

Produção Acadêmica de um Jogo Eletrônico Educativo

ANDRE LUIZ BATTAIOLA

LAI / albattaiola@ufpr.br

RAFAEL PEREIRA DUBIELA

LAI / rafaeldubiela@hotmail.com

FLÁVIO EDUARDO MARTINS

LAI / flavioemartins@yahoo.com.br

HENRI SIRO EVRARD

LAI / henri@ufpr.br

DANIEL AVILEZ

LAI / daniel.avilez@ufpr.br

VIVIANE GASPAR RIBAS

LAI / Viviane.gasparibas@ufpr.br

LAI-Laboratório de Animação Interativa

Departamento de Design – Universidade Federal do Paraná

Rua General Carneiro, 460, 8º andar, sala 814, Curitiba, Paraná

Resumo

Edugraph é um jogo Educativo em processo de implementação, voltado para o ensino de conceitos de computação gráfica. O roteiro elaborado para este jogo requer a geração de cenários e de um personagem. Este artigo descreve o processo criativo de elaboração destes elementos, bem como descreve os processos de modelagem, animação e sonorização utilizados para a geração do vídeo de abertura. Adicionalmente, o artigo descreve o processo de exportação dos elementos modelados no software Discreet 3D MAX para o ambiente do Macromedia Director viabilizando a publicação do jogo na Web através da tecnologia Shockwave 3D.

Palavras-chaves: Jogos de Computador, Modelagem, Animação, Sonorização.

1. Introdução

O *Edugraph* [i] é um jogo Educacional cuja função é ensinar alguns conceitos fundamentais de Computação Gráfica para alunos de Design e de Engenharia. Ele é composto por fases bi e tridimensionais implementadas na plataforma Microsoft Windows para publicação na Web através das tecnologias Macromedia Shockwave e Shockwave 3D. O *Edugraph* é parte do projeto LudicLearning, financiado pelo CNPq (Processo: 552173/2002-9).

Este artigo discute o processo de montagem, criação e de implementação da animação, que constitui a fase introdutória do sistema. Esta fase exhibe o Edu (Educativo), o personagem principal, caminhando. Em determinado momento, paredes aparecerão e irão aprisioná-lo. Surgem, em seguida, um

pedestal com controles, um portal com três encaixes e um conjunto de três esferas girando no teto. Edu utiliza o controle para mover as esferas e encaixá-las no portal, que se abre e transporta-o para um cubo. Durante este processo, Edu que originalmente tinha cinco cores, perde-as e se torna escuro. O cubo tem encaixes nas paredes e no teto, além de controles que permitem a movimentação de um objeto que entra nos encaixes e remete Edu a cinco mundos diferentes. Em cada mundo, dois com ambientes 2D e 3 com 3D, Edu é confrontado com problemas. Ao solucioná-los, Edu recupera as cores.

2. O Projeto de Desenvolvimento

O projeto de criação de um jogo de computador envolve uma série de etapas. Uma etapa importante é a criação da arte, isto é, a criação dos elementos que serão vistos,

ouvidos e até sentidos pelo usuário, de forma a viabilizar o nível preciso de percepção requerido pela ambientação, conforme definido pelo roteiro. A arte do jogo compatibiliza o que a programação permite fazer, com o que o roteiro requer que seja feito. Essa junção usualmente deve balancear desejos e recursos do usuário.

Os elementos que compõem a ambientação de um jogo são basicamente os personagens e os cenários, com suas respectivas animações. O processo de criação destes elementos demanda um planejamento preciso. O ponto-chave é compatibilizar requisitos vindos de roteiro (o que o roteiro estabelece) e de programação (o que o software permite que se implemente). Determinadas as restrições, pode-se, então, permitir que a criatividade flua.

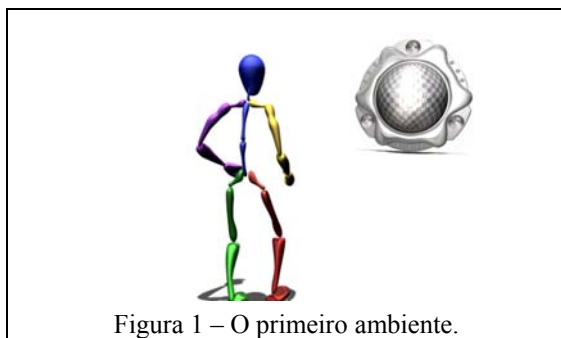


Figura 1 – O primeiro ambiente.

No projeto *Edugraph*, a escolha das cores e das formas para compor o personagem seguiu o princípio da simplicidade, isto é, cores fortes e claras, fáceis de serem notadas, e formas arredondadas e curvilíneas, buscando uma harmonia estética e uma fácil percepção. O princípio da percepção das formas e o estudo da percepção das cores são conceitos introduzidos de maneira didática pela escola alemã de *design*, a Bauhaus [ii], e aplicados tanto aos personagens quanto ao cenário.

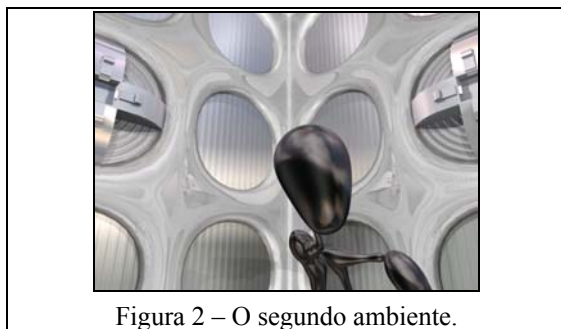


Figura 2 – O segundo ambiente.

O primeiro passo para dar alma ao personagem é atribuir-lhe um nome. Como ele é o protagonista do jogo *Edugraph*, um jogo Educacional, uma opção foi Educador, abreviado para Edu. O segundo passo é dar-lhe características humanas, como no caso do Edu, ser perspicaz, curioso, inteligente, corajoso, um pouco atrapalhado e um tanto desengonçado. Um bom herói para um jogo Educacional e interativo, o qual necessita da simpatia de seu protagonista.

Uma das características do jogo, estabelecida inicialmente, foi a da ambientação em um mundo sem grandes amarras com a realidade. Esta liberação incentiva o processo criativo e evita a busca pela perfeição estética do mundo real, uma meta difícil de ser alcançada com um nível de qualidade aceitável.

Neste contexto, decidi se que os objetos desse mundo fossem definidos por meio de formas e cores que acentuassem a mensagem a ser transmitida, bem como que fossem gerados e animados de forma eficiente em computador. Além disso, a estética estaria apoiada no fator lúdico. O resultado dessas premissas foi a idéia da criação de um mundo de fantasia, onde tudo é possível e permitido no contexto de se apresentar os conceitos da computação gráfica.

De acordo com esta concepção foram escolhidos dois ambientes iniciais. O primeiro (Figura 1), composto por um ambiente etéreo, totalmente branco com apenas comandos, paredes e um portal que oferecia um enigma a ser resolvido para passar para o outro próximo nível, que, por sua vez, seria composto por um ambiente fechado, o interior de um cubo, com comandos mais avançados e também tarefas a serem cumpridas utilizando os princípios aprendidos no nível anterior (Figura 2). Sempre buscando a associação entre entretenimento e ensino.

O ambiente etéreo, sem paredes e sem limites, todo branco, tem como função dar ao espectador a percepção de início de tudo, de que qualquer coisa é possível. Este ambiente associa conceito com facilidade de implementação. Adicionalmente, a idéia de um cenário fechado, opressor, que forçasse o Edu a querer sair dele usando os conceitos

aprendidos, sugeriu a criação de mais um ambiente no mesmo estilo, o cubo.

Visando direcionar a atenção do espectador para uma situação adequada ao aprendizado dos conceitos mais simples do universo 3D de forma lúdica e dinâmica, foi gerado um vídeo de animação, que contém os primeiros passos para a assimilação dos conceitos de rotação, translação e mudança de eixos do mundo 3D. Essa animação é composta pelo personagem e pelo primeiro cenário modelado.

O processo de criação dessa primeira animação obedeceu as etapas usuais de trabalho: o *story-board* (desenho prévio dos quadros da animação), a modelagem do personagem, a modelagem do cenário, a animação das câmeras, o teste de edição, o teste de sincronização, a análise de contexto, aplicação de luzes, a renderização¹ das cenas como um todo e a edição final das cenas com a inserção do som. Este procedimento permitiu que se criasse uma animação complexa em curto período de tempo.

3. Modelagem do Personagem

O primeiro passo para a criação de Edu foi o desenho em papel com, o esboço do personagem. O segundo foi os testes de cores com as imagens dos desenhos digitalizados (“escaneados”) no computador e, o último, foi um estudo das diferentes maneiras possíveis de se aplicar as ferramentas disponíveis para se completar a modelagem.

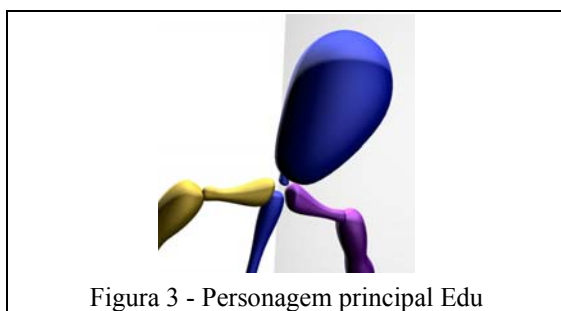


Figura 3 - Personagem principal Edu

Como dito anteriormente, Edu (Figura 3) é um personagem lúdico, irreal, caricato e, portanto, a única referência para a identificação entre o protagonista e o espectador era o fato dele possuir os mesmos membros de um ser humano, porém não necessariamente as mesmas formas. Dessa maneira ele possui cintura, tronco, braços, pernas, pescoço e cabeça. Considerou-se que as expressões faciais seriam desnecessárias e computacionalmente custosas. Assim, optou-se por uma representatividade mímica do personagem, sem associar a ele um rosto. Este procedimento também evitou a implementação de diálogos.

Todo procedimento de modelagem em 3D envolve uma série de ferramentas e conceitos, que podem ser basicamente resumidos dentro da geometria euclidiana e em seus mais básicos elementos, o ponto (ou vértice) a reta e o plano. Dentro da terminologia dos programas o ponto tem o nome de *vertex*, a reta de *spline*, *line* ou *edge* e o plano de *face* ou *polygon*, dependendo do modificador que esta sendo utilizado. Assim como na geometria euclidiana, são necessários no mínimo três pontos para se formar um plano, na terminologia 3D cada face é formada por três vértices. O conjunto de faces define uma malha (*mesh*) e uma malha pode ser a parte de um objeto ou até mesmo um objeto inteiro. O que dá o aspecto final na renderização é a junção dessa geometria com um mapa, imagens e texturas que revestem as malhas geométricas.

A modelagem de Edu usou operações booleanas [iii] de sólidos (união, subtração e intersecção) e primitivas simples (esferas e cubos), com posterior ajuste de vértices e faces diretamente na malha 3D. O arredondamento das faces foi obtido através do uso da técnica de *meshsmooth* [iv], procedimento que aumenta o número de faces do objeto a partir da média entre as normais² das próprias faces, o que resulta em um arredondamento do objeto. Em suma, realizou-se um *Box-modeling*, que consiste na modelagem de

¹ *Rendering* ou finalização, é o processo onde se juntam a geometria, o mapa, as luzes e a animação de forma a mostrar a imagem finalizada na tela. Quando uma cena é renderizada, ela é finalizada em um arquivo de saída que não pode mais ser alterado em 3D, pode somente ser renderizado novamente.

² A normal é uma reta perpendicular situada no meio da face que aponta para fora dela, recurso utilizado para definir os lados de dentro e de fora de uma face, e portanto de um objeto. É uma reta de referência, portanto, normalmente não aparece na tela.

malhas com baixo número de polígonos, isto é, com poucas faces e vértices, seguida da aplicação do modificador *meshsmooth*.

4. Animação do Personagem

A animação do personagem Edu foi dividida em duas etapas. Na primeira foram feitas as animações utilizadas no vídeo de abertura. Posteriormente, gerou-se as animações que foram inseridas no jogo. A diferença entre as etapas é basicamente uma restrição de implementação, pois, enquanto as animações para o vídeo de abertura possuem apenas as restrições impostas pelo programa, as animações feitas para o jogo possuem várias restrições técnicas. Por outro lado, ambas as animações estão relacionadas ao mesmo personagem, portanto, devem apresentar a mesma linguagem em qualquer condição.

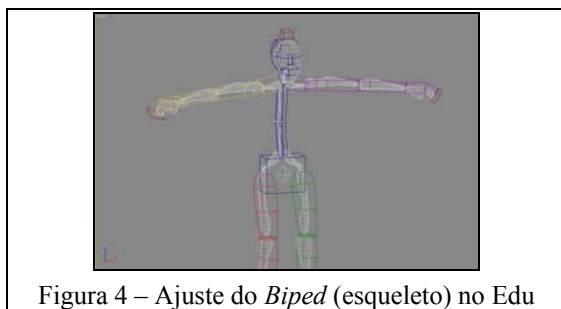


Figura 4 – Ajuste do *Biped* (esqueleto) no Edu

Como o personagem apresenta características humanas e é formado por membros similares ao de uma pessoa, ele deveria assumir também posturas de gente. Para isso foi usada uma técnica muito comum em 3D que é a inserção de um esqueleto de animação dentro do modelo do personagem Edu. Esse esqueleto tem como função servir como referência de deformação do modelo, isto é, fazer com que os vértices dele sigam a posição exata dos ossos (*bones*) do esqueleto (Figura 4). Esse procedimento no programa *3D MAX* tem uma nomenclatura própria. O esqueleto utilizado é o *biped* [v] e a interação do esqueleto com o modelo é feito por uma ferramenta chamada *physique*, que abre a possibilidade de se editar envelopes que atuam como ímãs sobre os vértices fazendo com que os mesmos sigam o *bone* ao qual ele atua (Figura 5).

A diferença entre a técnica de animação do vídeo inicial e das animações do jogo são

as especificações das ferramentas, pois o que muda são formalidades de programação e de programas, porém o conceito é o mesmo, ou seja, os *bones* vão sempre puxar os vértices da malha (Figura 6).

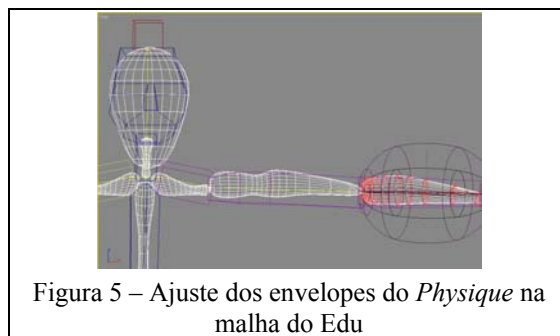


Figura 5 – Ajuste dos envelopes do *Physique* na malha do Edu

Basta, portanto, animar os *bones*. No caso do vídeo, os *bones* estão disponíveis no *biped* e prontos para uso, basta aplica-los. No jogo eles são feitos para ser um recurso similar, só que montados um a um. A ferramenta que transporta os vértices junto ao *bone* no vídeo chama-se *physique* e no jogo é o *skin* [vi]. Substancialmente o que muda entre um e outro, é que por se tratar de um recurso que já acompanha o programa, as ferramentas *biped* e *physique*, são mais ágeis e práticas de usar, pois já vêm com muitos ajustes feitos. Por outro lado os *bones* criados para o jogo e o *skin* devem ser ajustados um a um e são, sem dúvida, menos versáteis.

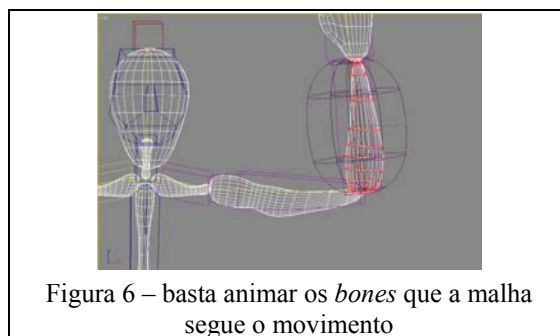


Figura 6 – basta animar os *bones* que a malha segue o movimento

5. Modelagem do Cenário

Os procedimentos para a modelagem dos cenários não diferenciam muito dos procedimentos utilizados na modelagem do personagem. Primeiramente foram elaborados os esboços, com base nos conceitos já criados pelo roteiro. Em seguida foi feito o estudo de cores e só então se iniciou a modelagem definitiva dos cenários. Estes procedimentos

servem para minimizar as possibilidades de erro e tentar validar o trabalho em apenas uma tentativa.

Os cenários deveriam funcionar como um recurso secundário. Assim, a opção foi destacar o Edu no ambiente em termos de cores. Para tanto, o ambiente foi criado utilizando majoritariamente tons monocromáticos do azul e cinza (Figura 7).

Em uma situação como essa, em que se fazem exatamente essas escolhas, a linha que limita um bom e um mau trabalho é muito tênue, porque, se não houver o cuidado prévio, a tendência é fazer um cenário pobre, logo cabe ao modelador saber dosar as intenções com o trabalho final esperado. Nos cenários foi adotado o princípio de aplicar sobre os objetos 3D materiais monocromáticos, porém que apresentassem um resultado estético sofisticado através do uso de espelhamento.

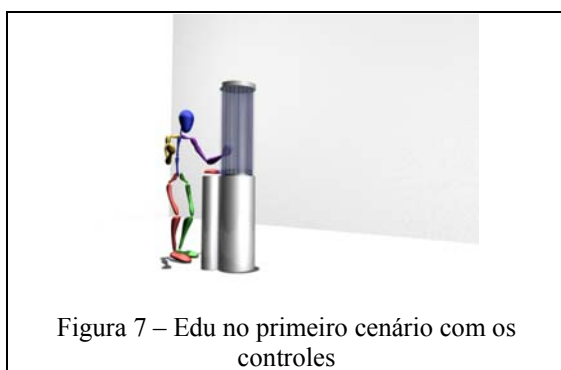


Figura 7 – Edu no primeiro cenário com os controles

Tecnicamente, compor os ambientes também não foi complexo, pois a técnica de modelagem para o primeiro ambiente foi basicamente a criação de sólidos primitivos como esferas, cilindros, etc. Tendo apenas um objeto, o portal, requerendo uma modelagem mais aprimorada, na qual foi utilizada a mesma técnica para se modelar o personagem, a técnica de *Box-modeling*. A composição do segundo ambiente não trás nada de novo ao que já foi exemplificado como técnicas de modelagem. Porém, a grande diferenciação se dá com a utilização simples de recursos básicos aliados a um conceito criativo para compor o ambiente.

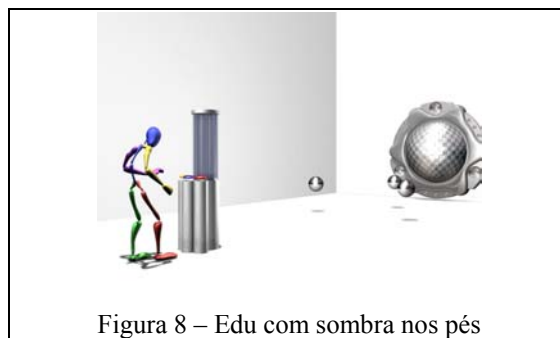


Figura 8 – Edu com sombra nos pés

As luzes que foram aplicadas aos dois ambientes tinham a obrigação de compor as idéias de cada um deles. Como o primeiro ambiente era formado por um universo etéreo, branco, foi dada uma especial importância para a visualização das sombras, em especial no chão, pois, se não fossem elas, haveria a idéia errônea de que Edu andava no espaço, sem chão (Figura 8). Portanto, não é exagero dizer que a noção de ambiente foi viabilizado inicialmente pelas sombras de Edu e, para que elas fossem perfeitamente compreendidas como tal, foi necessário utilizar um tipo especial de luz, gerada pelo *3D MAX*, chamada de *advanced raytrace* [vii]. No ambiente do cubo, a luz não possuía tanta importância, pois o único requisito era que o ambiente deveria ser composto de maneira a dar a impressão de ser bem iluminado e claro.



Figura 9 – paredes do cubo com *raytrace*

Um outro aspecto considerado na composição dos ambientes foi a aplicação dos materiais aos modelos. No primeiro ambiente, os materiais utilizados foram apenas aplicações de cores. Porém, em três objetos que compunham as cenas, foi utilizado um material reflexivo para adicionar um toque de sofisticação às cenas. Essa reflexão dos materiais foi obtida com a utilização de um tipo de recurso de material denominado de reflexão *raytrace* [viii]. No segundo ambiente, o *raytrace* foi utilizado em todas as paredes do

cubo (Figura 9), dando um efeito extraordinário de reflexão infinita, como quando são colocados dois espelhos um em frente ao outro.

7 Diferenças de produção de um vídeo e um jogo

Existem muitas diferenças entre produção de um vídeo de animação 3D e na produção de um jogo. O princípio de criação de ambas e o raciocínio lógico e intuitivo são completamente distintos. Em geral, isso é um problema significativo, por exemplo, no caso do projeto Edugraph, já que tanto um quanto o outro devem formar um projeto único, com uma só linguagem. Isso deve ser destacado por que foram tomados, no mínimo, dois caminhos para cada questão, seja em termos de modelagem, animação, iluminação, câmera, etc.

A principal diferença técnica entre um vídeo de animação 3D e um jogo 3D é que enquanto um trabalha com renderização em tempo real, o outro trabalha com cenas pré-renderizadas. Isso é uma grande restrição para o jogo, pois a maioria dos recursos que são utilizados nas animações, e que adicionam mais riqueza a ela, não podem ser utilizados, pois requerem um processamento pesado. Note-se que, como os quadros de uma animação podem ser gerados um a um e depois editados para formar um filme, cada quadro pode ser produzido por uma técnica específica, ao contrário dos jogos, que usualmente produzem animações seguindo métodos específicos.

Em um jogo, o número de polígonos que compõem os seus objetos determina a sua performance, logo, às vezes, se vai ao extremo de se utilizar apenas o número suficiente de polígonos para tornar o perfil de um objeto próximo daquilo que deveria ser.

Em termos de materiais, as restrições principais são no tamanho das texturas em *pixels*. Enquanto na animação o tamanho é restrito apenas ao processamento do micro, no jogo essa restrição é mais séria, fazendo com que os mapas sejam do limite do tamanho entre o que pode e o que se quer de resolução mínima. Existe também uma série de recursos que os jogos, em geral, ainda não aceitam e uma série de recursos que dependem da

potência da placa de vídeo do micro do espectador. Por exemplo, o recurso de *bump-mapping*³ é disponibilizado por pouquíssimas placas gráficas e, pelo método escolhido para se implementar o Edugraph, não foi possível de ser utilizado. O recurso de *raytrace* também ainda não foi implementado em nenhuma placa ou utilizado em qualquer jogo.

As animações para um vídeo são mais livres e lineares, devendo apenas seguir uma rotina de causa e efeito, sem necessariamente serem cíclicas. Existe a possibilidade de fazer o protagonista andar uma longa distância sem que um passo seja igual ao outro. Já no jogo os mecanismos existentes não permitem essas possibilidades, afinal as animações são cíclicas e a movimentação do protagonista no cenário é feita por programação. Logo, a criatividade deve ser confinada a essas restrições. No caso do protagonista andar uma longa distância, seguramente, em um jogo, ele repetiria o ciclo de andar algumas vezes.

Mas nem tudo na produção para jogo é só restrição. As câmeras utilizadas em um jogo são, em sua maioria, livres, isto é, o espectador pode observar o mesmo cenário em diferentes ângulos. Isso incrementa a qualidade interativa do jogo.

8. Exportação do 3D Studio para o Plasma e do Plasma para o Director

Como em outras tarefas do projeto, o processo para montar a arte do jogo Edugraph seguiu etapas bem definidas. A primeira foi a concepção 3D no programa *3D MAX*, e a segunda foi exportar os modelos para o *Plasma*. A exportação exigiu ajustes e, em alguns casos, houve a necessidade de se remodelar o objeto. Após estas duas etapas, os modelos foram exportados para a extensão *Shockwave* e importados definitivamente no *Director*, prontos para inserção no jogo.

Todas essas etapas apresentaram peculiaridades que merecem destaque. Por exemplo, ao se exportar os modelos de *3D MAX* para o *Plasma*, os modelos devem ser exportados com extensão “.3DS”. O primeiro

³ Bump-mapping é um recurso de mapeamento utilizado para fazer com que a textura aplicada sobre uma malha apresente um aspecto áspero. Em geral ele funciona bem com a aplicação de um mapa de textura para auxiliar sua melhor visualização.

inconveniente é que esse processo não exporta os quadro-chaves (*keyframes*) da animação e, portanto, todas as animações devem ser refeitas no *Plasma*. A vantagem dessa operação é que, como ambos os programas foram desenvolvidos pela mesma empresa, suas interfaces são muito parecidas, o que simplifica o trabalho. Também não é possível se confiar plenamente no mapeamento exportado no formato “.3DS”. Portanto, se o modelo for corretamente importado pelo *Plasma*, é aconselhável que o mapeamento seja feito por garantia. Dependendo do volume de trabalho, esse erro de mapeamento só será notado quando o jogo for rodado.

O *Plasma* possui muitas ferramentas para a criação e a edição de modelos. Portanto, sempre que um modelo apresentar problemas, é aconselhável refazê-lo no próprio *Plasma*. No caso do *Edugraph*, por exemplo, as paredes do cubo e o modelo do cubo-chave foram feitos inteiramente no *Plasma*. Só foram usados seus respectivos modelos no *3D MAX* para renderizar as suas texturas com efeitos de *bump-mapping* e *raytracing*, efeitos que como dito anteriormente não são exportáveis pelo *Plasma*. Já os modelos do controle e do Edu, que são mais complexos, foram importados diretamente do *3D MAX* e remapeados. As animações do personagem Edu foram analisadas à parte.

O processo de exportação do universo criado no *Plasma*^{ix} para o ambiente *Director* é relativamente simples (Figura 10). Se o objeto não possui animação própria, basta criar um controlador de colisão para *rigid body* [x] e seguir os passos de exportação contidos no *help* do programa que a operação é efetuada corretamente. No entanto, se o objeto possui uma animação própria, então existe a necessidade de se incluir o objeto junto com todos os componentes da possível animação em um grupo, e exportar apenas esse grupo.

Se existe a necessidade de exportar um personagem, e esse personagem possui animações de *bones* mais *skin*, o processo se complica. A construção dos *bones* requer cuidados e existem algumas regras que, se seguidas, diminuem a possibilidade de incompatibilidade na exportação. Estas regras são: 1- nunca mover objetos sem atentar precisamente para coordenadas (embora o

sistema de coordenadas do *Plasma* não seja o mais confiável, ele é mais preciso que o olho humano, portanto, todas as operações de *move* devem ser feitas numericamente). 2- No momento que se finaliza o sistema de *bones* de um dos lados do personagem, nunca copiar esses *bones* para o outro lado, mesmo usando *mirror*, esse procedimento não funciona. Cedo ou tarde, problemas surgirão, sendo o ideal refazer os *bones* e tentar aproximar o máximo possível às medidas de um lado para o outro. Tudo isso porque o sistema de coordenadas dos *bones* baseia-se no *pivot*⁴ do objeto, algo extremamente sensível e delicado, pois em geral essas coordenadas possuem precisão de até quatro casas decimais, o que as torna muito susceptíveis a variações indesejáveis.

Criado o sistema de *bones*, aplicar o *skin* é um processo que não requer nenhum cuidado adicional, sua implementação no *Plasma* pode ser feita de forma similar ao do *3D MAX*. As animações do personagem apresentam um bom resultado, porém sua exportação é um problema, pois a animação cíclica no *Director* tem um certo atraso entre o último e o primeiro frame, o que dá uma percepção de solavanco toda vez que ela se reinicializa. Uma maneira de se contornar esse problema é fazer ciclos mais longos que o comum, repetindo várias vezes o mesmo ciclo dentro dele mesmo. Dessa maneira na exportação o problema do atraso não é resolvido, porém ele se dilui na seqüência exportada.

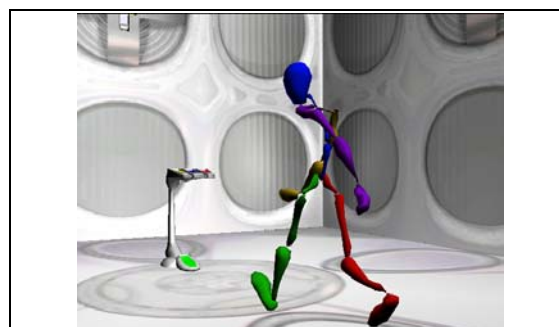


Figura 10 – Edu no jogo, imagem renderizada em tempo real

⁴ Todo o objeto no universo 3D possui coordenadas de posição no espaço cartesiano, nos eixos X, Y e Z. Esse posicionamento é feito baseado em um ponto para cada objeto, esse ponto em geral é o equivalente ao centro de massa do objeto e possui a nomenclatura de *pivot*.

Uma das soluções encontradas para a questão do salto entre os ciclos de animação é a citada acima. Outra é a tentativa de usar o recurso do Director para unificar animações, baseado em *keyframes* distintos, produzindo uma combinação entre as animações produzidas. Esse recurso não foi suficientemente pesquisado para ser possível dar um parecer conclusivo, ficando apenas o registro de sua existência.

Ressalte-se que muitos destes problemas foram detectados ao longo do processo de exportação para o Director e as soluções foram sendo determinadas a partir de pesquisas em várias fontes. Uma observação importante é que existe pouca informação disponível para pesquisa sobre estes problemas.

9. Gerenciamento do Processo

Os processos usados para executar tanto a animação quanto o jogo foram cada vez mais se tornando complexos. Mesmo seguindo uma concepção metodológica de implementação, houve situações isoladas em que trabalho precisou ser refeito.

Na animação inicial, as renderizações das cenas foram feitas em várias etapas, nas quais se respeitou o prazo de conclusão. A primeira parte foi renderizada totalmente duas vezes, sendo que ela é composta por quarenta e quatro cenas compostas por arquivos independentes. Na primeira renderização não foram utilizadas as luzes *advanced raytrace*, pois era preciso que todas as cenas fossem feitas o mais rapidamente possível, depois com mais tempo foram aplicadas as luzes certas e se renderizou as cenas novamente. No segundo ambiente, composto apenas por 4 cenas, houve apenas uma renderização total, porém como foram utilizados vários recursos que dificultam a renderização, essas quatro cenas foram as mais demoradas para ficarem prontas. Como previsto anteriormente no cronograma da animação, muitas cenas foram renderizadas isoladamente mais de uma vez, em alguns casos mais de quatro. Isso porque invariavelmente sobram detalhes não verificados que acabam dando resultados não esperados que, em sua maioria, não agradam em termos estéticos.

Como o jogo foi evoluindo por partes, a cada ponto concluído foi sendo organizado o

próximo. Dessa maneira, todas as etapas foram concluídas e todos os desafios vencidos.

10. Produção Sonora

Em um filme, o áudio é empregado para viabilizar a comunicação entre personagens, bem como aumentar o realismo de determinadas cenas e acentuar a carga emocional do filme como um todo.

Jogos também empregam o áudio para acentuar a transmissão de emoção para o jogador. Similarmente aos filmes, os jogos utilizam os seguintes tipos de áudio: vozes, ruídos, música e efeitos 3D. Vozes e ruídos são denominados usualmente de áudio digital, pois são usualmente armazenados no formato WAV (Microsoft *Windows Riff Wave*). Música pode ser armazenada no formato MIDI (*Musical Instrument Digital Interface*) porque ele é apropriado para armazenar música, gera arquivos pequenos em termos de espaço de armazenamento e permite que se programe procedimentos para a produção de sons. O áudio 3D é utilizado para simular efeitos de deslocamento (carro que se desloca da esquerda para a direita requer que o usuário ouça primeiramente o som no ouvido esquerdo), de reverberação (reflexão do som em objetos), de distância (sons produzidos por fontes próximas ao observador são ouvidos mais intensamente), e efeito Doppler (alteração da frequência do som em função de como a fonte se desloca em relação ao observador). Além destes elementos, jogos também utilizam sons reativos (aumento do som do motor de um carro em função de sua aceleração) e interativo (o som em uma sala aumenta à medida que um número maior de inimigos adentra).

Em sistemas de ensino multimídia, Mayer e Moreno ^{xi}examinaram a questão da integração de som, texto e animação e concluíram que a animação com associação de texto e som foi mais efetiva do que a animação com texto. Esta conclusão reforça a importância do áudio em jogos Educacionais.

A primeira parte do projeto a ser sonorizada foi o vídeo de apresentação do *Edugraph* porque este vídeo será a base para as opções sonoras que serão desenvolvidas no decorrer do jogo. A produção desta trilha sonora considerou os seguintes aspectos: a)

interação com o roteiro; b) os elementos da teoria musical e da arte composicional; c) a criação de efeitos sonoros e d) a tecnologia disponível para promover uma ótima interação entre os itens anteriores.

A primeira parte deste trabalho foi determinar as informações do roteiro relacionadas com os conceitos abstratos a serem transmitidos (sentimentos, comportamentos, expressões etc) no vídeo. Sabendo exatamente as sensações de cada uma das cenas pôde-se utilizar a produção sonora como um colaborador ativo e reforçador da mensagem intrínseca que o filme gostaria de transmitir. Trabalhou-se cada uma das cenas que possuía uma determinada intenção separadamente tanto nos efeitos quanto na música, remetendo o usuário a diferentes ambientes, cada qual com atmosfera e sons característicos. Do início do vídeo até o momento em que o Edu começa a ser enclausurado pelas paredes que surgem do chão, foi considerada a primeira parte. Nessa parte, a música tem somente uma linha de pensamento e desenvolvimento geral, apesar das nuances e dos elementos que vão se modificando. A segunda parte foi estipulada começando no final das paredes se fechando até o encaixe das esferas no portal. Este é um momento de plena atividade. Em função disto, trabalhou-se com uma rítmica mais veloz que da parte anterior e deixando em primeiro os efeitos sonoros das esferas que ocupam um papel crucial nesta segunda parte. O terceiro momento inicia-se logo após o encaixe das esferas. É o momento em que Edu vai ser teletransportado, o que requer um grande esforço na elaboração de efeitos que já foram suficientes para resolver o problema sonoro desta parte, sem necessidade de uma parte musical. A quarta seção inicia-se logo quando o personagem se vê dentro do Cubo. Toda a equipe trabalhou para que este momento tivesse uma carga dramática especial. A opção escolhida para sonorizá-lo foi a de sons mais artificiais e que trouxessem para a cena uma noção de espacialidade maior que nos outros momentos. Pode-se reparar que as músicas que foram produzidas para as partes 1 e 2 tinham um caráter mais comum, tanto na rítmica quanto nos timbres (qualidades de um som as quais permitem distinguir a sua fonte), porém,

a música da parte 4 soa estranha caso escutada separada do vídeo, ao contrário das anteriores [xii].

As músicas não foram baseadas no esquema de tensão e relaxamento em cadências como prevê o sistema tonal. Note-se que este sistema de tensão e relaxamento tonal se caracteriza por uma colisão que acontece entre certas frequências. Ou seja, a tensão é produzida pelas notas (frequências) em si, que quando tocadas juntas não soam harmoniosas, a ponto de produzir uma tensão tal que se requer outras notas em seqüência destas para resolver, harmonizar o atrito anterior. Para poder se lidar corretamente com este sistema é necessário seguir várias regras que começaram a ser estipuladas no século XVIII [xiii].

Neste vídeo foram usadas composições modais, as quais possuem as suas estruturas em alicerces bem diferentes das composições tonais, elaborando o progresso da música em elementos como: timbre, melodia, contraposição entre os planos (o que está no 2º plano vem para o 1º e vice-versa). Neste caso, há maior liberdade para se acrescentar ou retirar elementos em qualquer parte da música [xiv], o que é útil quando se começa a sonorizar o vídeo sem ele estar totalmente produzido. Além disso, o compositor não precisa se preocupar em manter um esquema harmônico (os esquemas de tensão e relaxamento da tonalidade) para elaborar a música.

A música da primeira parte começa apresentando lentamente os elementos que vão fazer parte dela. Assim, a partir de uma simples idéia, consegue-se um maior tempo de música interessante, mantendo a atenção do espectador pelo constante acréscimo de novos elementos, já que visualmente não há muitos acontecimentos. Um exemplo de sobreposição de planos (sistema modal citado no parágrafo anterior) é o ostinato (trecho musical repetido várias vezes) que aparece na região mais aguda, o qual vai permanecer praticamente por todo o restante da música, sendo evidente somente no começo da trilha. Note que o timbre de cada um dos elementos é diferenciado, renovando o interesse musical. A percussão foi elaborada para dar continuidade nos diversos elementos. Na primeira vez em que o Edu enxerga o portal (ainda na primeira

parte), uma melodia mais vibrante que as anteriores (violinos, viola, cello) e os tímpanos (instrumento orquestral de percussão) entram pontuando a cena. Esta melodia volta em cena mais tarde de maneira menos vibrante, pois não há nenhum momento a ser pontuado.

A música da segunda parte se apresenta como um pano de fundo, tendo um comportamento bem diverso da música anterior, sem interagir com a movimentação do vídeo e sem nenhum ponto de clímax, favorecendo a presença no 1º plano dos efeitos sonoros. Os efeitos sonoros das esferas foram trabalhados em função da sua posição no espaço e de sua conseqüente movimentação. As lateralidades foram resolvidas pelo maior posicionamento do som na caixa esquerda ou na direita e pela transição entre uma caixa e outra. Em função da movimentação vertical utilizou-se uma alteração da frequência de maneira descendente contínua quando a esfera descia e ascendente contínua quando a esfera subia, facilitando a compreensão da movimentação das mesmas.

Na parte em que Edu se encontra dentro do cubo suas reações de espanto são bem marcantes, por isso foram utilizados efeitos sonoros (que não fazem parte da música) para pontuá-los. Esta é a primeira vez no vídeo que ocorre um efeito sonoro para ilustrar uma ação que não possuiria nenhum tipo de emissão sonora. Algo semelhante somente ocorreu no momento que antecede a visão do portal por Edu, na primeira parte do vídeo, quando um novo timbre (cellos) com a ritmica forte aparece para justificar o espanto e a surpresa do personagem, porém, não é claro que ocorre um som real para tal ação. A solução encontrada pela equipe de produção sonora foi utilizar um timbre (cello) que é familiar para a grande maioria e o integrar à música que acontecia no momento, então um novo fato ocorreu, mas quando integrado totalmente à música, perde sua função de efeito sonoro. Todos efeitos anteriores à cena do cubo foram para acompanhar algo que evidentemente estaria emitindo um som.

Basicamente, os efeitos foram sintetizados no Sonic Foundry Sound Forge. Trabalhou-se somente com informações MIDI e amostras de instrumentos. Cada música foi gravada em

faixas separadas e os efeitos foram sincronizados no Adobe Premiere a posteriori.

11. Referências

-
- [i] BATTAIOLA, André L et al. Teaching Computer Graphics in a Ludic Learning Environment. Anais do WECTE'2044 World Congress on Engineering and Technology Education
 - [ii] WICK, Rainer. Pedagogia da Bauhaus. Martins Fontes. 1982.
 - [iii] LUCENA JR., Alberto. Arte de Animação – Técnica e Estética Através da História. SENAC. Pág. 176 e 190
 - [iv] 3D Max 5.1. Help / User Reference / Tutorial / Box-modeling
 - [v] 3D Max 5.1. Help / User Reference / Tutorial / Biped and Physique
 - [vi] 3D Max 5.1. Help / User Reference / Tutorial / Skin
 - [vii] 3D Max 5.1. Help / User Reference / Tutorial / Lights
 - [viii] 3D Max 5.1. Help / User Reference / Tutorial / Raytrace
 - [ix] Plasma. Help / User Reference / Managing Scenes and Projects / Exporting to Shockwave 3D
 - [x] Plasma. Help / User Reference / Havok Dynamics / Havok Objects / Rigid Bodies
 - [xi] Mayer, R. E.; Moreno, R. A split-attention effect in multimedia learning: evidence for dual processing systems in working memory. *Journal of Educational Psychology*, 90(2), 1998.
 - [xii] Menezes, Flo - Música Eletroacústica :Historias e Estéticas, Edusp, 1996, São Paulo.
 - [xiii] Hindemith, Paul – Traditional Harmony, SHOTT & Co. LTD., 1949, Londres.
 - [xiv] Schoenberg,Arnold – Fundamentos da Composição Musical, Edusp, 1993, São Paulo.