dinâmico de dificuldade em diversos gêneros de jogos.

2.3. Balanceamento Dinâmico de Dificuldade em Jogos

O processo de determinação do nível de dificuldade é reconhecido pela comunidade de desenvolvimento de jogos como elemento chave de um jogo de sucesso, sendo seu objetivo constante de manter o nível de dificuldade do jogo adaptado ao desempenho do jogador humano, independente de suas habilidades e seu progresso - diferencial típico entre jogadores novatos e especialistas [12].

Tradicionalmente, a dificuldade de um jogo aumenta de forma constante ao longo de seu progresso de forma linear, sendo possível determinar os valores dos parâmetros deste aumento, que modulam a dificuldade global do jogo, apenas selecionando estaticamente o nível de dificuldade antes de começar a jogar. Assim, é comum a ocorrência de experiências frustrantes tanto para jogadores experientes

quanto para jogadores inexperientes, uma vez que estes devem seguir uma curva de aprendizagem pré-selecionada e fixa, construída com base na experiência média do público alvo do jogo.

Balanceamento dinâmico de dificuldade tenta remediar este problema criando uma experiência feita sob medida para cada jogador. O ganho de habilidade do jogador decorrente de seu progresso no jogo deve ser refletido continuamente no nível de dificuldade adotado. É essencial, para tanto, que o sistema possa reconhecer momentos de dificuldade nos hábitos de cada jogador, e tomar providências para ajustar os elementos do jogo de forma a garantir a consistência do nível de dificuldade.

O balanceamento ideal é aquele que considera todos os perfis possíveis de jogador e possui um conjunto de parâmetros adaptados a cada um. Trata-se de um problema sem solução viável, uma vez que cada jogador possui características e preferências únicas. Desta forma, os parâmetros acabam sendo fixados em valores pré-definidos em níveis de dificuldade estáticos com base em médias e estimativas do público alvo, alienando os jogadores divergentes do padrão [13].

Manter o nível de dificuldade adequado é, portanto, um processo dinâmico. Devido às diferenças entre os jogadores e sua evolução comportamental como consequência natural da experiência adquirida ao longo do jogo, o nível de dificuldade deve ser constantemente alterado.

Segundo Arey, D. et al. [14], para ser classificado como balanceamento dinâmico, um processo deve satisfazer pelo menos três requisitos básicos, enumerados a seguir.

1. Identificar e adaptar-se ao nível inicial do jogador.
2. Identificar evoluções e regressões no desempenho do jogador.
3. Adaptar, de forma crível, os elementos do jogo para refletir o novo nível de dificuldade.

Diferentes abordagens podem ser encontradas na literatura para tratar o problema de balanceamento dinâmico de dificuldade em jogos, também conhecido como *Dynamic Difficulty Adjustment* (DDA). Em todos os casos é preciso definir

medidas de determinação da dificuldade do jogador em dado momento, geralmente por meio de uma função heurística denominada por alguns autores como "Função Desafio" (do inglês "challenge Function"), que mapeiam um estado do jogo a um valor que representa a dificuldade encontrada pelo jogador naquele momento.

Duas abordagens principais de balanceamento são identificadas por Pfeiffer, B. et al. [12] para adaptar dinamicamente os elementos e parâmetros de um jogo ao novo nível de dificuldade: a abordagem comportamental, na qual o comportamento dos personagens não jogadores (*Non-Player Character - NPC*) é alterado por mecanismos de aprendizagem de máquina, tais como algoritmos genéticos [1]; e a abordagem ambiental, na qual apenas os parâmetros ambientais são modificados.

Andrade, G et al. [15] propõem a utilização de uma versão modificada do algoritmo de aprendizagem por reforço Q-Learning para alterar o comportamento dos agentes de um jogo de luta em tempo real. Para tanto, os autores dividem o problema de balanceamento dinâmico em duas dimensões: competência (saber tão bem quanto possível) e desempenho (agir tão bem quanto necessário).

O mecanismo de aprendizagem proposto aborda ambas as dimensões, por meio de uma etapa inicial de aprendizagem off-line de *bootstrapping*, seguida de uma etapa de uma etapa online no decorrer do jogo. Isso pode ser feito deixando o agente jogar contra si próprio ou contra uma série de jogadores de teste em uma etapa de autoaprendizagem para definir a dificuldade inicial.

A ideia desenvolvida consiste em encontrar a política ideal de escolha de ações pelos agentes que proporcionem um equilíbrio no nível de dificuldade, ou seja, que proporcione a ilusão de dificuldade pretendida. Para uma dada situação, se o nível do jogo está muito difícil, os agentes podem não escolher as ações ideais em detrimento de ações cada vez mais suboptimais. Da mesma forma, se o nível de jogo se torna muito baixo, os agentes passam a escolher as ações cujos valores de desempenho são mais elevados até alcançar o desempenho ideal.

Na mesma linha de agentes inteligentes, abrangida pelo balanceamento comportamental, Demasi e Cruz [16] definiram agentes que empregam algoritmos genéticos para fazer a seleção dos agentes que melhor se adequam a cada jogador.

O método promove a evolução de agentes com um sistema co-evolutivo online para acelerar o processo de aprendizagem, utilizando modelos pré-definidos de agentes com boas características genéticas como ancestrais nas etapas do algoritmo. Similarmente à abordagem empregada por Andrade, G. et al. [15], esses modelos são construídos de forma off-line por meio de um processo de auto aprendizagem.

Uma abordagem para o balanceamento ambiental, proposta por Hunicke, Robin et al. [1], utiliza uma análise probabilística das variáveis envolvidas em um jogo de tiro em primeira pessoa para prever o comportamento do jogador de modo a alterar os parâmetros de balanceamento do jogo. Se o jogo está muito difícil, por exemplo, o jogador pode receber mais armas e pontos de vida, ou os inimigos podem se tornar menos frequentes ou possuírem vitalidade reduzida.

Essa abordagem possui bons resultados, mas para jogos em que as ações dos personagens não jogadores são complexas e influenciam diretamente na dificuldade do jogo, como é o caso em jogos de tiro, a alteração dos parâmetros ambientais pode se tornar evidente ao jogador. Nesse caso, uma abordagem comportamental associada ao balanceamento ambiental pode trazer melhores resultados.

2.3.1. Aplicações

Apesar do uso das técnicas descritas anteriormente não ter se propagado ao mercado de jogos devido à mudança brusca de paradigma de desenvolvimento, certas aplicações merecem destaque por alcançar o equilíbrio entre esquemas tradicionais de dificuldade e o uso imersivo de mecanismos de balanceamento de dificuldade. Em muitos casos, trata-se de cenários simples e restritos, cujo objetivo principal é a transparência para com o jogador.

O jogo *Left 4 Dead* (2008) é um exemplo de aplicação híbrida comportamental e ambiental, utilizando uma tecnologia de inteligência artificial apelidada de "Diretor AI" (do inglês "*AI Director*"), que controla a experiência cooperativa proporcionada pelo jogo [17].

Os ambientes multijogadores são criados de forma procedural pelo diretor, que monitora o desempenho individual dos jogadores e como estes trabalham juntos

em grupo para controlar o ritmo do jogo, determinando o número de zumbis encontrados pelos jogadores, assim como o posicionamento do chefe de cada fase. Além desses elementos, o diretor é responsável pela gestão dos elementos audiovisuais que acompanham e enfatizam as alterações efetuadas pelo processo de balanceamento.

Um exemplo típico de balanceamento ambiental é a série de jogos *Mario Kart*, composta por jogos de corrida com personagens da *Nintendo*. Todos os jogos da série envolvem a coleta de itens com efeitos diversos durante as corridas, que ajudam o jogador a ultrapassar os adversários [18].

A forma como esses itens são distribuídos ao longo da pista com base na colocação do motorista é um exemplo simples de aplicação de balanceamento dinâmico. Um jogador perto da última colocação, por exemplo, possui melhores chances de obter itens com efeitos mais poderosos e abrangentes que jogadores nas primeiras colocações. Trata-se de um caso simples, que contradiz a premissa de complexidade da área.

No jogo *Residente Evil 5*, de 2009, é empregado um sistema apelidado de "Escalador de Dificuldade" [19], que atribui uma nota na escala de 1 a 10 para o desempenho do jogador, e ajusta tanto o comportamento dos inimigos, caracterizados pelos ataques usados em cada situação, quando os fatores ambientais de dano causado e resistência dos inimigos, com base no desempenho dos jogadores.

A função de desempenho empregada contabiliza o número de mortes, ataques críticos e tempo de finalização dos estágios em uma heurística. O jogo permite a definição manual do nível de dificuldade utilizando a mesma escala de 1 a 10, mas efetua ajustes no decorrer do enredo.

Outros exemplos incluem jogos casuais, como *Flow* (2006), e mesmo jogos de interpretação de personagens, a exemplo da série *The Elder Scrolls*, que emprega uma variação de balanceamento dinâmico no ajuste das *quests*. Apesar disto, o gênero MMOG não possui nenhum exemplo representativo de aplicação de técnicas de balanceamento dinâmico.

2.4. Balanceamento Dinâmico de *Quests*

Conforme explicitado nas seções anteriores, poucos são os trabalhos efetuados voltados diretamente ao balanceamento de jogos MMOG e suas derivações. Para elaborar um processo de balanceamento voltado exclusivamente aos jogos do gênero MMORPG faz-se, portanto, necessário incorporar diversos elementos das técnicas desenvolvidas para os jogos off-line a um novo modelo de representação adaptado às características do gênero.

Mostrou-se que, como instância de MMOG, os jogos MMORPG são voltados a interação multijogador com o ambiente, sendo caracterizados por baixa complexidade comportamental e alto grau de parametrização ambiental. Desta forma, um processo de balanceamento ambiental melhor se enquadra a esse tipo de jogo.

Além disto, o balanceamento de MMORPGs está diretamente relacionado ao balanceamento dos mecanismos de progressão e evolução de personagem, visto que o objetivo, neste caso, é garantir o equilíbrio de dificuldade das ações que levam à obtenção de pontos de experiência ou habilidade. Um monstro cujo extermínio acarreta ganho de experiência deve possuir dificuldade condizente com o nível de experiência do jogador.

Para tanto, é possível balancear diretamente os elementos do jogo associados ao ganho de experiência, tais como os personagens não jogadores (NPC), as *quests*, os eventos, e as interações e ações dos jogadores. Em contrapartida, a *quest*, tal como apresentada anteriormente, é o elemento fundamental de um MMORPG, sendo responsáveis pelo posicionamento de todos os outros elementos citados. O problema se reduz deste modo do problema de balanceamento de MMORPGs ao problema de balanceamento de quests.

2.5. Metodologia de Projeto

Este trabalho segue a metodologia definida por Baily, C. et al. [20], segundo a qual é de importância crítica reconhecer que ajustes podem ser feitos ao ambiente e aos personagens não jogadores, bem como quando e como estes podem ser efetuados. Identificam-se desta forma três etapas principais, enumeradas a seguir.

1. Determinação dos parâmetros de balanceamento.
2. Definição do mecanismo de identificação de desbalanceamento.
3. Definição do mecanismo de balanceamento.

Devido ao amplo escopo abrangido por estas etapas, este projeto possui enfoque apenas na identificação dos parâmetros de balanceamento de MMORPGs. No contexto deste trabalho, tais parâmetros de balanceamento são os elementos constituintes de uma *Quest*.

O processo de identificação destes envolve, em um primeiro momento, a elaboração de uma classificação unificada e representativa do domínio. Isto se faz necessário pois diferentes tipos de *Quests* podem possuir conjuntos de parâmetros ambientais diferentes e, em vários casos, não presentes em outros tipos.

As classificações propostas por J. Doran et al. [21] [22] e Pita, James et al. [23] são fundamentais para este propósito, pois abrangem uma análise compreensiva de vários MMORPGs embora, por possuírem objetivos e estruturação específicos às necessidades dos autores, pouco se adequam ao processo de balanceamento dinâmico. Na classificação proposta por J. Doran et al. [21] [22], por exemplo, é apresentada uma gramática livre de contexto para geração procedural de *Quests*, construída após análise de 750 *Quests* de diversos jogos.

A identificação de parâmetros de balanceamento, sendo feita apenas pela análise individual de *quests*, levanta uma série de problemas quando a interação simultânea de múltiplos jogadores é considerada. Conflitos de balanceamento diretamente associados à escolha destes parâmetros devem ser resolvidos antes da definição dos mecanismos de balanceamento.

De modo a fundamentar a classificação paramétrica de *quests* definida, uma ferramenta de edição e gerenciamento de *quests* foi desenvolvida e implementada. Esta ferramenta, apesar de não validar a base teórica elaborada, permite uma verificação de integração das categorias concebidas, comprovação experimental de viabilidade do processo, e é essencial para a prototipação de jogos e sistemas de balanceamento dinâmico que possam validar os parâmetros escolhidos.

O escopo do projeto impossibilita, entretanto, que tal comprovação experimental seja feita neste momento, a ferramenta desenvolvida sendo apenas o primeiro passo em direção a uma implementação completa.

A metodologia definida acima pode ser resumida, portanto, às seguintes etapas:

1. Estado da arte de modelos de classificações de quest.
2. Elaboração de uma nova classificação
3. Definição das variantes de *quests*.
4. Identificação dos parâmetros de balanceamento de cada variante.
5. Identificação dos pontos de conflito de parâmetros.
6. Elaboração de uma ferramenta de edição e gerenciamento de *quests*.

3. Tipos e Variantes de Quests

Neste capítulo, é apresentado um novo modelo de classificação e representação de *quests* em MMORPGs com base na análise de múltiplos jogos e em esquemas de classificação existentes. Primeiramente, é apresentada uma análise dos trabalhos realizados na área relacionados ao problema de classificação unificada de *quests*, seguido da elaboração de um novo conceito classificador adaptado às necessidades dos mecanismos de balanceamento dinâmico, alvo deste estudo.

3.1. Estado da Arte

De modo a generalizar os parâmetros identificados nas instancias analisadas a uma *quest* qualquer, um mecanismo de classificação se faz necessário. Com este propósito, as classificações propostas por Doran, J. et al. [21], Pita, James et al. [23], Walker, Jill [24] e WoWWiki [25] foram utilizadas como base para elaboração de uma nova classificação que agregue as *quests* com base em similaridade de parâmetros.

A classificação proposta por Doran et al. [21] toma como base a motivação dos NPCs ao propor *quests* aos jogadores e descreve uma gramática para elaboração procedural de *quests* com base nesta classificação. Neste estudo, 750 *quests* de diversos jogos foram analisadas e reconstruídas como regras de produção da gramática desenvolvida. Esta classificação, no que diz respeito às necessidades deste trabalho, apresenta ampla cobertura do domínio de *quests*, mas provê separações de categoria que não impactam no problema de parametrização. Como exemplo deste fato, as *quests* do tipo *"Check on NPC"* e *"Interview NPC"* consideradas distintas por Doran et al. [22], foram identificadas como envolvendo as mesmas variáveis de ambiente (descritas em mais detalhes nas seções a seguir) após a etapa inicial de identificação de parâmetros.

Pita, James et al. [23], por outro lado, propõem uma classificação fundamentada na interação entre objetos do mundo virtual, segundo a qual um objeto representa um personagem jogador, um personagem não jogador, ou um item. Neste contexto, uma *quest* é um processo que envolve **N** objetos de entrada e

M objetos de saída. Um exemplo disto pode ser visto na *quest* do tipo "*Kill*", que consiste em uma função com dois objetos de entrada - o personagem jogador e um personagem não jogador - e um objeto de saída - o personagem jogador.

Quadro 3.1 – Comparação das diferentes classificações de Quest

WowWiki	Pita, James et al.	Walker, Jill	Doran, Jonathon et al.	Classificação Proposta
Kill	*Kill*	*Kill*	*Kill pests*	**Extermínio**
			Kill enemies	**Assassinato**
			Attack enemy	
			Revenge, Justice	
			Capture Criminal	
Loot	*Steal*	*Collect*	*Steal stuff*	**Saque**
			Steal supplies	
			Steal valuables for resale	
			Recover lost/stolen item	
Loot Container	*Retrieve*		*Obtain rare items*	**Coleta (Cenário)**
			Obtain luxuries	
			Gather raw materials	
			Obtain training materials	
Explore	*Discover*	*Explore Area*	*Visit a dangerous place*	**Exploração**
			Spy	
Make		*Learn*	*Use existing tools*	**Interação (Item)**
			Assemble repair	
			Treat or repair	
			Make valuables for resale	
			Assemble tool for new skill	
Interact			*Research a skill*	**Interação (NPC)**
			Practice combat	
			Practice skill	
		Interact	*Check on NPC*	
			Interview NPC	
Deliver			*Deliver item for study*	**Entrega**
			Deliver supplies	
			Trade for supplies	
Escort			*Guard Entity*	**Defesa**
			Attack threatening entities	**Escolta**
Unique			*Create Diversion*	**Interação (Cenário)**
			Assemble fortification	
			Rescue captured NPC	**Composta**
	-	-	-	**Única**

A metodologia empregada neste projeto para o processo de parametrização de *quests* teve por base uma abordagem *Bottom-Up*, na qual 260 instâncias de *quests* provenientes dos jogos *World of Warcraft*, *Everquest* e *Fly For Fun* foram analisadas e seus parâmetros de balanceamento identificados. As descrições das *quests* analisadas foram recuperadas dos repositórios online Wowhead [26], Allakhazam MMO Quest Databases [27] e MMODB Quest Databases [28], e associadas com os resultados provenientes dos trabalhos referenciados acima para compor um novo modelo de classificação e representação de *quests*.

O quadro 3.1 evidencia as diferenças de classificação das referências citadas acima para com a classificação proposta nas subseções subsequentes. Para manter uniformidade com os estudos referenciados, os termos originais foram mantidos em inglês.

Os quadros apresentados em cada subseção a seguir evidenciam os parâmetros característicos de cada tipo de *quest* da classificação definida. A coluna *Qtd* representa o número mínimo e máximo de instancias de cada parâmetro, sendo um parâmetro caracterizado como uma coleção de atributos que podem ser alterados de forma independente ou em conjunto. Cada atributo possui um tipo associado – *Numérico, Booleano, ou Coleção* –, uma breve descrição de seu significado, e uma estimativa do impacto de sua alteração em jogos online multijogadores.

Para cada categoria é apresentada também uma instância de *quest* decomposta segundo parâmetros relevantes. Múltiplas instâncias deste tipo foram analisadas para cada categoria de modo a compor um conjunto de parâmetros representativos.

Para a classificação desenvolvida, categorias com parâmetros similares foram agrupadas de forma hierárquica. O diagrama a seguir representa esta visão hierárquica, na qual é possível identificar três conjuntos de *quests* agrupadas: Caça, Coleta, e Interação.

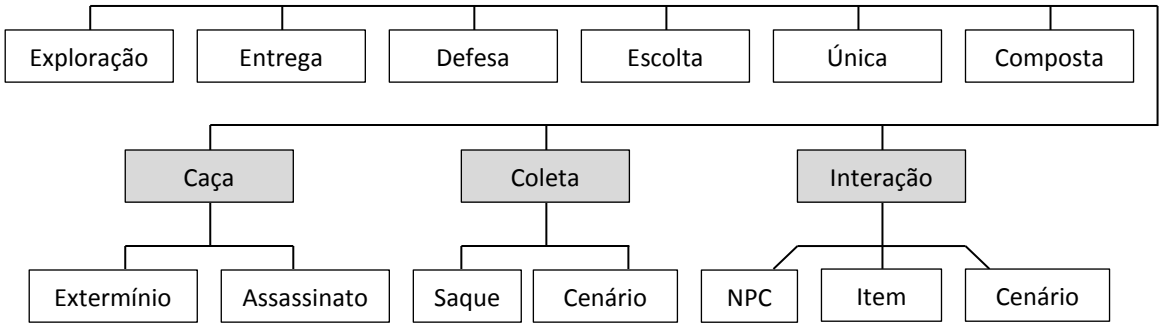

Figura 1 – Classificação Hierárquica de Quests

3.2. Quest de Extermínio

Quests de extermínio, primeira subcategoria das *quests* de caça, são *quests* nas quais é solicitado ao jogador matar uma quantidade específica de certo NPC não único presente em diversos locais no mundo do jogo. Esse tipo de *quest* pode ou não determinar um período de tempo dentro do qual o jogador deve atingir a cota solicitada, e se restringe geralmente a locais próximos de onde a *quest* foi obtida. Um caso de estudo representativo de *quests* de extermínio pode ser visto abaixo:

Quadro 3.2.1 - Quest de Extermínio: Caso de Estudo

Nome	**Jogo**	
Kill 'em All!	World of Warcraft	
NPC Inicial	**NPC Final**	
General Twinbraid	General Twinbraid	
Nível Mínimo	**Nível da Quest**	
32	35	
Descrição		
"I want to see some dead goblins, <name>! Just west of here you'll find a whole warren of those little insects burrowing into the mountain. Let's see to it that they're digging their grave. Destroy them. Lay waste to their whole grubby operation. They've sown their own demise in Bael'dun, <class>, and it is time to meet the reaper."		
Objetivo		
1. Matar 10 *Frazzlecraz Miners* na *Frazzlecraz Motherlode* em *Southern Barrens*.		
Recompensas		
1. 3,300 pontos de experiência 2. 250 pontos de reputação com *Ironforge* 3. 35 moedas de prata		
Parâmetros		
Iniciais	**Menor Dificuldade**	**Maior Dificuldade**
1. NPC	--	--
a. Nome: Frazzlecraz Miners	--	--
b. HP: 1,163 - 1,220	↑	↓
c. Reativo: Ataques (Não hostil)	--	--
2. Quantidade: 10	↓	↑
3. Localização: Frazzlecraz Motherlode	↑	↓

Neste exemplo, do jogo *World of Warcraft*, o NPC “General Twinbraid” solicita que o jogador mate 10 unidades do monstro comum “Frazzlecraz Miners” localizado em uma região específica do mapa. Este tipo de *quest* é típico de jogos MMO, sendo fácil de parametrizar. Neste caso, para tornar a *quest* mais difícil ou mais fácil, pode-se alterar o NPC ou monstro alvo, seus atributos, a forma como este reage ao personagem jogador (hostil/não hostil), a quantidade alvo de monstros e sua localização. Além destes, é importante levar em consideração parâmetros não imediatamente aparentes como o tempo limite para realização da *quest*.

O resultado da análise de múltiplas *quests* desta categoria pode ser observado no quadro 3.2.2. No caso específico das *quests* de extermínio, caso o enredo requeira mais de um NPC alvo, os parâmetros correspondentes - *NPC Alvo, Quantidade e Localização* - podem ser repetidos para cada instância de monstro.

Quadro 3.2.2 – Quest de Extermínio: Parametrização

Qtd.	Parâmetro	Atributo	Tipo	Descrição
[1, ∞[	NPC alvo	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Quantidade	--	Numérico	Quantidade de NPCs a serem exterminados.
	Localização	Spawn Point	Numérico	Localização física.
		Frequência	Numérico	Frequência de aparecimento.
		Quantidade Máxima	Numérico	Quantidade máxima de NPCs em um dado momento.
		Conhecida	Booleano	Indica se a localização é fornecida ao jogador.
[0, ∞[	NPC Colateral	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física.
		Frequência	Numérico	Frequência de aparecimento.
		Quantidade Máxima	Numérico	Quantidade máxima de NPCs em um dado momento.
[0, 1]	Tempo	--	Numérico	Tempo limite.
1	NPC Final	Localização	Numérico	Localização do NPC.

É importante notar também a possibilidade de incluir NPCs adicionais, não diretamente relacionados à *quest*, para servir de obstáculo à sua execução pelo jogador. Um exemplo simples do uso destes NPCs colaterais consiste em dispersar monstros agressivos em regiões que devem ser atravessadas pelo jogador para realizar a quest. A escolha de interagir com estes NPCs colaterais cabe apenas ao

jogador, mas sua agressividade torna a *quest* mais difícil, sem a necessidade de alterar a *quest* em si.

O uso deste recurso pode ocorrer não apenas em *quests* de extermínio como também em grande parte das *quests* descritas nas seções a seguir.

No contexto dos jogos online, em que os mesmos NPCs interagem com diversos jogadores, a adaptação dos atributos do NPC alvo ao nível de cada jogador traz uma série de problemas que podem ou não ser resolvidos pelo mecanismo de combate empregado pela engine do jogo. Em particular, se a engine permite múltiplos jogadores atacar o mesmo NPC simultaneamente, a alteração de seus atributos não pode ser efetuada, uma vez que há conflito entre a as versões diferentes do mesmo NPC ajustadas para cada jogador.

Dessa forma, os problemas decorrentes de conflitos de parâmetros devem ser tratados para garantir sua viabilidade para com os mecanismos de balanceamento. O próximo capítulo deste trabalho busca expor estes conflitos e apresentar soluções possíveis.

3.3. Quest de Saque

Outro tipo de *quest* bastante presente em jogos de RPG online, *quests* de coleta solicitam ao jogador obter certa quantidade de itens, seja de um monstro/NPC ou de um container diretamente no mundo do jogo. Estes itens podem ser componentes tradicionais do jogo, armazenados no inventário do personagem jogador, ou desenvolvidos especificamente para a *quest* em questão, muitas vezes indicados somente como uma mensagem de obtenção.

Quests de saque são uma subcategoria das *quests* de Coleta na qual o jogador obtém o item solicitado após matar um monstro/NPC que o possua e “ativar” ou recuperar seu corpo. Geralmente, múltiplos NPCs da mesma região ou com similaridades de tipo e nível podem possuir este mesmo item com diferentes probabilidades.

Um exemplo de *quest* de Saque encontra-se no quadro a seguir:

Quadro 3.3.1 - Quest de Saque: Caso de Estudo

Nome	**Jogo**
Wolves Across the Border	World of Warcraft
NPC Inicial	**NPC Final**
Eagan Peltskinner	Eagan Peltskinner
Nível Mínimo	**Nível da Quest**
1	2
Descrição	
"Whatever is affecting these wolves isn't going away. Each day that passes brings forth more sick wolves. It pains me to ask anyone to kill nature's beasts, but we must destroy the diseased young wolves before this little outbreak turns into an epidemic. Kill diseased young wolves in the valley and bring me their diseased wolf pelts as proof of your deed."	
Objetivo	
1. Coletar 8 *Diseased Wolf Pelts*.	
Recompensas	
1. À escolha do jogador: a. *Northshire Abbot's Shoes* b. *Borderlands Bracers* c. *Wolf Hunter's Vest* 2. 170 pontos de experiência. 3. 250 pontos de reputação com *Stormwind*.	

Parâmetros		
Iniciais	**Menor Dificuldade**	**Maior Dificuldade**
1. Item: *Diseased Wolf Pelt*	--	--
2. Quantidade: 8	↓	↑
3. Obtido de NPC:	↑	↓
a. Diseased Timber Wolf (Lvl 1) – 77%	↑	↓
b. Young Wolf (Lvl 1) – 14%	↑	↓
c. Mangy Wolf (Lvl 2-6) – 0%	--	↑
d. Gray Forest Wolf (Lvl 7-8) – 0%	--	↑
e. Prowler – 0%	--	↑
4. Localização: Elwynn Forest	--	--

Neste exemplo, do jogo *World of Warcraft*, o NPC inicial da *quest* deseja que o jogador obtenha 8 unidades do item "*Diseased Wolf Pelts*". Neste caso em particular, o monstro indicado pelo NPC que deve ser atacado para obter tal item é "Young Wolf", mas também é possível obtê-lo de "Deseased Timber Wolf".

Quests deste tipo são semelhantes às *quests* de Extermínio descritas anteriormente, com a adição do parâmetro referente ao item a ser obtido e a probabilidade do NPC especificado possuí-lo. O quadro 3.3.2 resume os parâmetros identificados para *Quests* de saque.

Novamente, o tempo limite para completar a *quest* e o uso de NPCs Colaterais podem servir como parâmetros adicionais de balanceamento.

Quadro 3.3.2 - Quest de Saque: Parametrização

Qtd.	Parâmetro	Atributo	Tipo	Descrição
[1, ∞[	NPC alvo	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Item	Drop rate	Numérico	Frequência com que o NPC possui o item.
	Quantidade	--	Numérico	Quantidade de itens a serem obtidos.
	Localização	Spawn Point	Numérico	Localização física dos NPCs.
		Frequência	Numérico	Frequência de aparecimento dos NPCs.
		Quantidade Máxima	Numérico	Quantidade máxima de NPCs em um dado momento.
		Conhecida	Booleano	Indica se a localização é fornecida ao jogador.
[0, ∞[	NPC Colateral	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física.
		Frequência	Numérico	Frequência de aparecimento.
		Quantidade Máxima	Numérico	Quantidade máxima de NPCs em um dado momento.
[0, 1]	Tempo	--	Numérico	Tempo limite.
1	NPC Final	Localização	Numérico	Localização do NPC.

3.4. Quest de Coleta (Cenário)

A segunda subcategoria das *quests* de Coleta corresponde às *quests* deste tipo nas quais os itens solicitados podem ser obtidos diretamente do ambiente do jogo por meio de uma série de containers auto regeneráveis acessíveis a qualquer jogador.

A dificuldade neste tipo de *quest*, que apresenta características semelhantes à categoria de *quests* de Interação, reside na localização dos containers (perigo inerente à região) e à sua taxa de aparecimento, ou "spawn". Um exemplo de *quest* do jogo *World of Warcraft* é apresentado no quadro 3.4.1.

Para essa *quest*, é solicitado ao jogador obter 10 unidades de "*Creature Carcass*", que se encontram em containers espalhados pela região "*Thousand Needles*". Para tornar a *quest* mais difícil pode-se alterar a localização dos containers para uma região de maior perigo, com monstros agressivos; aumentar a quantidade de itens necessários, de modo a forçar o jogador a passar mais vezes pela região de risco; ou ainda alterar a frequência de aparecimento dos containers, o

que tem por efeito obrigar o jogador a gastar mais tempo aguardando o "*respawn*". Estas alterações proveem maior dificuldade à *quest* de forma indireta, uma vez que esta não possui nenhuma dificuldade inerente a seu objetivo.

Quadro 3.4.1 – Quest de Coleta (Cenário): Caso de Estudo

Nome	**Jogo**	
Carcass Collection	World of Warcraft	
NPC Inicial	**NPC Final**	
Ajamon Ghostcaller	Ajamon Ghostcaller	
Nível Mínimo	**Nível da Quest**	
38	42	
Descrição		
"These organs be a good start, mon, but we still be needin' lotsa other tings. Normally, we'd be gettin' stuff from family now, but dats only me and my sistah's boy, and Tony ain't worth it! So, we be substitutin' some stuff. Lotsa carcasses, in fact! Deep down in the Shimmering Deep, there's tons of dem from when the water came rushin' in. Killed all of dem creatures. Go down there and bring me up some."		
Objetivo		
1. Coletar 10 *Creature Carcasses*.		
Recompensas		
1. 55 moedas de prata. 2. 4,820 pontos de experiência.		
Parâmetros		
Iniciais	**Menor Dificuldade**	**Maior Dificuldade**
1. Item: *Creature Carcass*	--	--
2. Quantidade: 10	↓	↑
3. Disponibilidade (*Spawn rate*)	↑	↓
4. Localização: *Thousand Needles*	--	--

Similarmente às *quests* de Interação (Cenário) apresentadas mais à frente, os parâmetros identificados para *quests* de Coleta (Cenário) se referem ao objeto ativável ou danificável (neste caso, agindo como container para o item), sua localização e a probabilidade deste possuir o item desejado (denominada "drop rate"). O quadro a seguir sumariza estes parâmetros.

Quadro 3.4.2 – Quest de Coleta (Cenário): Parametrização

Qtd.	Parâmetro	Atributo	Tipo	Descrição
[1, ∞[	Objeto de Cenário	Resistência	Numérico	Resistência
		Ativável	Booleano	Se verdadeiro, o jogador pode ativar diretamente o objeto.
		Consumível	Booleano	Indica se o objeto é consumido após ativação.
	Item	Drop rate	Numérico	Frequência com que o Objeto possui o item.
	Quantidade	--	Numérico	Quantidade de itens a serem obtidos.
	Localização	Spawn Point	Numérico	Localização física dos objetos.
		Frequência	Numérico	Frequência de aparecimento dos objetos.
		Quantidade Máxima	Numérico	Quantidade máxima de objetos em um dado momento.
		Conhecida	Booleano	Indica se a localização é fornecida ao jogador.
[0, ∞[	NPC Colateral	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física.
		Frequência	Numérico	Frequência de aparecimento.
		Quantidade Máxima	Numérico	Quantidade máxima de NPCs em um dado momento.
[0, 1]	Tempo	--	Numérico	Tempo limite.
1	NPC Final	Localização	Numérico	Localização do NPC.

3.5. Quest de Assassinato

Semelhante às *quests* de Extermínio, *quests* de Assassinato requerem ao jogador matar um personagem não jogador, com a diferença de que nestas últimas o alvo é único ao mundo do jogo. Estas *quests*, também conhecidas como "*Boss Quests*" em jogos como *World of Warcraft*, apresentam este NPC como elemento de enredo, de modo que apenas um jogador pode completar a *quest* em um dado momento.

O quadro 3.5.1 exemplifica esta categoria:

Quadro 3.5.1 – Quest de Assassinato: Caso de Estudo

Nome	**Jogo**
Landing the Killing Blow	World of Warcraft
NPC Inicial	**NPC Final**
Dark Ranger Lyana	Dark Ranger Lyana
Nível Mínimo	**Nível da Quest**
68	71

Descrição

"You've proven yourself in the field of battle, <name>. You might very well have changed the tides in our favor.

I've an important assignment for you now. I'm not sure how much longer the weakness we've exposed is going to last, so we must use the element of surprise to our advantage.

I want you to use the confusion created by our attack and go behind enemy lines to assassinate their leaders. In order of rank, they are: Captain Olster, Lieutenant Celeyne and Sergeant Lorric. Good luck, <class>."

Objetivo

1. Matar Captain Olster.
2. Matar Celeyne.
3. Matar Lorric.

Recompensas

1. À escolha do jogador:
    a. *Bramblethorn Greatstaff*
    b. *Coldstone Cutlass*
    c. *Cragthumper*
    d. *Earthspike*
    e. *Elekk-Horn Crossbow*
    f. *Pacifying Pummeler*
2. 9 moedas de ouro.
3. 40 moedas de prata.
4. 25,150 pontos de experiência.
5. 350 pontos de reputação com *The Hand of Vengeance*.

Parâmetros

Iniciais	**Menor Dificuldade**	**Maior Dificuldade**
1. NPCs Principais (Captain Olster, Celeyne, Lorric)	--	--
a. HP	↓	↑
b. Reativo: Ataques (hostil)	--	--
c. Localização	--	--
2. NPCs Auxiliares	--	--
a. Quantidade	↓	↑
b. NPCs	↔	↔

A *quest* "Landing the Killing Blow" apresentada acima é um exemplo típico de *quest* de assassinato. Neste exemplo, é solicitado ao jogador matar os três líderes de um exército inimigo. Como estes NPCs fazem parte do enredo do jogo, apenas uma instancia de cada um pode existir em dado momento.

A solução empregada por grande parte dos MMORPGs para lidar com esse problema de instanciação é definir um período fixo de "respawn" após o qual os NPCs assassinados são reinseridos ao jogo, o que permite a múltiplos jogadores efetuarem a *quest* em sequencia ou, em casos específicos, como parte de um grupo.

Quadro 3.5.2 – Quest de Assassinato: Parametrização

Qtd.	Parâmetro	Atributo	Tipo	Descrição
[1, ∞[	NPC alvo	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física dos NPCs.
		Frequência	Numérico	Frequência de aparecimento dos NPCs.
		Conhecida	Booleano	Indica se a localização é fornecida ao jogador.
[1, ∞[	NPC Auxiliar	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física dos NPCs.
		Frequência	Numérico	Frequência de aparecimento dos NPCs.
		Conhecida	Booleano	Indica se a localização é fornecida ao jogador.
[0, ∞[	NPC Colateral	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física.
		Frequência	Numérico	Frequência de aparecimento.
		Quantidade Máxima	Numérico	Quantidade máxima de NPCs em um dado momento.
[0, 1]	Tempo	--	Numérico	Tempo limite.
1	NPC Final	Localização	Numérico	Localização do NPC.

A principal diferença entre os parâmetros identificados para as *quests* de Extermínio e aqueles identificados para as *quests* de Assassinato está no NPCs auxiliares. De modo a aumentar a dificuldade deste tipo de *quest*, NPCs ou monstros auxiliares podem surgir juntamente com o NPC único. O quadro 3.5.2 põe em evidencia essa diferença.

3.6. Quest de Escolta

Quests de Escolta consistem em proteger um NPC que se desloca entre duas localizações diferentes no mapa, sendo uma categoria bastante expressiva em jogos RPG online. O NPC escoltado muitas vezes interage diretamente com o ambiente, atacando inimigos pelo caminho, ou interrompendo seu progresso quando atacado. Cabe ao jogador fazer com que os obstáculos sejam removidos e o NPC chegue vivo ao seu destino.

O quadro 3.6.1 apresenta a *quest* “Bitter Departure” do jogo *World of Warcraft*, que demonstra traços típicos de uma *quest* de Escolta.

Nessa *quest*, após o diálogo inicial com o NPC “Injured Goblin Miner”, este se locomove a velocidade constante por diversos pontos do mapa (“checkpoints”) até chegar à saída da caverna onde se encontra. O trajeto escolhido por este NPC é fixo e passa por locais onde há presença de monstros não necessariamente atrelados à *quest*.

Quadro 3.6.1 – Quest de Escolta: Caso de Estudo

Nome	**Jogo**	
Bitter Departure	World of Warcraft	
NPC Inicial	**NPC Final**	
Injured Goblin Miner	Ricket	
Nível Mínimo	**Nível da Quest**	
77	78	
Descrição		
“Oi - thank you! I thought for sure I'd just be a spider snack. Perhaps I'm speaking a bit soon though - let's get the heck out of here! Let me know as soon as you're ready to go... make it quick.”		
Objetivo		
1. Escoltar o NPC *Injured Goblin Miner* para fora da caverna *Crystalweb*. 2. Falar com o NPC *Ricket* em K3.		
Recompensas		
1. À escolha do jogador: a. *Snowblind Butcher* b. *Garm's Ward* c. *Daschal's Serrated Blade* d. *Twisted Reflection* 2. 6 moedas de ouro 3. 80 moedas de prata		
Parâmetros		
Iniciais	**Menor Dificuldade**	**Maior Dificuldade**
1. NPC	--	--
a. Nome: Injured Goblin Miner	--	--
b. HP: 9,103 - 9,416	↑	↓
c. Reativo: Ataques	--	--
2. Percurso: Direto	↓	↑
3. Velocidade: Normal	↑	↓
4. Ponto de Partida	--	--
a. Crystalweb Cavern	--	--
5. Ponto de Chegada	--	--
a. Storm Peaks (K3)	--	--
6. Inimigos	--	--
a. Icetip Crawler	↓	↑
b. Quantidade	↓	↑

A grande dificuldade em adaptar o nível de dificuldade deste tipo de *quest* é que, similarmente às *quests* de Coleta (Cenário), o desafio é definido por elementos já presentes no ambiente virtual do jogo, neste caso, os NPCs/monstros que habitam

naturalmente a caverna. Apesar disto, é possível ajustar tanto os atributos e vulnerabilidades do NPC escoltado quanto o trajeto escolhido por ele. Especialmente neste último, alterando-se o trajeto para passar por regiões protegidas por monstros de mais alto nível, o impacto desta modificação no mundo virtual é mínimo.

Similarmente às *quests* de assassinato, é possível definir NPCs defensores que acompanham o progresso do NPC escoltado mas não são essenciais para a completude da *quest*, servindo apenas como meio de facilitar a tarefa do jogador.

Encontram-se resumidos abaixo os parâmetros deste tipo de *quest*.

Quadro 3.6.2 – Quest de Escolta: Parametrização

Qtd.	Parâmetro	Atributo	Tipo	Descrição
[1, ∞[	NPC Escoltado	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Velocidade	Numérico	Velocidade de locomoção do NPC.
		Reativo	Booleano	Indica se o NPC reage a ataques.
[0, ∞[	NPC Defensor	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
1	Percurso	Caminho	Coleção	Pontos que constituem o caminho seguido pelo NPC.
[0, ∞[	NPC Atacante	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física do NPC.
	Quantidade	--	Numérico	Quantidade de NPCs atacantes.
[0, ∞[	NPC Colateral	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física.
		Frequência	Numérico	Frequência de aparecimento.
		Quantidade Máxima	Numérico	Quantidade máxima de NPCs em um dado momento.
[0, 1]	Tempo	--	Numérico	Tempo limite.
1	NPC Final	Localização	Numérico	Localização do NPC.

3.7. Quest de Exploração

São consideradas *quests* de Exploração as *quests* nas quais o jogador não interage diretamente com o ambiente. Em MMORPGs, estas *quests* muitas vezes servem como elemento de ligação entre regiões do jogo, ou como forma de estimular os jogadores a conhecerem melhor o cenário desenvolvido. Para completar uma *quest* desse tipo, o personagem jogador deve passar por uma série de pontos no mapa, o que na maioria dos casos tem por efeito expandir o mapa do jogo.

Quadro 3.7.1 – Quest de Exploração: Caso de Estudo

Nome	**Jogo**	
Explore Alterac Mountains	World of Warcraft	
NPC Inicial	**NPC Final**	
-	-	
Nível Mínimo	**Nível da Quest**	
-	-	
Descrição		
-		
Objetivo		
1. Explorar as *Alterac Mountains*, revelando as areas cobertas do mapa mundo.		
Recompensas		
1. Achivement: Explore Alterac Mountains 2. 10 *achivement points*.		
Parâmetros		
Iniciais	**Menor Dificuldade**	**Maior Dificuldade**
1. Pontos a serem explorados (Localização):	--	--
a. Chillwind Point	--	--
b. Corrahn's Dagger	--	--
c. Crushridge Hold	--	--
d. Dalaran	--	--
e. Dandred's Fold	--	--
f. Gallows' Corner	--	--
g. Gavin's Naze	--	--
h. Growless Cave	--	--
i. Lordamere Internment Camp	--	--
j. Misty Shore	--	--
k. Ruins of Alterac	--	--
l. Sofera's Naze	--	--
m. Strahnbrad	--	--
n. The Headland	--	--
o. The Uplands	--	--

No exemplo acima o personagem jogador deve passar por 15 pontos de localização para completar a *quest*. Neste caso em específico, trata-se de uma *quest* do tipo *Achivement*, na qual não há dialogo com NPCs. Novamente, é possível alterar a dificuldade da *quest* com a adição de elementos tais quais um limite de tempo e NPCs Colaterais, conforme ilustrado no quadro 3.7.2.

Quadro 3.7.2 – Quest de Exploração: Parametrização

Qtd.	Parâmetro	Atributo	Tipo	Descrição
[1, ∞[	Localização	Localização	Numérico	Localização a ser explorada.
		Conhecida	Booleano	Indica se a localização é fornecida ao jogador.
[0, ∞[	NPC Colateral	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física.
		Frequência	Numérico	Frequência de aparecimento.
		Quantidade Máxima	Numérico	Quantidade máxima de NPCs em um dado momento.
[0, 1]	Tempo	--	Numérico	Tempo limite.
[0, 1]	NPC Final	Localização	Numérico	Localização do NPC.

3.8. Quest de Interação (NPC)

Quests de Interação são subdivididas em três categorias segundo o alvo de sua interação, seja um NPC, um objeto de cenário, ou um item.

A primeira categoria, *quests* de interação (NPC), agrega *quests* nas quais o único objetivo estipulado ao jogador é dialogar com um ou mais NPCs alvo. MMORPGs utilizam este tipo de *quest* como elemento de ligação entre subquests para formar sequencias coerentes comumente denominadas de “quest chains”. Uma instância característica de *Quest de Interação (NPC)* é a *quest* “Inform Leroxx!” do jogo *World of Warcraft* reproduzida abaixo.

Quadro 3.8.1 – Quest de Interação (NPC): Caso de Estudo

Nome	**Jogo**	
Inform Leoroxx!	World of Warcraft	
NPC Inicial	**NPC Final**	
Spiritcaller Dohgar	Leoroxx	
Nível Mínimo	**Nível da Quest**	
65	68	
Descrição		
“Quickly, you must inform Leoroxx of what is transpiring up at Razaan's Landing! Those orbs were comprised of souls, some of them Mok'Nathal!”		
Objetivo		
1. Falar com Leoroxx.		
Recompensas		
1. 1.200 pontos de experiência.		
Parâmetros		
Iniciais	**Menor Dificuldade**	**Maior Dificuldade**
1. Localização	--	--
a. Mok'Nathal Village.	↔	↔

Os parâmetros observados para este tipo de *quest* são apenas o NPC alvo da interação, e possivelmente o item necessário para inicialização do evento. NPCs Colaterais e um tempo limite podem, assim como nas *quests* citadas anteriormente, alterar a dificuldade sem modificar a essência da *quest*.

O quadro 3.8.2, apresentado a seguir, põe em evidência os parâmetros identificados.

Quadro 3.8.2 – Quest de Interação (NPC): Parametrização

Qtd.	Parâmetro	Atributo	Tipo	Descrição
[1, ∞[	NPC alvo	--	Classe	NPC alvo.
	Localização	Localização	Numérico	Localização a ser explorada.
		Conhecida	Booleano	Indica se a localização é fornecida ao jogador.
[0, 1]	Item	--	Classe	Item necessário para interação.
		Consumível	Booleano	Indica se o item é consumido após uso.
[0, ∞[	NPC Colateral	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física.
		Frequência	Numérico	Frequência de aparecimento.
		Quantidade Máxima	Numérico	Quantidade máxima de NPCs em um dado momento.
[0, 1]	Tempo	--	Numérico	Tempo limite.
[0, 1]	NPC Final	Localização	Numérico	Localização do NPC.

3.9. Quest de Interação (Cenário)

Nas *quests* de interação (cenário), o jogador deve interagir com certo número de objetos de cenário, os quais podem requerer um item para ativação. Exemplos deste tipo de *quest* são utilizar uma chave para abrir uma porta, deslocar rochedos de um caminho bloqueado, ou ainda ativar pilares arcanos.

O exemplo descrito no quadro 3.9.1 é uma *quest* composta das categorias Coleta (Cenário) e Interação (Cenário).

Quadro 3.9.1 – Quest de Interação (Cenário): Caso de Estudo

Nome	**Jogo**
Let the Fires Come!	World of Warcraft
NPC Inicial	**NPC Final**
Mensagem	--
Nível Mínimo	**Nível da Quest**
1	--
Descrição	
"You must help us! The Headless Horseman will attack the village at any moment! Fires will be everywhere! The children aren't safe! Please, <name>, will you face the fires of the Headless Horseman? When he comes, you must join the fire brigade. Grab a bucket and toss it on the fires, or pass it to a friend who's closer to the blaze. If you put out all the fires then we may yet be saved!"	
Objetivo	
1. Coletar baldes com água (fonte). 2. Usar baldes nas chamas.	
Recompensas	
1. 9 moedas de ouro. 2. 40 moeads de prata. *3.* 1 *Tricky Treat* 4. 1 *Weighted Jack-o'-Lantern*	

Parâmetros		
Iniciais	**Menor Dificuldade**	**Maior Dificuldade**
1. Localização (Baldes)	↔	↔
2. Chamas	↔	↔
a. Quantidade	↓	↑
b. Tamanho	↓	↑
c. Localização	↔	↔

Os parâmetros envolvidos em uma Quest de Interação (Cenário) são o objeto de cenário em si, o item necessário para ativação e sua localização no mundo do jogo. O objeto de cenário pode ser ativável, requerendo simplesmente uma interação direta por parte do jogador, ou não-ativável, devendo ser atacado e destruído para completar a quest. Neste caso, o atributo resistência indica a vulnerabilidade do objeto.

O quadro 3.9.2 apresenta em resumo os parâmetros desta categoria.

Quadro 3.9.2 – Quest de Interação (Cenário): Parametrização

Qtd.	Parâmetro	Atributo	Tipo	Descrição
[1, ∞[	Objeto de Cenário	Resistência	Numérico	Resistência
		Ativável	Booleano	Se verdadeiro, o jogador pode ativar diretamente o objeto.
		Consumível	Booleano	Indica se o objeto é consumido após ativação.
	Item	--	Classe	Item necessário para ativar o objeto.
		Consumível	Booleano	Indica se o item é consumido após uso.
	Localização	Spawn Point	Numérico	Localização física dos objetos.
		Frequência	Numérico	Frequência de aparecimento dos objetos.
		Quantidade Máxima	Numérico	Quantidade máxima de objetos em um dado momento.
		Conhecida	Booleano	Indica se a localização é fornecida ao jogador.
[0, ∞[	NPC Colateral	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física.
		Frequência	Numérico	Frequência de aparecimento.
		Quantidade Máxima	Numérico	Quantidade máxima de NPCs em um dado momento.
[0, 1]	Tempo	--	Numérico	Tempo limite.
1	NPC Final	Localização	Numérico	Localização do NPC.

3.10. Quest de Interação (Item)

As *quests* de Interação (Item) completam a classe de Quest de Interação. Estas *quests* correspondem às *quests* de criação de item (*crafting*) presentes em grande parte dos MMORPGs, e são caracterizadas por produzirem diretamente um item com base em uma “receita”.

Os parâmetros típicos deste tipo de *quest* são o item base (um ou mais) caracterizado pelo item em si, pela quantidade necessária, e por uma indicação de se este é consumido no processo de criação; e pelo item produzido. É importante notar que não é necessária a produção de um item para que a *quest* seja considerada uma *quest* de Interação (Item). O simples fato de obter todos os elementos necessários pode ser o objetivo final da quest.

Diferente das *quests* tratadas anteriormente, as *quests* desta categoria não admitem a utilização de NPCs Colaterais como elemento adicional de balanceamento, visto que não há interação direta do jogador com o mundo do jogo.

Estes fatos encontram-se representados no quadro a seguir:

Quadro 3.10.1 – Quest de Interação (Item): Parametrização

Qtd.	Parâmetro	Atributo	Tipo	Descrição
[1, ∞[	Item Base	--	Classe	Item necessário para ativar o objeto.
		Consumível	Booleano	Indica se o item é consumido após uso.
		Quantidade	Numérico	Quantidade produzida.
[0, ∞[	Item Produzido	--	Classe	Item produzido após uso do item base.
		Quantidade	Numérico	Quantidade produzida.
[0, 1]	Tempo	--	Numérico	Tempo limite.
[0, 1]	NPC Final	Localização	Numérico	Localização do NPC.

3.11. Quest de Entrega

As *quests* de entrega são uma variação das *quests* de interação (NPC), na qual a interação é apenas parte das condições necessárias para completar a quest. Os mesmos parâmetros podem então serem observados, com a exceção do parâmetro *Item*, uma vez que a interação com o NPC neste caso é feita de forma direta.

Para completar uma *quest* de entrega, o jogador deve possuir os itens requeridos pelos NPCs alvo em seu inventário, podendo estes ser resultado de outra *quest* ou obtidos diretamente do NPC Inicial.

O quadro abaixo apresenta o conjunto de parâmetros identificados para esta categoria de *quests*.

Quadro 3.11.1 – Quest de Entrega: Parametrização

Qtd.	Parâmetro	Atributo	Tipo	Descrição
[1, ∞[	NPC alvo	--	Classe	NPC alvo.
	Localização	Localização	Numérico	Localização do NPC.
		Conhecida	Booleano	Indica se a localização é fornecida ao jogador.
[0, ∞[	NPC Colateral	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física.
		Frequência	Numérico	Frequência de aparecimento.
		Quantidade Máxima	Numérico	Quantidade máxima de NPCs em um dado momento.
[0, 1]	Tempo	--	Numérico	Tempo limite.
[0, 1]	NPC Final	Localização	Numérico	Localização do NPC.

3.12. Quest de Defesa

Constituem a categoria *quests* de defesa missões nas quais o jogador deve defender uma área ou um conjunto de NPCs essenciais por um determinado tempo.

As *quests* desta categoria apresentam similaridades para com os membros da categoria Quest de Escolta, sem o deslocamento constante associado a este tipo de quest. Pelo contrário, muitas vezes o ambiente onde se passa a *quest* é um espaço distinto do resto do mundo, podendo inclusive ser instanciado exclusivamente para cada jogador ou grupo de jogadores participantes.

Os inimigos a serem enfrentados são então adicionados ao conflito em ondas (ou "waves"), possuindo uma frequência de aparecimento específica e atacando o jogador, os NPCs essenciais ao sucesso da *quest*, e seus possíveis defensores. Caso não haja NPCs essenciais, tal como na *quest* apresentada a seguir, a sobrevivência do personagem jogador pelo período estipulado de defesa constitui a condição de completude da quest.

Quadro 3.12.1 – Quest de Defesa: Caso de Estudo

Nome	**Jogo**	
Defend Grom'gol	World of Warcraft	
NPC Inicial	**NPC Final**	
Commander Aggro'gosh	Commander Aggro'gosh	
Nível Mínimo	**Nível da Quest**	
85	85	
Descrição		
"I hope you like to fight, <name>. Your troll friend has been through here and warned us that the Gurubashi were on the rampage. I never thought the battle would be this glorious though. Come, join in our fight and let's show these Gurubashi trolls the might of the Horde!"		
Objetivo		
1. Defender Grom'gol Base Camp por 3 minutos.		
Recompensas		
1. 16 moedas de ouro. 2. 54 moedas de prata. 3. 69,400 pontos de experiência.		
Parâmetros		
Iniciais	**Menor Dificuldade**	**Maior Dificuldade**
3. NPCs	--	--
a. Quantidade: ?	↑	↓
b. HP: ?	↑	↓
c. Reativo: Ataques (hostil)	hostil	passivos
d. Essenciais: Não	--	Sim
4. Ondas	--	--
a. Quantidade	↓	↑
b. Intervalo	↑	↓
c. NPCs	↔	↔

No quasro 3.12.1 se encontra descrita a *quest* “Defend Grom’gol” do jogo *World of Warcraft*, na qual o jogador deve sobreviver aos ataques de um grupo de inimigos ao vilarejo Grom’gol. Trata-se de um exemplo de *quest* de defesa em que não há NPCs essenciais cuja sobrevivência deve ser garantida pelo jogador.

O quadro a seguir evidencia os parâmetros identificados para esta categoria.

Quadro 3.12.2 – Quest de Defesa: Parametrização

Qtd.	Parâmetro	Atributo	Tipo	Descrição
[0, ∞[	NPC Essencial	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
[0, ∞[	NPC Defensor	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
[1, ∞[	NPC Atacante	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física do NPC.
	Tempo	--	Numérico	Tempo de espera antes do aparecimento dos NPCs.
	Quantidade	--	Numérico	Quantidade de NPCs do mesmo tipo.
[0, ∞[	NPC Colateral	Resistência	Numérico	Indicativo da resistência do NPC a ataques.
		Dano	Numérico	Indicativo do poder do NPC.
		Skills	Coleção	Habilidades especiais do NPC.
		Agressivo	Booleano	Indica se o NPC pode iniciar situações de combate.
		Reativo	Booleano	Indica se o NPC reage a ataques.
	Localização	Spawn Point	Numérico	Localização física.
		Frequência	Numérico	Frequência de aparecimento.
		Quantidade Máxima	Numérico	Quantidade máxima de NPCs em um dado momento.
1	Tempo	--	Numérico	Tempo total da quest.
[0,1]	NPC Final	Localização	Numérico	Localização do NPC.

3.13. Quest Composta

Muitas vezes as *quests* encontradas em MMORPGs não se conformam a nenhuma das categorias apresentadas anteriormente, mas ainda assim podem ser representadas como uma composição de parâmetros de múltiplas categorias. A categoria Quest Composta agrega *quests* com essa propriedade.

Um exemplo típico é a *quest* “An Unusual Patron”, descrita no quadro 3.21, composta por uma *quest* de Interação (Cenário) e uma *quest* de Saque. O jogador deve primeiro utilizar o item recebido do NPC inicial em um objeto de cenário, para em seguida recuperar outro item de um NPC agressivo e entregá-lo ao NPC final.

Quadro 3.13.1 - Quest Composta: Caso de Estudo

<table>
<tr><th>Nome</th><th>Jogo</th></tr>
<tr><td>An Unusual Patron</td><td>World of Warcraft</td></tr>
<tr><th>NPC Inicial</th><th>NPC Final</th></tr>
<tr><td>Nemeth Hawkeye</td><td>Nemeth Hawkeye</td></tr>
<tr><th>Nível Mínimo</th><th>Nível da Quest</th></tr>
<tr><td>24</td><td>26</td></tr>
<tr><th colspan="2">Descrição</th></tr>
<tr><td colspan="2">“So this Naias is the mastermind behind the Bloodscalp and their exotic magic? I'll admit, this is the last place I expected my inquiry to lead, but no matter, our findings may still prove useful.

Off the coast to the northwest is an island inhabited by lesser water elementals. There's a crude stone slab there that might serve as an altar for the likes of the trolls. Take the totem you found to that altar. Let's see if we can draw this Naias out and distill his power into a more agreeable form.!”</td></tr>
<tr><th colspan="2">Objetivo</th></tr>
<tr><td colspan="2">1. Usar o item Gift of Naias no Altar of Naias para invocar Naias.
2. Vencer Naias e coletar o item Heart of Naias.
3. Entregar o item Heart of Naias à Nemeth Hawkeye, localizado em Grom'gol Base Camp.</td></tr>
<tr><th colspan="2">Recompensas</th></tr>
<tr><td colspan="2">1. À escolha do jogador:
a. Junglewalker Boots
b. Monnions of Raw Power
c. Blade of Exotic Magic
2. 2650 pontos de experiência
3. 250 pontos de reputação</td></tr>
<tr><th colspan="2">Parâmetros</th></tr>
<tr><td colspan="2"><table>
<tr><th>Iniciais</th><th>Menor Dificuldade</th><th>Maior Dificuldade</th></tr>
<tr><td>5. Localização (Altar of Naias)</td><td>--</td><td>--</td></tr>
<tr><td>6. NPC</td><td>--</td><td>--</td></tr>
<tr><td>c. Nome: Naias</td><td>--</td><td>--</td></tr>
<tr><td>d. HP: 9,103 - 9,416</td><td>↑</td><td>↓</td></tr>
<tr><td>e. Reativo: Ataques</td><td>↔</td><td>↔</td></tr>
</table></td></tr>
</table>

Quests compostas apresentam, portanto, apenas um NPC inicial, um NPC final, e um conjunto de recompensas, mas podem incluir um número qualquer de etapas ou sub-quests que devem ser efetuadas em sequencia. O quadro a seguir apresenta a parametrização deste tipo de quest.

Quadro 3.13.2 - Quest Composta: Parametrização

Qtd.	Parâmetro	Atributo	Tipo	Descrição
[1, ∞[	Sub-quest	-	Classe	Compreende os parâmetros de outra categoria de quest, com a exceção do parâmetro NPC Final.
1	Tempo	--	Numérico	Tempo total da quest.
[0,1]	NPC Final	Localização	Numérico	Localização do NPC.

3.14. Quest Única

Pode ocorrer de uma *quest* não se enquadrar em nenhuma das categorias citadas, e não poder ser decomposta em sub-quests, como é o caso das *quests* compostas, por ter sido elaborada utilizando recursos de implementação não disponíveis ao *quest designer*. Este é o caso das *Scripted Quests* do jogo *World of Warcraft*; *quests* para comemorações como Natal ou Páscoa, e *quests* inspiradas em outros jogos, tal como a *quest* “Lawn of the Dead” - que tem por base o jogo Plants Vs. Zombies.

É possível também que uma *quest* se enquadre na classificação proposta, mas apresente elementos únicos que alterem o comportamento do jogador e influenciem no nível de dificuldade geral. No jogo *World of Warcraft,* trata-se das *Vehicle Quests*, ou *quests* de veículos, nas quais alguma etapa da *quest* deve ser efetuada com o personagem jogador sobre um animal ou dentro de um equipamento que possui controles e movimentação distintos do usual para o jogador. Neste caso, a dificuldade é em grande parte devida a não familiaridade do jogador com o veículo em questão. Este tipo de *quest* pode, entretanto, ser alvo do processo de balanceamento tal qual como a *quest* não única que possui como base.

3.15. Síntese do Capítulo

Sobre a classificação proposta, pode ser observado um padrão recorrente do uso de parâmetros e atributos, o que sugere que apenas alguns elementos base são rearranjados para compor as diferentes categorias de *quest*. Isto fica evidente observando-se o Quadro 3.15.1 - Parâmetros , que apresenta os parâmetros identificados para cada categoria de *quest*.

Unidade Atômica de Quest

NPC	Objeto de Cenário	Item	Quantidade	Tempo	Localização

Figura 2 - Unidade Atômica de Quest

Fica claro que apenas os elementos NPC, Localização, Item, Objeto de Cenário, Tempo, e Quantidade são suficientes para descrever qualquer categoria da classificação apresentada. Essa característica é importante, pois facilita a representação de *quests* e simplifica eventuais mecanismos de balanceamento

dinâmico, permitindo a definição de uma unidade atômica para o vetor de parâmetros observáveis, como pode ser visto na figura 2.

Quadro 3.15.1 - Parâmetros de Quest por Categoria

		Quest												
Parâmetro		**Extermínio**	**Assassinato**	**Saque**	**Coleta (Cenário)**	**Exploração**	**Interação (Item)**	**Interação (NPC)**	**Entrega**	**Defesa**	**Escolta**	**Interação (Cenário)**	**Composta**	**Única**
NPC Colateral	**NPC**	×	×	×	×	×		×	×	×	×	×	?	?
	Localização	×	×	×	×	×		×	×	×	×	×	?	?
NPC Atacante	**NPC**		×							×	×		?	?
	Localização		×							×	×		?	?
	Tempo									×			?	?
	Quantidade		×							×	×		?	?
NPC Defensor	**NPC**									×	×		?	?
	Localização									×	×		?	?
NPC Alvo	**NPC**	×	×	×				×	×	×	×		?	?
	Quantidade	×		×									?	?
	Localização	×	×	×				×	×	×	×		?	?
	Item			×				×					?	?
Objeto de Cenário	**Objeto**				×							×	?	?
	Quantidade				×							×	?	?
	Localização				×							×	?	?
	Item				×							×	?	?
Tempo	**-**	×	×	×	×	×	×	×	×	×	×	×	?	?
Percurso	**-**										×		?	?
Item Base	**-**			×	×		×	×				×	?	?
Item Produzido	**-**						×						?	?
Localização	**-**					×							?	?
NPC Final	**-**	×	×	×	×	×	×	×	×	×	×	×	?	?

4. Conflitos de Balanceamento

Neste capítulo são tratados os problemas decorrentes de conflitos de balanceamento ocasionados pela variação dos parâmetros identificados no capítulo anterior. Para cada um dos elementos fundamentais de uma *quest* – NPC, Item, Localização e Objeto de Cenário – são descritos os conflitos identificáveis e possíveis soluções para resolvê-los.

Como será visto, a solução empregada em cada caso depende muitas vezes da implementação do jogo em questão e do processo de balanceamento utilizado, de forma que múltiplas soluções são discutidas para cada problema.

4.1. Conflitos de NPC

O principal fator de risco para a arquitetura de *quests* proposta são os conflitos de NPC, sendo este o elemento representativo de grande parte das categorias. A seguir são apresentados os problemas decorrentes dos dois tipos possíveis de conflitos de NPC, assim como soluções viáveis.

4.1.1. Conflitos Intra-Quest

Um dos principais pontos de conflito de balanceamento devido à participação simultânea de múltiplos jogadores em uma mesma *quest* ocorre quando personagens jogadores diferentes atacam (ou são atacados) por um NPC balanceado para outro jogador. Pela classificação desenvolvida, três tipos de NPC são definidos: NPCs alvos, NPCs colaterais, e NPC final. O conflito ocorre para os dois primeiros, apenas em situações de combate, tal como nas *quests* de Saque, Assassinato, ou Extermínio, e pode ser resolvido de várias formas de acordo com as necessidades do jogo:

1. A primeira alternativa consiste em manter conjuntos de NPCs balanceados para cada jogador, efetivamente trocando os parâmetros definidos no processo de balanceamento segundo o personagem jogador com o qual o NPC está interagindo. Esta solução pode ser utilizada em grande parte dos jogos atuais, pois o mecanismo de detecção de interação, ou *Target Lock* (do inglês, fixação

de alvo), é uma característica do gênero. Por outro lado, o mecanismo proposto não leva em consideração a possibilidade de vários jogadores interagirem com o mesmo NPC simultaneamente, o que pode ser problemático nos jogos em que este tipo de interação é permitido, como *World of Warcraft*. O diagrama abaixo ilustra este método:

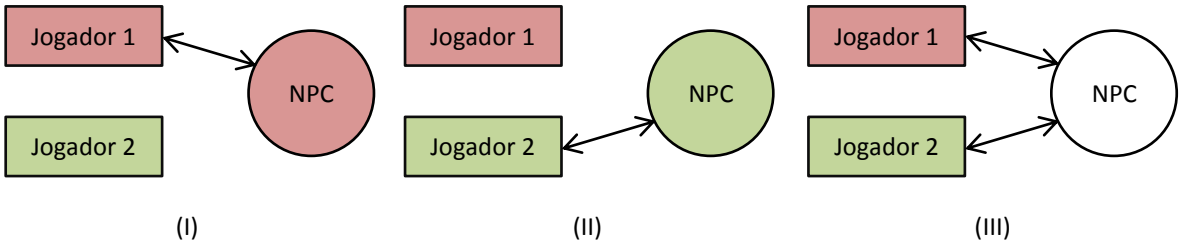

Figura 3 - Conflito de NPC: solução inicial

(I) Apenas o Jogador 1 interage com o NPC. (II) Apenas o Jogador 2 interage com o NPC. (III) Ambos os jogadores interagem simultaneamente com o NPC. Este caso não é permitido.

2. Uma extensão do método acima para resolver o problema da interação simultânea atribui ao NPC os parâmetros de um dos jogadores segundo um critério arbitrário, tal como o jogador mais fraco, aquele que interagiu primeiro, ou o alvo de interação atual do NPC. Independente da heurística empregada, este mecanismo apresenta brechas que podem ser exploradas pelos jogadores de modo a burlar o sistema de balanceamento. Um exemplo evidente deste problema pode ser observado com o uso da heurística simples "*O jogador que primeiro interage com o NPC define seus parâmetros*". Caso os jogadores estejam colaborando para a resolução da *quest*, é mais vantajoso que esta primeira interação seja efetuada pelo jogador mais fraco, o que efetivamente diminui o nível de dificuldade para o jogador mais experiente, que pode facilmente matar o NPC. Este problema é ilustrado a seguir:

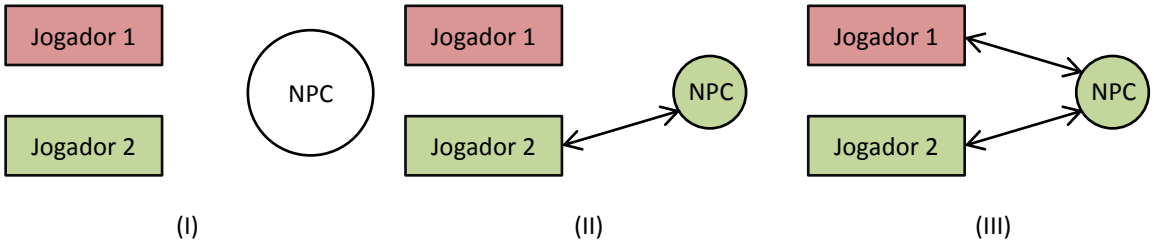

Figura 4 - Conflito de NPC: solução heurística

(I) Antes da interação. (II) O Jogador 2 interage com o NPC. (III) Ambos os jogadores interagem simultaneamente com o NPC, que se encontra balanceado para o Jogador 2.

3. A terceira alternativa consiste em manter instâncias distintas do NPC balanceadas para cada jogador, sendo a interação NPC→Jogador resolvida de forma independente, e a interação Jogador→NPC propagada a todas as instâncias de NPC de forma proporcional. Para efeito de clarificação, considera-se o exemplo abaixo com dois jogadores, para os quais o mecanismo de balanceamento determinou duas instâncias de NPC: uma instância verde, e uma instância vermelha (cinco vezes mais forte). Caso o **Jogador 2**, atacando sua instância do NPC (conforme item (II) da ilustração abaixo), cause dano **x**, o NPC virtual vermelho receberá uma quantidade de dano igual à **5x**, proporcionalmente. Para o caso inverso do **Jogador 1** estar atacando, o NPC verde receberá cinco vezes menos dano.

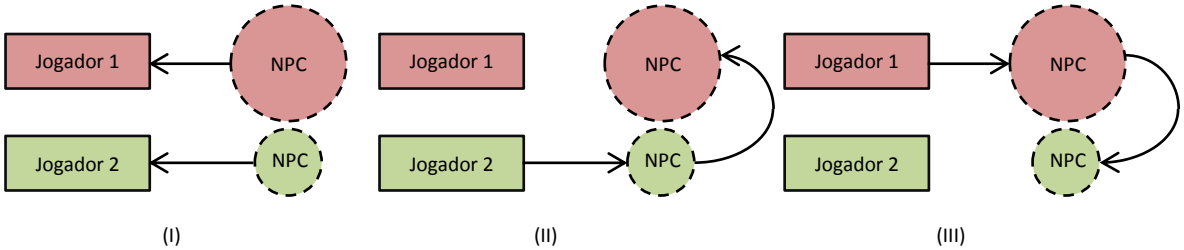

Figura 5 - Conflito de NPC: solução proporcional

(I) O NPC interage com o Jogador 1 e o Jogador 2. (II) O Jogador 2 interage com o NPC. O resultado da interação é refletido proporcionalmente sobre o NPC vermelho. (III) O Jogador 1 interage com o NPC. O resultado da interação é refletido proporcionalmente sobre o NPC verde.

A vantagem deste método é que os conflitos de interação simultânea são resolvidos de forma segura e não violável desde que o mecanismo de balanceamento funcione corretamente, mas apesar de bem estruturada, esta solução provê a ilusão de que ambos os jogadores possuem o mesmo nível de experiência (uma vez que o dano causado é proporcional), o que pode desestimular o jogador mais experiente.

4.1.2. Conflitos Inter-Quest

Conflitos de NPC inter-quests ocorrem quando um jogador interage com um NPC posicionado ou balanceado para outro jogador sem estar participando da quest. Visto que o jogador não está efetuando a *quest*, não existe um conjunto de parâmetros a ele associado que reflita o seu nível de experiência, tornando impossível o uso dos mecanismos de resolução de conflitos intra-quests

apresentados acima. Faz-se, portanto, necessário elaborar novas estratégias de resolução, dentre as quais merecem destaque as enumeradas a seguir:

1. Uma abordagem imediatamente aparente consiste em utilizar o mecanismo de balanceamento de *quests* para criar uma instância balanceada da *quest*, da qual é possível extrair os parâmetros do NPC em questão. Assim feito, torna-se possível utilizar os métodos de resolução de conflitos intra-quest no contexto inter-quest.

   Este método é dependente do mecanismo de balanceamento empregado, que considera a *quest* como um todo e não apenas o NPC desejado, o que não garante um resultado consistente com o nível do jogador. Pode ocorrer, por exemplo, de o NPC balanceado possuir um nível de dificuldade baixo devido a outro elemento da quest. Um exemplo concreto deste problema, no qual se deseja obter os parâmetros adaptados do **NPC 2** pelo balanceamento da *quest* a qual pertence, encontra-se ilustrado no quadro a seguir:

Quadro 4.1.1 - Conflito de NPC: Reuso de Blanceamento

Parâmetro	Dificuldade padrão	Dificuldade pós-balanceamento
NPC 1	5	7
NPC 2	**5**	**2**
NPC 3	5	6

2. Outra alternativa simples é utilizar os parâmetros tais como definidos pela *quest* para seu caso base ou na definição do NPC. Desta forma, apesar do NPC não ser adaptado ao jogador, não há variações de dificuldade entre diferentes instancias do mesmo. É comum no gênero MMORPG a presença de NPCs com parâmetros elevados em áreas específicas do jogo para controlar a distribuição geográfica dos jogadores e, apesar de quebrar o paradigma de balanceamento, este método é uma extensão dessa ideia.

3. Uma terceira forma de tratar o problema consiste em impedir a interação entre o personagem jogador e NPCs de quests das quais não participa. Isto pode ser feito controlando sua visibilidade, bloqueando as ações de interação ou impedindo o acesso à região na qual é executada a quest. Esta solução é adequada apenas a jogos que já possuam mecanismos de oclusão de NPCs, ou

nos quais o combate transcorre em uma região isolada e particular. Exemplos típicos desta categoria são os jogos *Atlantica Online* e *Dofus*.

4.2. Conflitos de Item

Outro tipo de conflito que pode resultar em situações de desbalanceamento é o conflito de itens que, mesmo possuindo menor abrangência que o conflito de NPC, pode levar a situações em que um mesmo elemento encontra-se balanceado para jogadores diferentes. Nas subseções a seguir são evidenciados os casos em que estas situações ocorrem, assim como medidas possíveis para arcar com os problemas levantados.

4.2.1. Conflitos Intra-Quest

Conflitos de balanceamento de itens podem ocorrer quando um jogador compartilha um item de *quest* com outro jogador, estando ambos realizando a *quest* em questão. Nesta situação específica, surge o problema de identificação do item recebido como sendo um item de *quest* necessário ao segundo jogador, uma vez que seus parâmetros diferem do esperado pela quest. Há duas formas de resolver conflitos dessa natureza:

1. A primeira alternativa é impedir o compartilhamento de itens de *quest*, notadamente de itens cujos parâmetros podem ser alterados pelo processo de balanceamento (e não aqueles que devem ser obtidos pelo jogador como parte da *quest*). Jogos como *World of Warcraft*, *Fly for Fun* e *Ragnarok Online* possuem marcações específicas para identificar itens de *quest*, que não podem ser trocados ou descartados e muitas vezes não ocupam espaço no inventário do jogador, sendo este método apenas uma extensão dessa prática.

2. Uma segunda alternativa consiste em modificar os parâmetros do item segundo o NPC que o possui, de forma que ao receber um item, seus parâmetros são automaticamente ajustados para o novo jogador. Essa solução é indicada para os jogos nos quais a troca de itens de *quest* é permitida.

4.2.2. Conflitos Inter-Quest

Há ainda um segundo tipo de conflito de itens, decorrente da interação do jogador com um elemento de outra *quest*. Ocorre do jogador obter um item de *quest* de um objeto de cenário ou de um NPC associado a uma *quest* que não esteja realizando no momento, mas que passa a realizar após certo tempo. Neste caso, o jogador passa a necessitar do item recebido anteriormente para completar a *quest*, mas seus parâmetros podem não coincidir com os esperados pela *quest* balanceada.

A solução ideal é uma extensão do tópico 1 descrito acima, impedindo o recebimento de itens de *quest* das quais o jogador não participa no momento. A solução alternativa de modificar os parâmetros do item segundo a necessidade não é indicada por levar a uma perda de credibilidade e imersão no ambiente do jogo, uma vez que o jogador está ciente de que houve uma alteração nos itens que possui.

4.3. Conflitos de Localização

Conflitos de localização é o terceiro tipo de conflitos possível, e se referem especificamente ao elemento Localização. Trata-se do elemento com menor possibilidade de conflitos, uma vez que a variável indicadora de posicionamento é única para cada personagem jogador, independente de jogo ou implementação.

4.3.1. Conflitos Intra-Quest

Não há possibilidade de ocorrência de conflitos de localização intra-quest, pois este parâmetro é inerentemente único a cada jogador e não pode ser influenciado por outras *quests*. Os conflitos decorrentes de alterações na localização são conflitos indiretos de NPC e Objeto de Cenário, e devem ser resolvidos utilizando os métodos intra-quest desenvolvidos para tal.

4.3.2. Conflitos Inter-Quest

Similarmente, conflitos inter-quest devem ser resolvidos pela aplicação dos mecanismos inter-quest de NPC e Objeto de Cenário. A exceção é o problema de aceitação posterior de *quests*, para o qual o jogador atinge a localização antes de ter

inicializado a *quest*. Neste caso, para evitar conflitos de parametrização da localização conhecida para com a localização balanceada posteriormente pela *quest*, a solução é desconsiderar quaisquer todas as localizações previamente conhecidas após a aceitação da *quest*.

4.4. Conflitos de Objetos de Cenário

Os conflitos de balanceamento decorrentes de objetos de cenário são similares àqueles identificados para NPCs, visto que em ambos os casos o jogador interage com elementos do jogo posicionados no mundo virtual e compartilhados entre todos os participantes.

A diferença, entretanto, é que para os objetos de cenário, ocorre apenas a interação Jogador→Objeto de Cenário, e não em ambas as direções como no caso da interação com um NPC. Apesar disto, as mesmas soluções elaboradas para a resolução de conflitos de NPC apresentadas anteriormente podem ser utilizadas para resolver os problemas decorrentes do balanceamento de objetos de cenário.

4.5. Síntese do Capítulo

Neste capítulo, foram identificados os pontos de conflito possíveis do modelo de representação de *quests* elaborado no capítulo anterior, assim como eventuais métodos de resolução e mitigação dos problemas ocasionados por esses conflitos.

Trata-se de uma etapa essencial no processo de elaboração do mecanismo de balanceamento dinâmico em MMORPGs, como deixa claro Arey, D. et al. [14], quando definem como característica necessária de qualquer processo de balanceamento dinâmico que as alterações efetuadas ao jogo por tal processo sejam críveis para todos os jogadores envolvidos.

Os detalhes de implementação factual das soluções apresentadas dependem, conforme explicitado, do jogo ao qual se aplicam e do funcionamento do mecanismo de balanceamento utilizado.

5. DDA Quest Designer

Para validar o modelo de classificação e parametrização de *quests* apresentado nos capítulos anteriores, o editor e gerenciador de *quests* DDA Quest Designer (*Dynamic Difficulty Adjustment Quest Designer*) foi concebido e implementado. Os detalhes do sistema e sua implementação são descritos neste capítulo.

5.1. Conceitos Gerais

Editores de *quest* são ferramentas de software desenvolvidas com o propósito de facilitar o desenvolvimento de jogos do gênero RPG. Por não existir uma representação unificada do modelo de *quests*, a implementação de um editor adaptado a cada jogo faz parte do processo de elaboração. O resultado é uma ferramenta altamente especializada de difícil adaptação ao modelo proposto para balanceamento.

Desta forma, apresenta-se neste capítulo um *Editor de Quests* concebido com base no modelo paramétrico de *quests* elaborado nos capítulos anteriores, com os objetivos de facilitar a implementação de um mecanismo de balanceamento dinâmico a um MMORPG existente, e a integração com os parâmetros únicos de cada jogo.

5.2. Requisitos de Sistema

O sistema desenvolvido visa atender uma série de requisitos, que serão apresentados nas subseções a seguir.

5.2.1. Requisitos Não-Funcionais

Em termos de usabilidade, é essencial que o usuário possa customizar a ferramenta para o jogo com a qual será utilizada, visto que, apesar das *quests* utilizarem o modelo de classificação proposto nos capítulos anteriores, os atributos de cada parâmetro podem divergir de jogo para jogo. Um exemplo simples desta

ocorrência são os atributos “Resistência”, “Magia” e “Skills”, que podem ou não estar presentes no parâmetro NPC dependendo do jogo ao qual se aplicam.

No que diz respeito a requisitos de integração, optou-se por utilizar a linguagem XML para representação da base de *quests* criada com uso da ferramenta. Por ser uma linguagem de representação semi-estruturada, e possível de ser validada com um XML schema, esta escolha permite maior facilidade de integração com quaisquer jogos ou implementações de processos de balanceamento.

Os conjunto completo de requisitos que descrevem os aspectos não-funcionais do sistema podem ser observados no quadro a seguir.

Quadro 5.2.1 - Requisitos Não-Funcionais

Identificador	Categoria	Descrição
RNF/USA-01	Usabilidade	A interface do sistema deve ser simples de utilizar e similar à interface de ferramentas similares.
RNF/USA-02	Usabilidade	A customização da ferramenta para adequação com os parâmetros únicos de cada aplicação deve ser efetuada por meio da interface do sistema.
RNF/INT-02	Integração	A linguagem de representação de quests empregada pela ferramenta deve ser única, facilmente identificável e deve abstrair o conteúdo da quest do mecanismo de balanceamento.
RNF/INT-02	Integração	O arquivo XML exportado/importado pelo sistema deve ser validado pelo schema provido.

5.2.2. Requisitos Funcionais

A seguir são apresentados os requisitos funcionais identificados para a ferramenta. As funcionalidades principais consistem em criar uma base de quests exportável para um arquivo XML, e adicionar novas quests das diversas categorias apresentadas. Cada categoria possui, conforme visto nos capítulos anteriores, uma

série de parâmetros e atributos característicos, apresentados pela ferramenta de forma modular e expansível.

O quadro a seguir sumariza os principais requisitos que serviram como base no desenvolvimento do sistema.

Quadro 5.2.2 - Requisitos Funcionais

Número de Ordem	Definição	Descrição
RF-01	Criar Base de Quests	O usuário pode criar uma base de quests, acessível pelo gerenciador de quests. A nova base substitui a base atual.
RF-02	Salvar Base de Quests	O usuário pode exportar a base de dados em um arquivo no formato XML.
RF-03	Carregar Base de Quests	O usuário pode carregar um arquivo XML no schema da base de quests.
RF-04	Adicionar Quest	O usuário pode adicionar uma nova quest a uma base existente através do gerenciador de quests. Esta ação exibe o editor de quests.
RF-05	Remover Quest	O usuário pode remover uma quest presente na base de quests atual através do gerenciador de quests.
RF-06	Editar Quest	O usuário pode editar uma quest que tenha previamente adicionado à base de quests. Esta ação exibe o editor de quests.
RF-07	Filtrar Quests	O usuário pode filtrar as quests presentes no gerenciador de quests por categoria de classificação.
RF-08	Gerenciar Quests	O usuário pode retornar ao gerenciador de quests após finalizar a edição de uma quest pelo editor.
RF-09	Duplicar Parâmetros	O usuário pode duplicar os parâmetros de quest definidos para o nível de dificuldade padrão para o nível de dificuldade selecionado no momento.

5.3. Projeto

O projeto do *DDA Quest Designer* é totalmente amparado por tecnologias livres e de código aberto, com a finalidade de permitir uma maior adaptabilidade e portabilidade por parte dos usuários. Nesta seção, são descritas as tecnologias,

ambiente de desenvolvimento e arquitetura utilizados no processo de implementação.

5.3.1. Arquitetura

O sistema desenvolvido foi elaborado de forma a ser modular, expansível e de fácil integração com as diversas implementações de jogos existentes. Com este propósito optou-se pela utilização de uma arquitetura de software em camadas [29], que distingue a aplicação entre dados, domínio e apresentação.

A camada de apresentação é responsável pela interação entre o usuário e a interface do sistema, e sua complexidade depende exclusivamente do meio de representação escolhido, seja por linhas de comando, ou com o uso de interfaces gráficas complexas.

A camada de domínio de negócio trata das necessidades do domínio específico da aplicação, tais como manipulações dos dados obtidos da camada de dados, validação de informações decorrentes da camada de apresentação, e seleção da fonte de dados apropriada a cada situação.

Por fim, a camada de dados é responsável pela obtenção e armazenamento dos dados necessários pela aplicação, o que pode ser feito por meio de diferentes fontes, tais como bancos de dados, arquivos textuais ou memória temporária. A escolha da fonte deve ser abstraída por esta camada.

Para a ferramenta DDA Quest Designer, além da arquitetura em camadas, foi feito uso de padrões de projeto [30] que visam facilitar o desenvolvimento e manutenção do código. O quadro 5.3.1 resume os padrões empregados, com uma breve descrição de cada um.

Quadro 5.3.1 - Padrões de Projeto

Padrão	Categoria	Descrição
Singleton	Criação	Padrão que garante a existência de apenas uma instância de classe, com referência compartilhada entre toda a aplicação.
Composite	Estrutural	Padrão utilizado para representar objetos que incluem em sua declaração outros objetos de mesma hierarquia. Este padrão é utilizado principalmente em listas e interfaces gráficas.
Façade	Estrutural	Façade é um padrão de encapsulamento que consiste na utilização de uma classe com métodos simples com o propósito de abstrair chamadas diretas as classes inferiores.
Observer	Comportamental	Padrão que define um objeto Observer, o qual é notificado sempre que determinado evento ocorre na classe observada, que pode possuir um número qualquer de observadores, todos notificados quando necessário.
Template Method	Comportamental	Consiste na utilização de métodos abstratos, implementados por eventuais subclasses, de modo a garantir reuso de código e melhor estruturação de herança.
Iterator	Comportamental	Padrão que consiste na utilização de objetos agregadores denominados Iterators para navegação de uma coleção de dados de forma segura e padronizada.

5.3.2. Tecnologias

Para a implementação da ferramenta, foi utilizada a linguagem de programação JAVA; uma linguagem livre, amplamente utilizada, multiplataforma e apoiada por diversas ferramentas de código aberto. Em conjunto com o ambiente de desenvolvimento Eclipse, trata-se de uma das plataformas de desenvolvimento de ferramentas e aplicações de software mais utilizadas.

5.4. Interface do Usuário

Essa seção tem por objetivo apresentar os dois módulos de interface que compõem os sistema e ilustrar as principais funcionalidades de cada um: o gerenciador de *quests* (*Quest Manager*) e o editor de *quests* (*Quest Editor*).

A transição do gerenciador para o editor ocorre quando uma nova *quest* é adicionada ou uma *quest* existente é editada, enquanto que a transição do editor para o gerenciador é feita por meio da opção de menu "*File* > *Return to Manager*".

Ao retornar ao gerenciador, as alterações efetuadas pelo editor à *quest* são automaticamente atualizadas.

5.4.1. O Editor de Quests

O editor de *quests* é o módulo de interface responsável por permitir a inserção dos dados de uma *quest,* possuindo uma região de parâmetros comuns a todas as categorias de *quests*, e uma região dedicada aos parâmetros únicos de cada tipo. O editor apresenta ainda três abas superiores para navegar entre os níveis de dificuldade representados por cada conjunto de parâmetros. A figura abaixo ilustra o editor de *quests* configurado para inserção de uma *quest* de assassinato (*"Boss Quest"*).

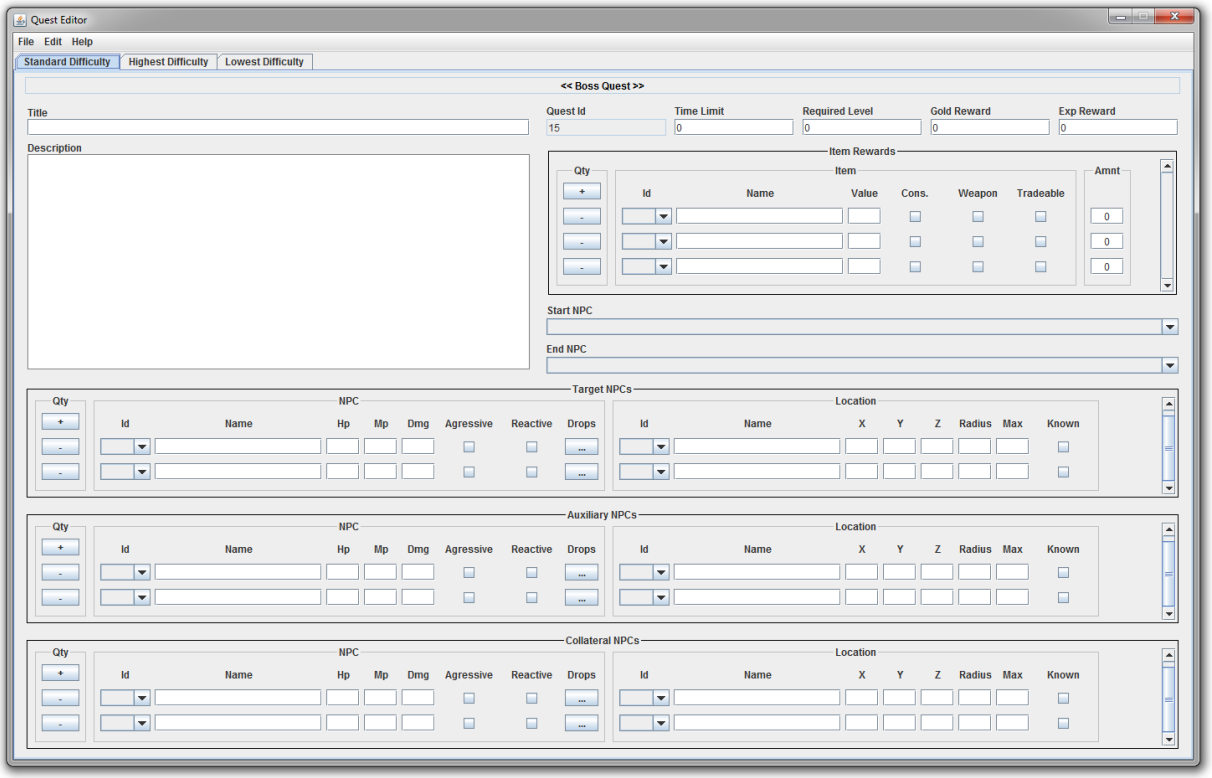

Figura 6 - Editor de Quests: Boss Quest

Como pode ser observado, os parâmetros únicos estão representados na metade inferior da janela, e podem possuir número variável de valores associados, conforme indicado pelo tipo de *quest* em questão. Para a *quest* de assassinato ("*Boss Quest*"), o número possível de NPCs alvo é ilimitado, podendo ser expandido ou reduzido com o uso dos botões no lado esquerdo da janela.

A figura a seguir apresenta o mesmo editor configurado para a inserção de uma *quest* composta, cujos parâmetros únicos são determinados pela natureza de suas sub-quests. Nota-se que é possível adicionar e remover sub-quests com o uso das abas *"New Quest +"* e *"[x]"* visíveis na parte superior do painel inferior.

Após preencher todos os valores de parâmetro, o usuário deve utilizar a opção de menu "*File* > *Return to Manager*" para retornar ao gerenciador de quests, módulo apresentado na subseção a seguir.

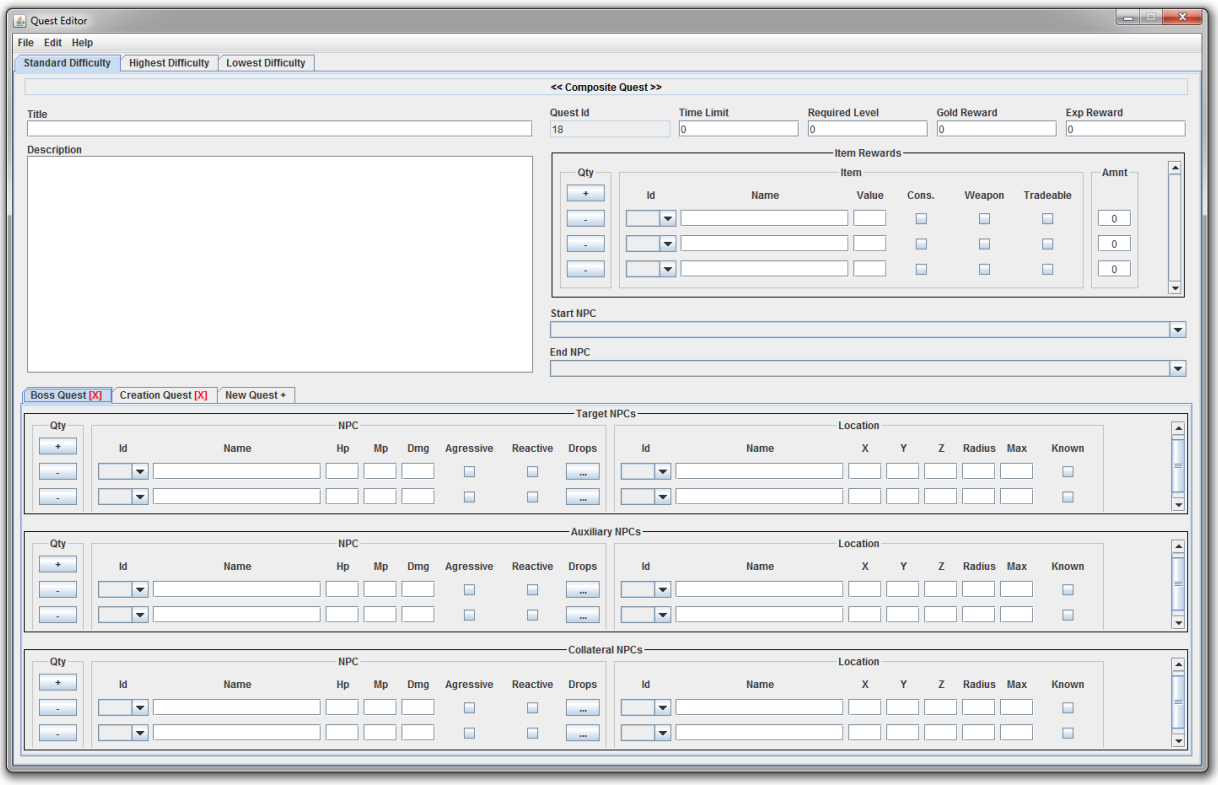

Figura 7 - Editor de Quests: Composite Quest

5.4.2. O Gerenciador de Quests

O gerenciador de *quests* é o segundo módulo de interface da ferramenta *DDA Quest Designer*, e possui a função de permitir a gestão das *quests* criadas com o uso do editor, assim como exportar e importar arquivos XML que representam a base de *quests*. A figura 11 apresenta a janela correspondente ao gerenciador.

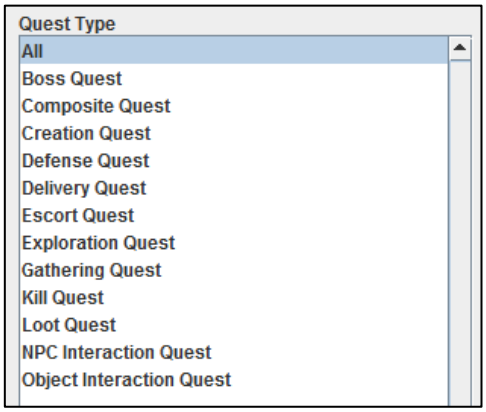

Figura 8 - Filtro de Quests

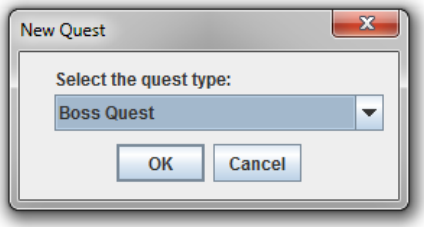

Figura 9 - Nova Quest

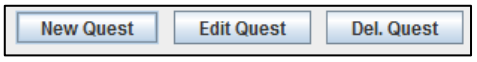

Figura 10 - Funções de Gerenciamento

Na janela apresentada pelo gerenciador, é possível filtrar a lista de quests da base por categoria com o uso da lista "*Quest Type*", posicionada no canto superior esquerdo (vide figura 8). A tabela no canto direito exibe o identificador, tipo, e título das *quests* filtradas, e permite a seleção de *quests* para edição ou remoção.

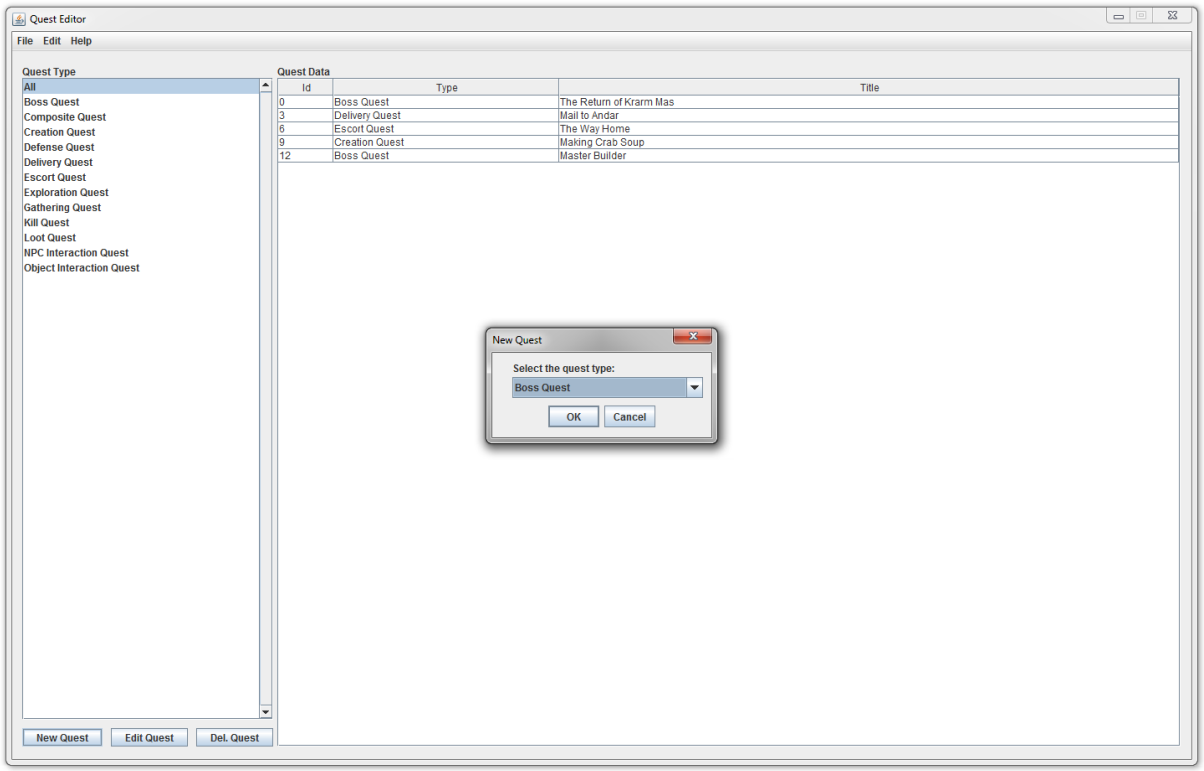

Figura 11 - Gerenciador de Quests

Para adicionar, remover ou editar uma *quest*, utilizam-se os botões no canto inferior esquerdo da janela, como pode ser visto em detalhes na figura 10. Ao clicar em "*New Quest*", uma janela é exibida para a escolha do tipo de *quest* a ser criada, seguido da exibição do editor de *quests*. A figura 9 apresenta essa janela em maiores detalhes.

5.5. Síntese do Capítulo

Em vista das funcionalidades atribuídas à ferramenta *DDA Quest Designer,* é possível afirmar que os objetivos propostos de permitir a criação e gerenciamento de *quests* no modelo proposto neste projeto foram atingidos de forma satisfatória.

O sistema desenvolvido foi concebido com uma arquitetura modular e em camadas que facilita a manutenção e expansões futuras, e a customização possível dos atributos de cada parâmetro de *quest*, por meio de arquivos XML, constitui a base para adaptar a ferramenta a qualquer jogo ou implementação.

6. Considerações Finais

Este capítulo relata as considerações finais relativas ao trabalho desenvolvido ao longo deste projeto. Nas seções a seguir, são apresentadas as contribuições esperadas deste processo, assim como as dificuldades e restrições encontradas, e possibilidades de expansão do trabalho efetuado em projetos futuros.

6.1. Contribuições

Este trabalho apresentou uma nova modelagem para *quests* em jogos massivos de interpretação de personagens, com uma abordagem especificamente voltada a processos de balanceamento dinâmico. Uma hierarquia de classificação de *quests* foi proposta, e um estudo paramétrico de *quests* presentes em jogos atuais efetuado para validar o modelo elaborado. Os eventuais problemas e conflitos decorrentes do uso de tal modelagem foram considerados, e diversas soluções abrangentes sugeridas. Desenvolveu-se ainda a ferramenta *DDA Quest Designer* para a construção e gestão de quests com a representação especificada em vias de reforço e validação do modelo proposto.

Trata-se de uma abordagem inovadora no campo de ajuste dinâmico de dificuldade em jogos, pois a arquitetura proposta permite o uso de sistemas de balanceamento dinâmico não apenas caso a caso, como nos estudos efetuados até o momento, mas considera a possibilidade de ajuste simultâneo para múltiplos jogadores no ambiente dinâmico e persistente dos MMORPGs.

6.2. Dificuldades Encontradas

Tendo em vista que MMORPGs compõem um gênero de jogos relativamente recente, e que a área de balanceamento dinâmico de dificuldade encontra-se em sua infância, a principal dificuldade encontrada no decorrer deste projeto foi a baixa incidência de material acadêmico que combinasse processos de ajuste dinâmico de dificuldade com elementos característicos de jogos de interpretação (RPG), ou mesmo qualquer subgênero de jogos online massivos.

Apesar disto, utilizando publicações do campo de geração procedural de quests – a exemplo do trabalho elaborado por Doran, J. et al. [22], e aplicações de balanceamento dinâmico em outros gêneros de jogos off-line, foi possível conceber a modelagem apresentada neste documento.

No que diz respeito à elaboração da ferramenta DDA Quest Designer, a principal dificuldade foi encontrar exemplos de sistemas com funcionalidade semelhante para análise e estudo. Apesar de certo que o desenvolvimento de uma ferramenta de edição de *quests* faz parte do processo de construção de um jogo, a mesma raramente é disponibilizada ao público, sendo utilizada internamente pela equipe de concepção de quest.

6.3. Trabalhos Futuros

Conforme indicado anteriormente em referência ao trabalho de Arey, D. et al. [14], para ser classificado como balanceamento dinâmico, o processo de concepção de uma arquitetura voltada ao balanceamento dinâmico envolve três etapas principais, enumeradas a seguir.

1. Determinação dos parâmetros de balanceamento.
2. Definição do mecanismo de identificação de desbalanceamento.
3. Definição do mecanismo de balanceamento.

Fica claro, portanto, que devido ao amplo escopo identificado, apenas a primeira etapa de identificação de parâmetros e modelagem estrutural de *quests* foi contemplada neste projeto. A prorrogação do mesmo para a elaboração dos mecanismos de balanceamento consiste então no próximo passo lógico, assim como o refinamento do modelo desenvolvido com os resultados de uma implementação prática.

Em termos do DDA Quest Designer, várias possibilidades de expansão podem ser propostas, como funcionalidades de customização de parâmetros – e não apenas de atributos, como é feito no momento, ou mesmo a integração com bases de dados de NPCs, itens e outros elementos de instâncias específicas de jogos.

7. Referências

1. HUNICKE, R.; CHAPMAN, V. AI fot Dynamic Difficulty Adjustment in Games. **Challenges in Game Artificial Intelligence AAAI Workshop. San Jose**, 2004.

2. ADAMS, E. The Designer's Notebook: Difficulty modes and Dynamic Difficulty Adjustment. **Gamasutra.com**, 2008.

3. RACHID, O. et al. Alternate Reality Gaming. **Comunications of the ACM**, Fevereiro 2008. 36-42.

4. DUCHENEAUT, N.; J. MOORE, R. The Social Side of Gaming: A Study of Interaction Patterns in a Massively Multiplayer Online Game. **Communications of the CSCW'04. Chicago, Illinois.**, 2004. 360-369.

5. STEINKUEHLER, C. A. Learning in Massively Multiplayer Online Games. **Communications of the CHI 2004. Vienna, Austria.**, 2004. 521-528.

6. ACHTERBOSH, L.; PIERCE, R.; SIMMONS, G. Massively multiplayer online role-playing games: The past, present, and future. **ACM Compt. Entertain., 5, 4, Article 9**, Março 2008. 1-33.

7. WU-CHANG, F.; BRANDT, D.; DEBANJAN, S. A Long-Trem Study of a Popular MMORPG. **Communications of NetGames'07. Melborne, Australia.**, 2007. 19-24.

8. MULLIGAN, J.; PATROVSKY, B. Developing Online Games: An Insider's Guide. **New Riders**, 2003. 474-477.

9. EVE Online - FAQ. **EVE Online**. Disponivel em: <http://www.eveonline.com/faq/faq_01.asp>. Acesso em: 2 dez. 2011.

10. VEHICLE Quests. **WoWWiki**. Disponivel em: <http://www.wowwiki.com/Vehicles_(mechanic)>. Acesso em: 2 dez. 2011.

11. RAPTORAK. MMORPG’s – Grinding vs Questing. **Teach me to Game**, 2011. Disponivel em: <http://www.teachmetogame.com/blog/mmorpgs-grinding-vs-questing/>. Acesso em: 2 dez. 2011.

12. PFEIFFER, B. AI to Control Pacing in Games. **Proceedings of the IC2 GameDev Workshop, University of Texas, Austin.**, 2003.

13. VAN LANKVELD, G.; SPRONCK, P.; RAUTERBERG, M. Difficulty Scaling Through Incongruity. **Proceedings of the Fourth Artificial Intelligence and Interactive Digital Entertainment Conference, Stanford, California, USA.**, 2008.

14. AREY, D.; WELLS, E. Balancing Act: The Art and Science of Dynamic Difficulty Adjustment. **2001 Game Developers Conference, San Jose**, 2001.

15. ANDRADE, G. R.; CORRUBLE, V. Automatic Computer Game Balancing: A Reinforcement Learning Approach. **Game Developer**, 2005.

16. DEMASI, ; CRUZ , A. Online Coevolution for Action Games. **Proceedings of The 3rd International Conference on Intelligent Games And Simulation. London**, 2002. 113-120.

17. VALVE CORPORATION. Left 4 Dead - The AI Director. **Left 4 Dead**, 2009. ISSN http://web.archive.org/web/20090327034239/http://www.l4d.com/info.html. Disponivel em: <Left 4 Dead>. Acesso em: 2 dez. 2011.

18. NINTENDO. Pick a Potent Power-Up. **Mario Kart 7**, 2011. Disponivel em: <http://mariokart7.nintendo.com/items/>. Acesso em: 2 dez. 2011.

19. PIGGYBACK. **Resident Evil 5 Official Strategy Guide**. [S.l.]: Prima Publishing, 2009.

20. BAILEY, C.; KATCHABAW, M. An Experimental Testbed to Enable Auto-dynamic Difficulty in Modern Video Games. **Department of Computer Science, University of Western Ontario, Ontario, Canada.**, 2005.

21. DORAN, J.; PARBERRY, I. Towards Procedural Quest Generation: A Structural Analysis of RPG Quests. **Technical Report LARC-2010-02, Laboratory for Recreational Computing, Dept. of Computer Science & Engineering, University of North Texas**, Maio 2010.

22. DORAN, J.; PARBERRY, I. A Prototype Quest Generator Based on a Structural Analysis of RPG Quests. **Proceedings of the Second International Workshop on Procedural Content Generation in Games. Bordeaux, France.**, 2011.

23. PITA, J.; MAGERKO, B.; BRODIE, S. True story: dynamically generated, contextually linked quests in persistent systems. **Future Play '07 Proceedings of the 2007 conference on Future Play, New York, USA**, 2007. 145-151.

24. WALKER, J. A network of quests in World of Warcraft. **Second Person: Role-Playing and Story in Games and Playable Media, eds. Pat Harrigan and Noah Wardrip-Fruin. Cambridge, MA, USA.**, 2007.

25. WOWWIKI. Disponivel em: <http://www.wowwiki.com/>. Acesso em: 2 dez. 2011.

26. WOWHEAD. Disponivel em: <http://www.wowhead.com>. Acesso em: 2 Dezembro 2011.

27. ALLAKHAZAM MMO Quest Databases. Disponivel em: <http://zam.com>. Acesso em: 2 dez. 2011.

28. MMODB Quest Databases. Disponivel em: <http://www.mmodb.com>. Acesso em: 2 dez. 2011.

29. CHIBA, C.; NARDI,. Desenvolvimento em Camadas. **MSDN**. ISSN http://www.microsoft.com/brasil/msdn/tecnologias/arquitetura/Layers_Developing.mspx. Acesso em: 2 dez. 2011.

30. GRAND, M. **Patterns in Java:** A Catalog of Reusable Design Patterns Illustrated with UML. 2a. ed. [S.l.]: Wiley, v. 1, 2002.

31. KIM, J. Y.; JONATHAN, P. A.; EDWARDS, R. Extending Cyberspace: Location Based Games Using Cellular Phones. **ACM Comput.Entertain., 4, 1, Article 3C**, Janeiro 2006. 1-18.

32. KUAN-TA, C. et al. Game Traffic Analysis: An MMORPG Perspective. **Communications of the NOSSDAV'05, Stevenson, Washington**, 2005. 19-24.

33. CORNETT, S. The Usability of Massively Multiplayer Online Games: Designing for New Users. **Communications of the CHI 2004. Vienna, Austria.**, 2004. 703-710.

34. FLEMING SEAY, A. et al. Project Massive: A Study of Online Gaming Communities. **Communications of the CHI 2004. Vienna, Austria.**, 2004. 1421-1424.

35. STUART, B. Virtual Worlds as a Medium for Advertising. **The DATA BASE for Advances in Information Systems., 38, 4**, Novembro 2007. 45-55.

36. GUO, Y.; BARNES, S. Why People Buy Virtual Items in Virtual Worlds with Real Money. **The DATA BASE for Avances in Information Systems., 38, 4**, Novembro 2007. 69-76.

37. APONTE, M.-V.; LEVIEUX, G.; NATKIN, S. Scaling the Level of Difficulty in Single Player Video Games. **Proceedings of the 8th International Conference on Entertainment Computing (IEC'09), Springer-Verlag, Berlin, Heidelberg.**

38. YUN, C. et al. PADS: enhancing gaming experience using profile-based adaptive difficulty system. **Sandbox '10 Proceedings of the 5th ACM SIGGRAPH Symposium on Video Games, New York, USA**, 2010.

39. KOSTER, R. **Theory of Fun for Game Design**. Phoenix: Paraglyph Press, 2004.

40. ADAMS, E. Balancing Games with Positive Feedback. **Gamasutra.com. Último acesso 08-10-2011.**, 2002.