

NINTENDO BLAST

www.clubblast.com.br

MONSTER HUNTER B Tri

EDIÇÃO Nº08 MAI/2010



ÍNDICE

MARIO BITS	Comparando SMG2 com SM64	03
ANÁLISE	Picross 3D (DS)	06
ANÁLISE	2010 FIFA World Cup (Wii)	09
PERFIL	Cha Cha	12
NINTENDO CHRONICLE	Nintendo 64 e os Primeiros Erros	14
ANÁLISE	Monster Hunter Tri (Wii)	19
ANÁLISE	Dementium II (DS)	26
PRÉVIA	Sin & Punishment 2 (Wii)	29
BLAST FROM JAPAN	Monster Hunter G (Wii)	33
GAMEDEV	Conceitos de Programação no XNA	35
COLUNA	Tempo de Jogar	43

REDAÇÃO

- Rafael Neves
- Alveni Lisboa
- Sérgio Oliveira
- Gustavo Assumpção
- Filipe Gatti
- Eduardo Jardim
- Pedro Pellicano

REVISÃO

- Alveni Lisboa
- Rafael Neves
- Sérgio Estrella
- Sérgio Oliveira
- Henrique Dória
- Douglas Fernandes
- Filipe Gatti

DIAGRAMAÇÃO

- Sérgio Estrella
- Gustavo Assumpção
- Felipe de França
- Rafael Esau
- Alex Silva
- Marcus Dejean
- Diego Miler

INTRODUÇÃO

Monster Hunter, quem diria, está conquistando o ocidente! Com seu mundo vasto e modo online sensacional, a franquia da Capcom chega ao Wii em grande estilo. Nessa edição, você confere uma mega análise do game, assim como um Perfil e um Blast from Japan dedicados à série. Também trazemos uma prévia de Sin & Punishment 2 e análises de Picross 3D, Dementium II, e já em clima de Copa, uma análise especial de 2010 FIFA World Cup para Wii. Ah!, e o Mario Bits é dedicado àqueles que ficaram hipnotizados por Super Mario Galaxy 2! Aproveite, e boa leitura!

- Sérgio Estrella

APOIAM A REVISTA



MARIO BITS

Comparando Super Mario Galaxy 2 com Super Mario 64

Quando penso em Super Mario Galaxy 2 e no fato desta ser a primeira vez em que um título 3D do Mario ganha uma sequência direta no mesmo console, me lembro de algumas falhas de antigamente. Não as sequências dos tempos de 8-bits, por favor... me refiro ao que começou a acontecer quando anunciaram o sistema Nintendo 64DD, já na era tridimensional. O 64DD ("DD" vindo de Disk Drive) era uma máquina que se conectava ao Nintendo 64 regular e podia reproduzir jogos em um disquete de alta capacidade com design próprio (obrigado, MadMax77!). Infelizmente ou não, acabou sendo um fracasso comercial, o que o impediu de ser lançado fora do Japão. Contando com apenas 9 jogos em toda a sua história, 4 destes eram da série Mario Artist (Mario Artist: Communication Kit, Mario Artist: Talent Studio, Mario Artist: Paint Studio e Mario Artist: Polygonal Studio); jogos como Sim City 64 e F-Zero X Expansion Kit também deram as caras no Japão, mas, obviamente, nenhum desses softwares conseguiu cativar a população gamer.

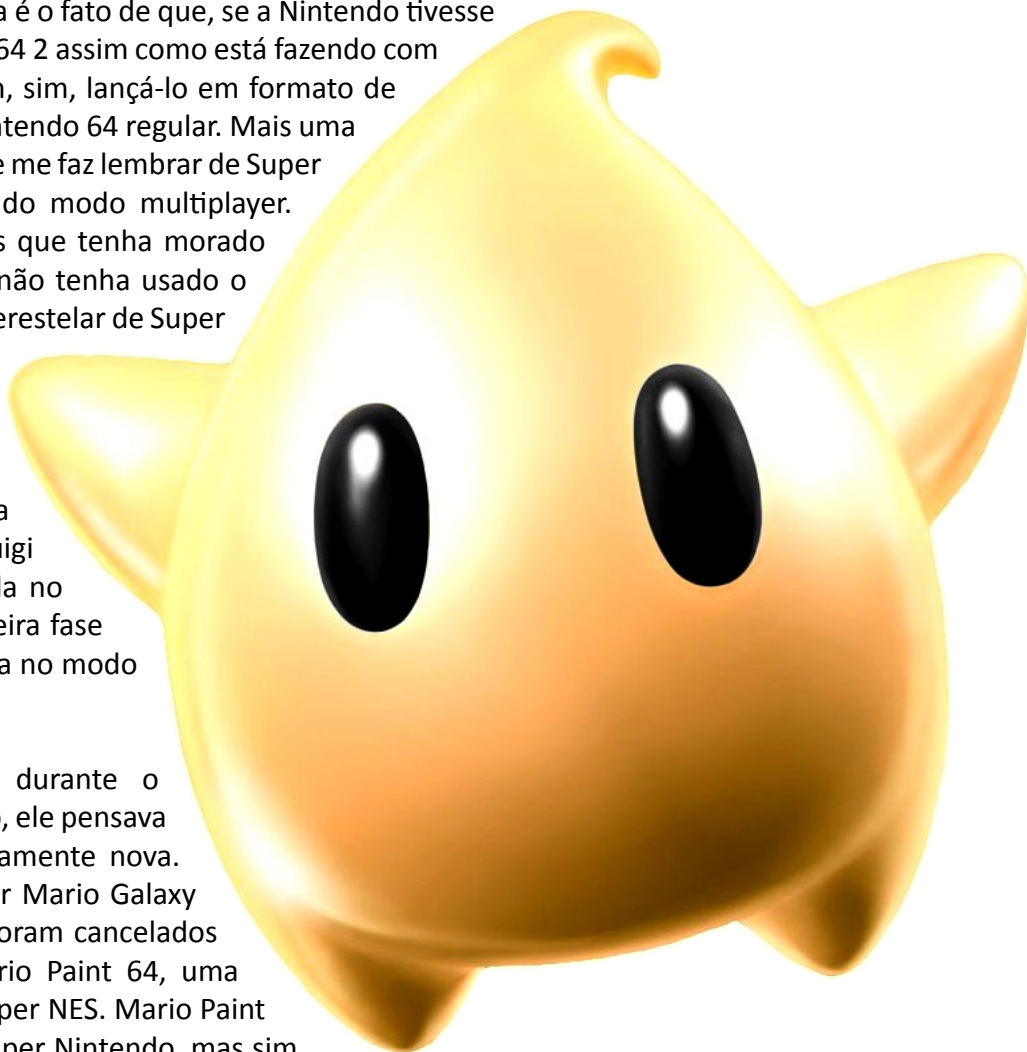




Entre os jogos cancelados, ficaram os outros 4 da série Mario Artist (Mario Artist: Game Maker, Mario Artist: Graphical Message Maker, Mario Artist: Sound Maker, Mario Artist: Video Jockey Maker) e, entre outros, o projeto Super Mario 64 2, que foi anunciado no fim de 1999. O único motivo que cancelou logo o começo de sua produção foi o fracasso do 64DD, no entanto, uma fase demo foi feita. É possível que alguns dos conceitos usados neste jogo estejam incluídos, hoje em dia, em Super Mario 64 DS, para Nintendo DS. Também não seria surpresa que Super Mario Sunshine ou Super Mario Galaxy tivessem pego emprestado os elementos projetados para Super Mario 64 2, como já mencionado pela Nintendo em alguma E3.

O que me intriga é o fato de que, se a Nintendo tivesse realmente investido em Super Mario 64 2 assim como está fazendo com Super Mario Galaxy 2, eles poderiam, sim, lançá-lo em formato de cartucho, tornando-o operável no Nintendo 64 regular. Mais uma característica do Super Mario 64 2 que me faz lembrar de Super Mario Galaxy 2 é o uso planejado do modo multiplayer. Como todo mundo já sabe, a menos que tenha morado numa rocha nos últimos meses (ou não tenha usado o Twitter de modo útil), a sequência interestelar de Super Mario Galaxy permitirá jogatina simultânea (cooperativa) entre 2 jogadores; confirmações indicam que o player 2 poderá auxiliar o jogo como uma Luma (o que explica sua presença na capa). A presença de Luigi em Super Mario 64 2 era confirmada no modo multiplayer, e, de fato, a primeira fase do jogo cancelado tomava a mecânica no modo cooperativo.

Miyamoto também afirmou que, durante o processo de desenvolvimento do jogo, ele pensava em lançá-lo numa plataforma inteiramente nova. É o que todos esperávamos de Super Mario Galaxy 2, não? Outros títulos da Nintendo foram cancelados após anúncio inicial, tais como Mario Paint 64, uma sequência clara de Mario Paint de Super NES. Mario Paint 64 não usaria o clássico mouse do Super Nintendo, mas sim uma leva de novos acessórios e periféricos que infelizmente foram mantidos em segredo.



Super Mario RPG: Legend of the Seven Stars também teria uma sequência compatível com o 64DD, e o projeto era chamado Super Mario RPG 2. Dizem que este é o nome original de Paper Mario, mas teve que ser mudado quando a Square e a Nintendo, os co-desenvolvedores de Super Mario RPG, se separaram. O fato de que Super Mario RPG pertencia a ambas as companhias tornou o lançamento de uma sequência direta sem a permissão da Square em algo ilegal. A versão para 64DD foi cancelada, mas mudanças menores foram feitas e Paper Mario foi lançado para o Nintendo 64 regular.

A questão é que essas e outras dúvidas ainda permanecem em total segredo. Só o que rolaram foram mais e mais boatos que poluem e poluem tópicos de fóruns online. Quem não garante que a polêmica frase “L is real 2401” refere-se, na verdade, ao modo cooperativo de Mario 64 2? Uma grande incógnita permanece nesse capítulo da saga dos vídeo-games, e bem que a Nintendo poderia publicar um livro respondendo a nossas perguntas, com certeza venderia que nem água no Saara!

Super Mario Galaxy 2 (Wii)



Super Mario 64 (Nintendo 64)



Tenho certeza que a Nintendo não vai mais repetir os erros do passado, e além do mais, tudo indica que Super Mario Galaxy 2 será um estrondoso sucesso. Portanto, faça sua encomenda, ou vá guardando suas moedas e preparando os ossos para uma viagem sideral de tirar o fôlego. Continue ligado no Nintendo Blast e no Reino do Cogumelo para as novidades galácticas.



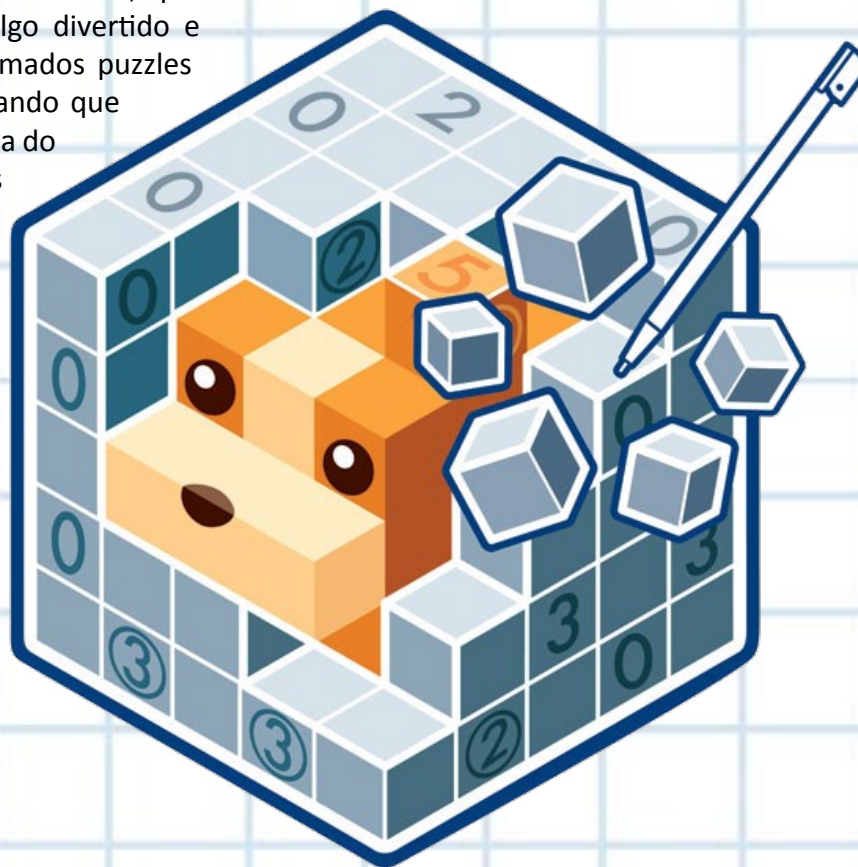
Análise

Picross 3D (DS)

PICROSS 3D

Depois de Picross DS transformar as malhas quadriculadas, que eram um terror nas aulas de matemática, em algo divertido e prático, chega a seqüência de um dos mais renomados puzzles do DS: Picross 3D. Em primeiro lugar, não vá achando que bloquinhos tridimensionais saltarão para fora da tela do portátil. O nome "3D" não tem nada a ver com os novos filmes em três dimensões que invadiram o cinema. Quando dizemos que o novo Picross é 3D, estamos falando da substituição de malhas quadriculadas por placas cúbicas. Logo, ao invés de figuras bidimensionais, teremos formas geométricas volumosas.

Mesmo assim, o jogo segue bem o estilo do antecessor, recheado de puzzles cada vez mais desafiadores que, quando resolvidos, concedem imagens animadas no formato do quebra-cabeça montado. Para jogadores apaixonados por enigmas e para aqueles que procuram somente uma distração para o dia-a-dia, Picross 3D cumpre bem o seu papel.



Do Quadrado ao Bloco

A maioria das séries de vídeo-game já se depararam com uma certa ponte. E o termo "ponte" não é uma metáfora para expansão de mercado ou de público consumidor. A expressão refere-se à ponte dimensional. Quando uma série se dá bem com jogos 2D, ela logo vê novas oportunidades no 3D e cruzam um caminho rumo a esta nova dimensão.

A passagem do plano para o tridimensional conseguiu trazer grandes benefícios para franquias como Mario e Zelda, mas também fizeram outras sofrerem muito, como os diversos Bombermans que saíram para o Nintendo 64. Sob esta visão, o que deveria ser





uma ponte de livre circulação acaba por se transformar numa peneira, selecionando somente alguns entre vários. O caso é que Picross cruzou esta ponte e acabou por abrir novos horizontes e muitas adições para o jogo de enigmas. Em Picross 3D, os puzzles são resolvidos em formas geométricas tridimensionais, divididas em vários cubinhos. Os cubos agem da mesma forma que os quadradinhos das malhas de Picross DS. Logo, o jogador terá de se guiar pelos números nas extremidades das formas para marcar alguns cubos e excluir outros.

Utilizando-se muita lógica e desafiando os jogadores, os puzzles começam bem fáceis e cheios de tutoriais, mas logo se tornam complicadíssimos e passam a requerer mais e mais tentativas, uma vez que o jogador precisa cumprir os quebra-cabeças no tempo certo e sem errar muitas vezes. A perspectiva em 3D exige que o jogador mude o ângulo de visão a fim de poder enxergar melhor as soluções para o puzzle, nesse aspecto a stylus funciona bem para girar e virar as formas geométricas com precisão.

Mas a tridimensionalidade também trouxe certos transtornos, pois quando você exclui um cubinho que não será usado, o coitado é literalmente destruído, a fim de modelar uma figura ao fim da partida. Então se você for muito apressado, poderá destorcer a imagem a ponto de ficar difícil enxergar o enigma. É sempre recomendável que você destrua logo as colunas zero (onde nenhum cubo é usado) e pintar logo as linhas com números grandes para depois começar a parte mais complicada do puzzle.

Passa-tempo ou Grande desafio?

A casualidade nos vídeo-games é um assunto muito discutido na atual geração, principalmente no Wii e DS. O imenso surto de jogos feitos apenas para crianças e pessoas da meia idade é muito criticado pelos jogadores mais experts. A grande questão é que são poucos os jogos que conseguiram balancear os dois lados, proporcionando diversão para qualquer tipo de jogador. Bem, é mais ou menos por aí que Picross 3D está. Nesse meio-termo, tal qual foi seu antecessor. Os jogadores menos experientes e que procuram no game uma distraçãozinha poderão jogar nos níveis mais fáceis, recheados de tutoriais e com baixo nível de desafio. Já aqueles que literalmente estão num “quebra-cabeça”, irão pelo caminho dos enigmas complicados e trabalhosos, procurando aproveitar todas as aproximadas 30 horas de jogo. Isso só depende do jogador e do que ele espera do jogo.

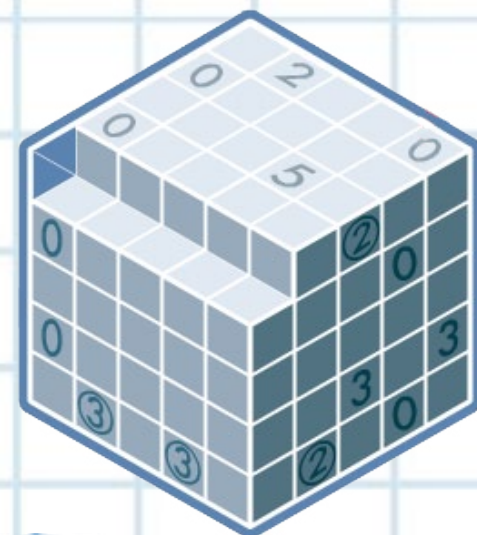
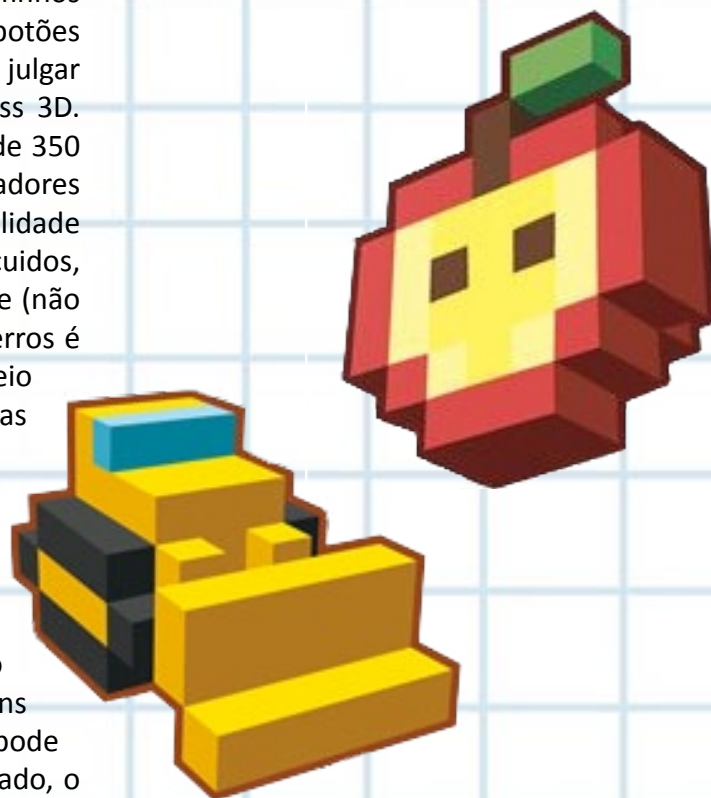


Ligando o jogo, tudo parece muito simples: Mascotes fofinhos dando instruções meio óbvias e menus recheados de botões grandes e coloridos. Bem, já dizia alguém que não se deve julgar o conteúdo pela forma, e tal regra se aplica bem à Picross 3D. Pois é por baixo desta fofura toda que se escondem cerca de 350 enigmas prontos para serem desvendados! E, para os jogadores mais hardcore, informo que neste game a mecânica de penalidade muda em relação ao anterior. Ao invés de erros e descuidos, aumentaram gritantemente o tempo de resolução do puzzle (não podendo passar de 1 minuto). A questão do tempo e dos erros é dividida, há um limite de descuidos e um limite de tempo meio que paralelos e o jogador deve sempre estar entre essas duas marcas.

Quebrando a Cabeça

Picross 3D consegue evoluir em relação à Picross DS em muitos aspectos, mas a passagem para o mundo tridimensional não foi completamente benéfica. Digamos que houveram alguns “escorregões”. A depender do ângulo de visão, o jogador pode terminar se atrapalhando e acabar por destruir o bloco errado, o que é um transtorno, pois há um limite de erros e há jogadores que lutam por vitórias perfeitas. As figuras literalmente “esculpidas” ao decorrer do puzzle não são tão interessantes quanto as imagens em 8-bits de Picross DS e, falando em 8-bits, as referências a jogos passados foram retiradas em Picross 3D. Fases com músicas temas como a de Super Mario fizeram sucesso no primeiro game, mas não entraram na versão 3D. O que é um grande mistério, já que a própria Nintendo fez o jogo. O editor de fases e a possibilidade de dividi-las ao redor do mundo pela conexão Wi-Fi ou local continua lá, mas é claro que construir modelos tridimensionais será mais exigente do que puzzles em 2D. Logo, em Picross 3D, fazer seu próprio enigma requererá prática.

Por fim, Picross 3D é a mais nova versão da série de puzzles e, apesar de formas em 3D dificultarem a visão do enigma e haverem alguns problemas na jogabilidade e em alguns outros detalhes do jogo (como a falta de estágios retrô), consagra-se como um jogo intenso e relaxante, agradando a vários tipos de jogadores. Como você faria para transformar um bloco retangular numa girafa ou até num skate? Se você se sentiu desafiado, não há outro caminho senão detonar Picross 3D!



8.0

Picross 3D (DS)

GRÁFICOS 8.0 | SOM 8.0

JOGABILIDADE 8.0 | DIVERSÃO 9.0

Análise

2010 FIFA WORLD CUP SOUTH AFRICA™ (Wii)



Desde 1998 EA e FIFA lançam games sobre a Copa do Mundo. Aproveitando o embalo da maior competição esportiva do planeta, tais jogos costumam ser bem rentáveis para os produtores. No entanto, assim como os jogos baseados em filmes, os “FIFA World Cup” costumam decepcionar pela falta de conteúdo e de inovação. Infelizmente, **2010 FIFA World Cup: South Africa** não é uma exceção.

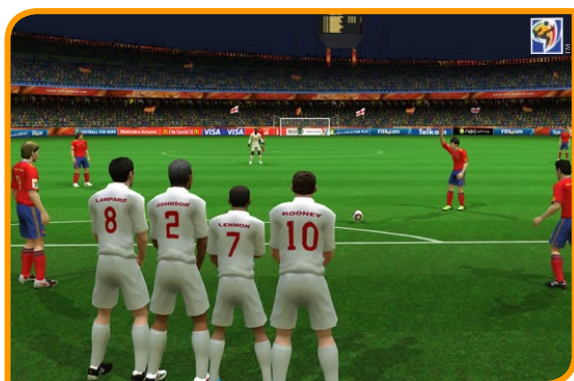
Novidades que não empolgam

A sensação de *deja vú* ao iniciar o game já diz tudo. Você tem aquela impressão de que já viu isso antes. O menu segue o mesmo layout de FIFA 10, com seis opções e imagens cartunescas. A música é a novidade aqui. Toda a trilha sonora lembra temas africanos: ritmos animados, tambores e batidas empolgantes. Tudo para fazer você entrar no clima da Copa.

As seleções são variadas e bem equilibradas. É o jogo com maior número de países selecionáveis: 199. Brasil, Argentina, Itália, Espanha, Inglaterra, Alemanha, França, Portugal, quase não há diferença técnica entre as seleções de ponta. Você pode ganhar da França, mas perder facilmente para o Uruguai ou Dinamarca, por exemplo. Na primeira vez que joguei a final foi entre Brasil e Gana, que bateu Alemanha e Portugal. Estranho, mas perfeitamente possível de acontecer dado o cenário de igualdade que o futebol mundial atravessa.

A principal novidade é o modo “Zakumi’s Dream Team”. Zakumi é o nome do mascote da Copa da África do Sul.





Você começa com um time fraquíssimo e tem que cumprir objetivos em jogos contra seus adversários. Caso cumpridos, você vai destravando conquistas e tem o direito de escolher um jogador rival para reforçar sua equipe. O nível dos adversários vai aumentando e os objetivos acompanham. Até aí, tudo bem. O problema é para conseguir atingir os objetivos propostos. Trocar 10 passes antes de fazer um gol se torna extremamente difícil devido ao plantel fraco e à jogabilidade ruim.

Jogabilidade inalterada

Se você jogou FIFA 10, vai estar familiarizado com o estilo de jogo aqui. É possível optar entre Wiimote, Wiimote+Nunchuk ou Classic Control. O recomendado é usar o Clássico, que fornece maior precisão. Usando o Wiimote+Nunchuk é impossível controlar a força do chute e do passe, por exemplo. Tudo é muito fácil e complicado ao mesmo tempo. Vou explicar.

Primeiramente, os goleiros beiram o ridículo. Mesmo no maior nível de dificuldade, eles são patéticos e tomam os gols mais bizarros que se pode imaginar. Basta correr com a bola, chutar da intermediária e sair pro abraço (é possível escolher entre quatro tipos de comemoração após o gol). Ou seja, vitórias por 1x0 são inexistentes.

Segundo, acertar um passe é muito difícil, pois você controla a força. Isso é interessante quando se quer aplicar longos lançamentos, mas não funciona bem nos passes curtos. Acertar os passes torna-se uma tarefa complicada, o que leva o jogador a optar pelo drible na maioria das vezes – realizado apenas por um apertar de botão.

Por outro lado, cobranças de falta, escanteiros e pênaltis são muito fáceis. Se você pegar o jeito da coisa, fará vários gols e não tomará nenhum. Basta mover a alavanca na hora correta e pronto. Porém, até se acostumar, prepare-se para levar vários gols de bola parada. Esse mesmo mecanismo é aplicado em disputas de bolas aéreas. Quem acertar o momento exato sai com a bola dominada (sim, finalmente é possível dominar a bola no peito!) ou cabeceia pro gol – se estiver dentro da área adversária.

Pó, rimel e blush no visual

O visual do game é ligeiramente superior ao de FIFA 10, mas isso não significa que seja bom. Muito pelo contrário, é um gráfico pobre, feio e inferior até aos jogos de PS2. É cerrilhado, cabeças quadradas, personagens sem expressão facial... A EA apenas aplicou uma maquiagem leve para tornar os personagens mais realistas. Diferente das versões para Xbox 360 e PS3, aqui é difícil reconhecer os jogadores pelas características ou estilo de jogo. Os torcedores são chapados e mal feitos, bem ao estilo Nintendo 64.

Igual ao antecessor, **2010 FIFA World Cup: South Africa** tem uma barrinha chamada “Momentum”, que mede a empolgação da torcida em cada ataque e influencia na finalização. Há também a câmera lenta nos momentos de perigo. Quando se chuta para o gol, a ação é desacelerada para criar uma expectativa: a bola vai entrar ou não? Infelizmente, nem sempre conseguimos ver. Isso porque a câmera perde o foco da bola, e daí o replay se torna indispensável.

O modo online cumpre o papel burocraticamente. Para os brasileiros, há quedas de conexão, mas o game funciona bem em rede. No entanto, não há novidade alguma: escolha o adversário e ganhe para alcançar as melhores colocações no ranking.

Bate-bola

A EA continua apostando no público casual para os seus jogos de esporte no Wii. Prova disso é 2010 FIFA World Cup South Africa, que nada mais é do que um remake idêntico de FIFA 10 com seleções apenas. O modo “Zakumi’s Dream Team” tenta ser divertido, mas empaca na jogabilidade ruim. O game peca em facilitar demais as coisas, o que resulta em placares absurdos e gols irritantes. A falta de incentivos, como times clássicos, estádios ou jogadores extras tornam o fator replay praticamente nulo – salvo apenas pelo modo online. É um daqueles títulos para se jogar apenas num domingo à tarde reunido com a família ou amigos. Ponto positivo somente para as músicas que dão todo o clima da competição.

Não foi dessa vez que a nação nintendística recebeu um game de futebol decente. Por enquanto, PES 2010 ainda reina absoluto nos gramados do Wii. Copa do Mundo? Só mesmo na TV, em junho. Vamos lá, Brasil, rumo ao hexa!



5.8

2010 FIFA World Cup South Africa (Wii)

GRÁFICOS 5.0 | SOM 7.0

JOGABILIDADE 5.0 | DIVERSÃO 6.0

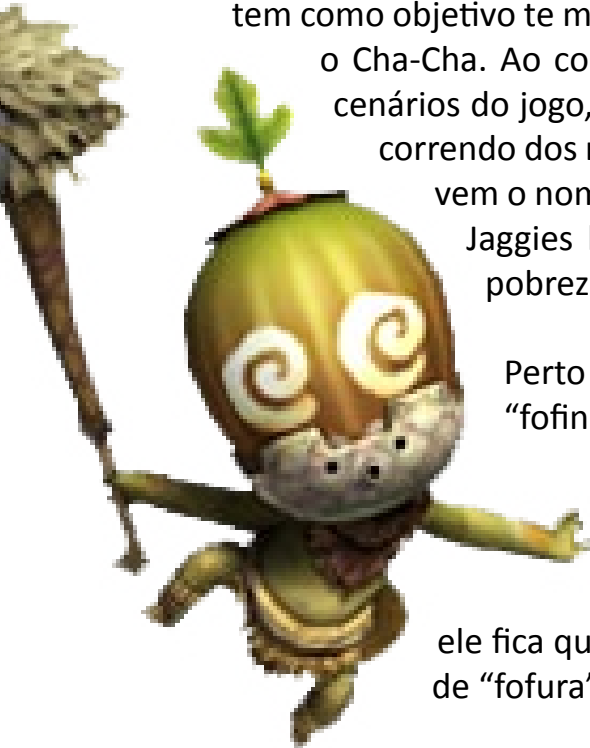


Cha-Cha

É incrível como a fofura e a pureza conseguem existir mesmo em universos completamente hostis. A série Monster Hunter é famosa pela alta violência e monstruosidade de seus elementos, e jogando Monster Hunter Tri dá para se ter uma noção de que tudo no jogo tem como objetivo te matar. Bem, eis que surge uma figura para contrariar tal paradigma: o Cha-Cha. Ao contrário dos monstros dinossaurescos que irão infestar os vastos cenários do jogo, Cha-Cha é um fiel companheiro e talvez o único que não irá sair correndo dos monstros, te deixando na mão. Membro da tribo Shakalaka – e daí vem o nome do mascote – Cha-Cha é encontrado sendo atacado por ferozes Jaggies logo nas primeiras missões de Monster Hunter Tri. Ao salvar o pobrezinho, ele se torna seu grande e inseparável amigo.

Perto de mascotes como o Yoshi ou Navi, Cha-Cha está longe de ser “fofinho”. Ele não possui cores fortes e vibrantes nem é constituído por gráficos em cel-shading, na verdade, ele é considerado até meio feio por muitos jogadores. Mas se compararmos Cha-Cha com os monstros hostis de Monster Hunter, ele fica quase bonito, e é daí quem vem o único vestígio de “fofura” que restou à série MH.

Monster Hunter não seria nada se não fosse pelo seu alto grau de estratégia. E o mesmo se aplica ao seu mascote, pois até ele é envolvido numa mecânica de balanceamento de habilidades, uso de máscaras e danças estratégicas. Cha-Cha tem diversos atributos, como se fosse um personagem do jogo, e estes atributos são significativos numa batalha. Conforme vai evoluindo, o mascote também adquire novas técnicas fundamentais numa luta. Tendo um amor por danças e máscaras, o tribal Cha-Cha tem nesses hobbies características estratégicas para o jogo. As máscaras da criatura são várias e todas têm sua própria aparência, mas não é a estética que será avaliada pelo jogador, e sim as vantagens e desvantagens de cada acessório. E as danças ritualísticas também têm papel de destaque, concedendo bônus ao jogador, essenciais para o cumprimento de missões no modo single player.



Cha-Cha é um tanto temperamental, uma vez que sua personalidade varia muito, tudo depende da máscara que traz em sua cara. Um máscara em forma de semente que cubra seu rosto é capaz de deixá-lo altamente agressivo, e basta trocar para uma máscara estranha e metálica para acalmar seus nervos. Mas seu cajado é característico, e isso nada pode mudar, bem como sua alta fidelidade ao jogador e elevado companheirismo.



Acorn Mask

Mystery Pot Mask



Fluffy Mask

Ancient Mask



Grill Mask

False Feline



Lamp Mask



Nintendo Chronicle

por Gustavo Assumpção

PARTE 8: O NINTENDO 64 E OS PRIMEIROS ERROS

No nosso último Nintendo Chronicle vimos como o Super Nintendo alcançou o sucesso no mercado de consoles domésticos, suas dificuldades e também o duro golpe que a Nintendo sofreu da até então parceira Sony, golpe esse que se refletiu nos sucessos e fracassos do seu terceiro console, o Nintendo 64.



Apenas com o seu segundo console doméstico, o Super Nintendo, a Big N já havia conquistado praticamente todo o mercado dos consoles. Sua hegemonia era incontestável. Ao fim do ciclo de vida do Super NES a empresa de Kyoto vendera mais de 49 milhões de unidades e faturado bilhões e bilhões (quem sabe trilhões) de dólares por todo o mundo.

Apesar de todo o sucesso, já era hora de trabalhar em um novo console. Em 1993 a Nintendo liberou as primeiras informações do seu novo console, até então conhecido por "Project Reality" – um console que seria capaz de criar gráficos tridimensionais com a tecnologia da Silicon Graphics - uma das pioneiras em computação gráfica tridimensional. Entretanto, por dois anos não houve mais nenhuma informação sobre a evolução e andamento do projeto.

Em 1995 a Big N começou a adiar o lançamento do seu novo console, já conhecido por Nintendo 64 à época, repetindo o mesmo erro do seu antecessor, o Super Nintendo. Só isso já era motivo suficiente para a indústria e a mídia especularem que o console não teria tantos jogos no seu lançamento, o que realmente aconteceu.

Finalmente, em junho de 1996, o Nintendo 64 chegou ao mercado após inúmeros adiamentos e contava com Super Mario 64, Wave Race 64 e PilotWings 64 em seu portfólio de estreia. Em seguida o console foi lançado nos Estados Unidos, Europa, Austrália e Brasil. Com a promessa de trazer gráficos tridimensionais jamais vistos, o Nintendo 64 inovou em diversos aspectos, porém alguma coisa não estava certa.

Um console à frente do seu tempo, mas com cartucho

É inegável que o Nintendo 64 foi um console à frente do seu tempo – sua arquitetura de 64 bits era superior a todas as outras presentes no mercado na época (32 bits), possuía um processador gráfico dedicado capaz de produzir cores de até 32 bits e grande capacidade de processamento de áudio. Para o usuário, o que era mais perceptível eram as 4 portas de entradas de controles (algo quase inédito, pois só havia sido explorado em 1977 pelo Bally Astrocade) e um direcional analógico no controle (também implementado uma única vez pelo Arcadia 2001).

Aliás o controle é algo que realmente chamava muita atenção no N64 – seu formato de tridente com dez botões e um gatilho na traseira, apesar de ser estranho, era extremamente confortável e funcional. Com a adição do direcional analógico o jogador agora tinha a possibilidade de apontar para 256 direções diferentes. O gatilho na traseira do controle, conhecido como Z Trigger, proporcionava realismo jamais alcançado principalmente nos shooters em 1ª pessoa. Entretanto nem tudo foi um mar de rosas.

O principal erro da Nintendo, que talvez tenha afetado todo o potencial do Nintendo 64, foi insistir na utilização dos cartuchos como mídia de distribuição de jogos. Enquanto a concorrência já distribuía jogos em mídia digital (CD-ROM), a Nintendo espantou produtores e limitou o potencial do console com a utilização dessa mídia.

Antes mesmo do lançamento oficial do console, a Nintendo perdeu a exclusividade sob a série Final Fantasy, que migrou para os consoles da Sony. Logo em seguida foi a série Dragon Quest e mais adiante toda a Squaresoft que migrou para o Playstation. O mais engraçado aqui é que Final Fantasy VII ainda chegou a ser demonstrado na plataforma da Big N.

A decisão da Squaresoft, e de tantas outras softhouses que migraram seus jogos para o Playstation, se deu aos custos de produção e às limitações dos cartuchos. Enquanto um lote de jogos para Playstation levava apenas alguns dias para ser produzido e custava relativamente pouco, a produção dos cartuchos para o Nintendo 64 levava semanas e só podiam ser produzidos pela Nintendo. Além disso os chips utilizados nos cartuchos tinham que ser reprogramados quando comprados, o que dificultava mais ainda o trabalho de manufatura dos cartuchos e aumentava o seu custo.



Com capacidade de armazenamento de 64Mb (10 vezes menos que os CDs da época), não era possível armazenar as famosas animações gráficas (CGs) que rapidamente se popularizaram no Playstation da Sony. Pior, isso dificultava bastante o trabalho dos programadores para a renderização das texturas – muitas vezes imagens iguais eram utilizadas para diferentes propósitos. Apenas um único jogo conseguiu a façanha obtida nos CD-ROMs no Nintendo 64 – Resident Evil 2! Para isso a Capcom teve que desenvolver um novo sistema de compressão de dados para que tudo desse nos 64Mb dos cartuchos – trabalho e dinheiro que nem todos estavam dispostos a “perder”.

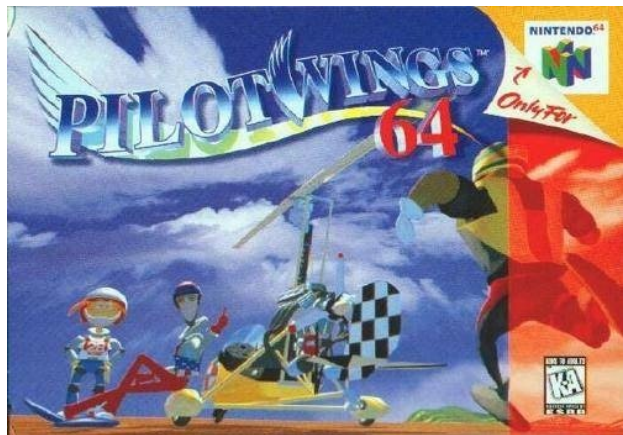
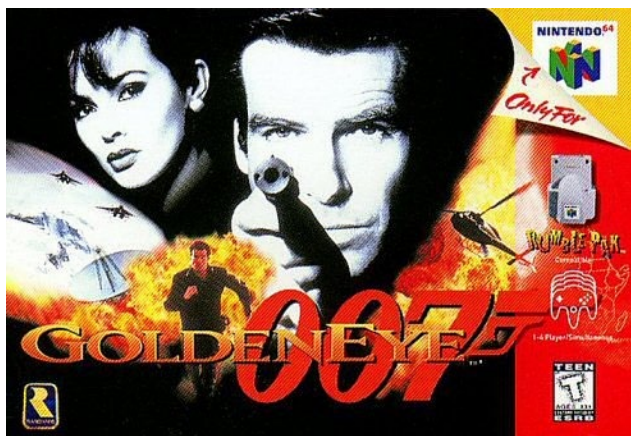
O resultado disso é que o preço de produção dos cartuchos tinham que ser repassados para o consumidor e cada cartucho chegava a custar algo em torno de US\$80,00. Em contrapartida, os CDs que tinham um custo de produção extremamente baixo, dando maior margem de lucro às softhouses, raramente custavam mais que US\$50,00. O agravante de planejamento de produção dos cartuchos complicava ainda mais a situação – se os fabricantes pedissem mais cartuchos que a demanda, eram obrigados a arcar com os prejuízos; se pedissem de menos, perdiam a oportunidade de vender mais, já que haviam que esperar semanas para o novo lote chegar às lojas.



Bons jogos para todos

Apesar de todos os contratemplos, o Nintendo 64 teve seus momentos de glória. Além do já falado controle com design inovador, a Nintendo conseguiu, mais uma vez, trazer inovações para o console. Periféricos como o Rumble Pak, que fazia o controle vibrar de acordo com ações no jogo, foram inéditos e copiados pelos concorrentes. Além dele tivemos o Expansion Pak, que em alguns jogos habilitava gráficos em alta resolução (640x480 pixels na época), Transfer Pak, que permitia a comunicação entre Nintendo 64 e Game Boy (especialmente em jogos dos Pokémons) e outros que não fizeram tanto sucesso e/ou foram lançados apenas no Japão, como o 64 Disk Drive.

Assim como os acessórios e periféricos que são uma marca registrada da Nintendo, os jogos não deixaram a desejar. A Nintendo conseguiu, mesmo com um console desfavorecido por suas próprias limitações, montar uma biblioteca de jogos respeitável, muitos deles clássicos até hoje. Super Mario 64 é um bom exemplo disso – primeiro jogo disponível para o console e que já utilizava boa parte de sua capacidade. Em Super Mario 64 os jogadores puderam sentir todo o poder gráfico do console em cenários mágicos, coloridos e totalmente tridimensionais. As funcionalidades do novo controle também foram exploradas ao máximo, em especial o direcional analógico.



Jogos como Star Fox 64, Mario Kart 64, Mario Party, Pokémon Stadium, Castlevania, Donkey Kong 64, The Legend of Zelda e Mickey's Speedway 64 procuravam atrair a atenção do público infantil e os vindos da antiga plataforma da companhia, enquanto GoldenEye 007, Perfect Dark, Resident Evil 2, Carmageddon 64, Destruction Derby 64, Doom 64, e Duke Nukem 64 atraíam a atenção do público mais adulto, mostrando claramente a preocupação da Nintendo em atender todos os públicos.

Mesmo com a dificuldade de programar para o console, o Nintendo 64 teve algo em torno de 400 títulos lançados para ele, muitos deles inovadores como Star Wars: Battle for Naboo, que trazia efeitos de chuva, neve e fogo de uma maneira que só foi vista posteriormente no Playstation 2! Seu último jogo foi Tony Hawk's Pro Skater 3 lançado em 2002.

Claramente a Nintendo dispunha de um console pronto para as mídias digitais: processadores distintos para funções específicas, arquitetura de 64 bits e um grande leque de produtoras prontas para a tridimensionalização de seus jogos. No entanto, o duro golpe aplicado pela Sony causou o atraso do lançamento do console e, possivelmente, uma adaptação da plataforma para a já defasada tecnologia dos cartuchos.

Quem teve a oportunidade de ter um Nintendo 64 sabe que ele foi um console memorável, proporcionador de bons momentos de diversão com os amigos. A Nintendo vendeu algo em torno de 33 milhões de unidades e de uma vez por todas percebeu que não poderia mais errar – acabou perdendo 2 gerações de liderança para a Sony, que com o seu primeiro console conquistou 52% do mercado. Era hora de aprender com seus próprios erros. Será mesmo? Esse já é um assunto para a nossa próxima coluna Nintendo Chronicle. Até lá!

Análise

Monster Hunter Tri (Wii)



Dedicação, dedicação e mais dedicação

Aos que não conhecem a série, **Monster Hunter Tri** pode ter uma jogabilidade esquisita de início. É complicado estar acostumado aos jogos de ação em terceira pessoa onde é possível travar a mira nos inimigos; ter controle automático de câmera; ou até mesmo ver uma barra com a vida dos oponentes e, de repente, estar sem nenhum desses elementos. Você praticamente volta no tempo, naqueles dias onde passava horas brigando com o controle até pegar o jeito.

Eu, que comecei minha jornada na série há muito tempo, ainda no PSP, posso afirmar que a curva de aprendizado diminuiu muito. A Capcom implementou um sistema muito mais amigável, com um mundo enorme e várias áreas para explorar sem perigo, onde é possível aprender com calma e muita dedicação. Aliás, dedicação deve ser o substantivo gravado na sua mente como peça chave para o sucesso. Se prepare para muitas horas de jogo até começar a entender o funcionamento das armas e escolher sua preferida. Adicione mais algumas horas para estudar os possíveis combos e a melhor forma de aplicá-los nos inimigos. Aí sim, você estará pronto para entrar em missões contra criaturas mais poderosas. Se você não é o tipo de jogador que está disposto

A série Monster Hunter é imensa no Japão, não há dúvidas disso. Seu estilo de jogo recompensa o jogador puramente por seu esforço e habilidade, e é preciso muita paciência para aprender com maestria a forma certa de enfrentar as gigantescas criaturas que habitam o mundo. Porém, o mercado Ocidental sempre se sentiu intimidado com sua forma de jogo, o que dificultou o sucesso e reconhecimento comercial nos Estados Unidos e também por aqui. Mas a Capcom decidiu que vai mudar essa imagem, e em **Monster Hunter Tri**, primeiro título da série que não foi desenvolvido para um console da Sony, ela fez o impossível se tornar possível, mantendo a fórmula principal no mesmo estilo hardcore que os fãs da série esperam, mas tornando tudo mais simples e prático, sem assustar o novo público.





a gastar horas e mais horas para tirar o proveito máximo do que pode ser oferecido, então **Monster Hunter Tri** não é para você – caso contrário, prepare-se para uma das experiências mais incríveis que você já sentiu em toda a sua vida de gamer.

Mais cedo ou mais tarde você perceberá outra coisa: não existe um sistema de nível. Os monstros não te dão experiência, seus stats não sobem ao derrotar criaturas mais fortes e não existe um “número” que diz em que nível você está. Em **Monster Hunter Tri**, tudo depende da sua habilidade e seus equipamentos, que são confeccionados com itens adquiridos ao matar os monstros. Portanto, a impressão de que você está ficando mais forte não vem com a mudança de um número, e sim de seu próprio sentimento de superação ao aniquilar aquele maldito Rathalos que te derrotou uma dezena de vezes antes.

O grande objetivo... ou não tão grande assim.

Muitos dizem que o trunfo da série Monster Hunter está no modo online – e eles não estão enganados. Mas isso não significa que não há um modo single-player digno de ser jogado. Ao chegar em *Moga Village*, um pacífico vilarejo que vive a base da pesca, você recebe o trabalho de caçador local. Seu objetivo é fortalecer o vilarejo e aniquilar a terrível criatura que ameaça constantemente os moradores da região – o Lagiacrus, uma espécie de dragão que vive debaixo d’água e estampa a caixinha do seu jogo original.

Tudo isso realizado com o seu companheiro Cha-Cha – uma criança perdida da tribo Shakalaka –, responsável por te ajudar nas quests do modo single player, tendo um amplo espectro de customização. Ele pode parecer fraco, mas faz uma grande diferença, principalmente quando chama a atenção dos monstros para cima dele, abrindo uma janela para você atacar por uns momentos sem se preocupar.

Basicamente é isso. Claro que até ter equipamentos e habilidade suficientes para combater seriamente o Lagiocrus será necessário completar uma série de quests que implicam em retalhar monstros menores, mas igualmente furiosos, durante a sua jornada. Na essência, é nisso que se resume a história do jogo. Todavia, chegará o momento que você vai perceber que derrotar o Lagiocrus é apenas o começo. Mais do que a metade dos monstros em **Monster Hunter Tri** não aparecem pelo menos durante as primeiras 50 horas de jogo, e as inúmeras quests no modo online renderão algumas boas centenas de horas aos fãs mais vorazes e sedentos por criaturas monumentais.

O vasto arsenal e o imenso leque de itens

Existe um vasto arsenal de armas que podem ser usados em **Monster Hunter Tri**. Elas variam desde as conhecidas Great Sword e o temível Hammer, até o novíssimo Switch Axe, arma com um potencial de destruição extremamente alto. Para aqueles que preferem atacar a distância, as Bowguns continuam presentes.

Cada arma representa uma “classe”, pois a jogabilidade com cada uma delas é extremamente distinta. Portanto um jogador especialista em usar a Long Sword pode ser considerado um novato quando dado um Sword and Shield a ele.

Vale lembrar também: é extremamente não recomendado usar o WiiMote para jogar **Monster Hunter Tri**. Este título é muito melhor aproveitado se usando o Classic Controller. Na verdade, a experiência é completamente diferente e é praticamente impossível derrotar as criaturas mais fortes sem um Classic Controller por perto.

Tanto as armas, quanto as armaduras, são criadas a partir dos mais variados materiais, compostos por escamas e outros pedaços que são adquiridos ao “abrir” as criaturas que você matou. Geralmente, para criar um set completo com arma e armadura do mesmo tipo, é necessário uma grande quantidade de materiais de um determinado monstro. Ou seja, para ter uma armadura completa de Diablos, por exemplo, prepare-se para enfrentá-lo várias vezes.



Os outros materiais necessários para a confecção destes equipamentos podem ser encontrados nas próprias regiões onde acontecem as quests, ou então na fazenda do vilarejo, onde você pode contratar Felynes e Melynxes, raças de gatos bípedes que falam o idioma dos humanos, para trabalharem e coletarem parte desses recursos para você. Além de serem importantes na confecção dos armamentos, a combinação entre essas centenas de itens podem resultar em outras coisas como poções, armadilhas e até mesmo munição para as armas.

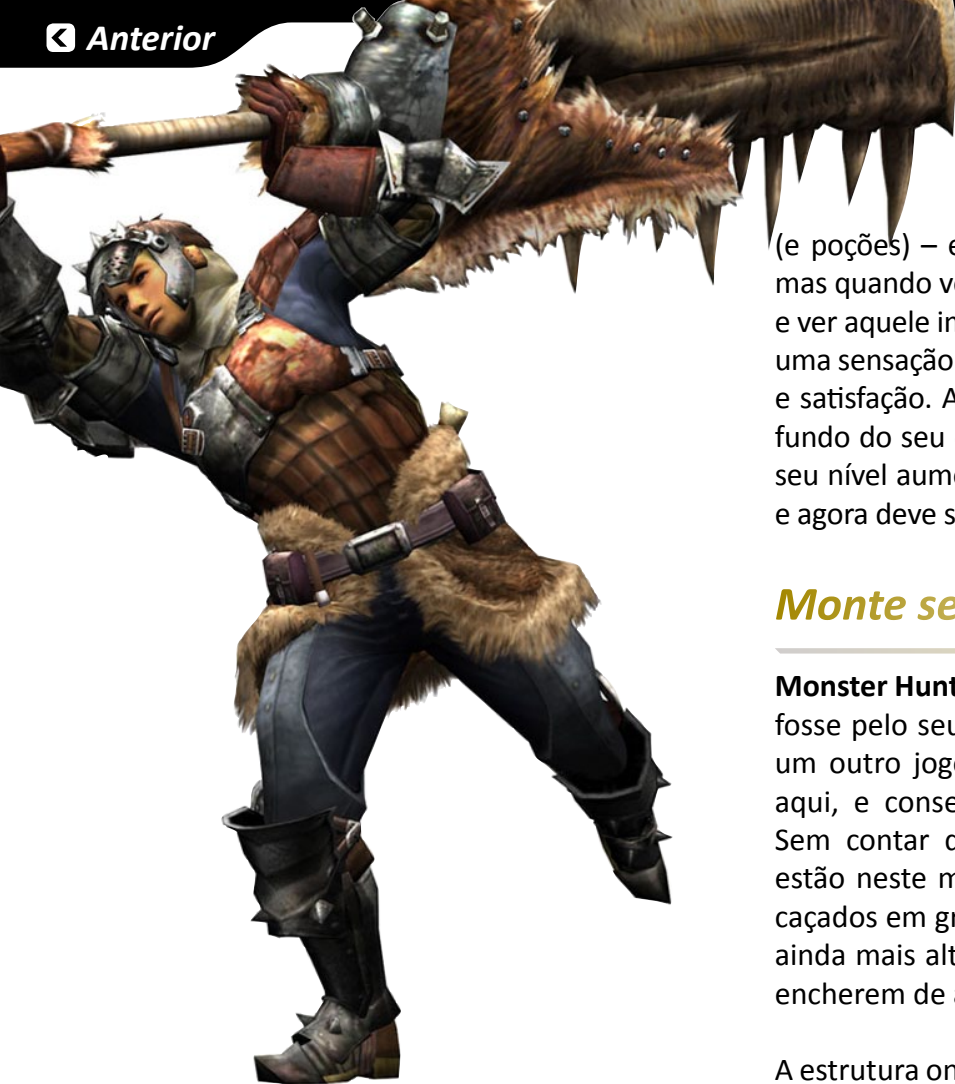
Criaturas realistas, ambientes impressionantes e uma sensação incrível

Os ambientes no jogo são os mais variados possíveis, principalmente com a adição de poder jogar dentro d'água, e todos muito impressionantes – provavelmente os melhores até o momento para o Wii. Ao andar pelos diversos locais é possível presenciar criaturas difíceis, que fogem quando uma criatura feroz aparece na região, ou então as mais agressivas, que te observam por um tempo até decidirem o melhor momento para te atacar, tudo isso animado com um realismo estonteante.



Ao atacar as criaturas, algumas revidarão até ficarem fracas e tentarem fugir. Alguns dos chefes também mostram sinais de fraqueza e fogem para diferentes regiões do mapa quando estão com problemas. Outros, como o Rathalos, voam para um local com monstros menores para se alimentar deles e recuperar a força. Uma coisa é certa, a Capcom conseguiu algo incrível ao criar toda essa inteligência artificial orgânica, transformando cada uma das criaturas em seres reais, com instintos próprios que variam de acordo com a situação.

Mas nada consegue ser equivalente ao encontro com os chefões, as maiores e mais terríveis criaturas que um caçador já viu. Você arregala os olhos, pois sabe que nesse momento é preciso focar. Seu coração começa a bater mais rápido – o monstro ainda não te viu, então há tempo de colocar uma ou duas armadilhas e se preparar melhor. Até que você é avistado e dá-se início a uma batalha épica. Tudo deve ser calculado, o monstro tem que ser previamente estudado, os seus movimentos precisam ser precisos, e qualquer falha é brutalmente punida. Uma simples batalha contra um chefe pode durar mais de 40 minutos de muito esforço



(e poções) – e acredite, as chances de falhar são altas –, mas quando você finalmente consegue atingir seu objetivo e ver aquele imenso dragão cair estrondosamente no chão, uma sensação incrível de vitória enche sua alma de orgulho e satisfação. A vibração é real, você sente a superação no fundo do seu coração. Nada de numerozinho dizendo que seu nível aumentou. Você sabe que evoluiu internamente, e agora deve se preparar para o próximo desafio.

Monte seu grupo de caçadores

Monster Hunter Tri não seria o incrível título que é se não fosse pelo seu maravilhoso modo online. É praticamente um outro jogo, com novas criaturas acessíveis somente aqui, e conseqüentemente, novas armas e armaduras. Sem contar que os monstros mais poderosos do jogo estão neste modo, e só podem ser derrotados se forem caçados em grupo, o que significa um nível de dificuldade ainda mais alto, que fará os jogadores mais dedicados se encherem de ânimo.

A estrutura online usada pela Capcom foi muito bem feita. Não é necessário nenhum Friend Code ou algo do gênero.

Monster Hunter Tri possui um sistema próprio onde você pode entrar em servidores públicos e encontrar caçadores que estejam no mesmo rank que o seu, podendo até mesmo conversar com eles usando o WiiSpeak, ou simplesmente por mensagens de texto caso você tenha um teclado USB sobrando. Você pode adicionar outros caçadores na lista de amigos interna e se teleportar para a sala deles, criando um ambiente fácil e simples para você encontrar seus amigos e retalhar alguns dragões juntos.



Como dito anteriormente, as quests online são diferentes do modo single-player, onde você aumenta o seu rank de caçador e habilita novas quests com criaturas mais poderosas. Isso significa que há um conteúdo muito grande no jogo, que passa com facilidade centenas de horas de muita aventura, planejamento, e satisfação. Até mesmo os NPCs vendedores agem de forma diferente no modo online, oferecendo itens únicos ou combinações antes não existentes.

E, se por um acaso, seu Wii não pode conectar online, **Monster Hunter Tri** conta com um modo split-screen onde você e um amigo entram em quests especiais, cada um com

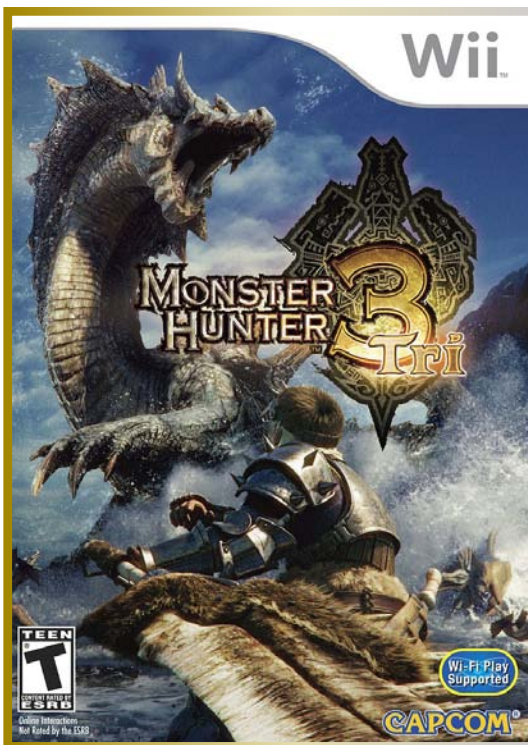


armas e armaduras pré-montadas para vencer os chefões. Isso pode ser útil simplesmente para treinar, ou ensinar o funcionamento do jogo para um amigo ainda novato.

Considerações finais

Monster Hunter Tri não é para qualquer um. Apesar de uma curva de aprendizado menor, é um título que exige paciência e muita dedicação do jogador – algo que o público casual do Wii pode não se sentir confortável. Os desafios que são apresentados podem até parecer impossíveis em certos momentos, mas com o planejamento certo, você será capaz de matar as mais ferozes e imensas criaturas, que rendem uma sensação de satisfação incomparável. Seja no single player ou online, a recompensa que é oferecida certamente vale todo o seu esforço. Novos desafios, armas e equipamentos farão com que você prossiga no objetivo de derrotar bestas cada vez maiores, aproveitando cada momento de glória no processo.

Para aqueles que preferem jogos que lhe recompensam pela sua habilidade e que oferecem ampla opção de customização, seja nos diferentes modos de jogabilidade com as armas, ou simplesmente no vasto conteúdo oferecido, este é o fim da sua procura. Esse é o melhor **Monster Hunter** já feito, e mesmo que boa parte do jogo seja familiar aos fãs da franquia, ele continua sendo um título de experiência única no Wii.

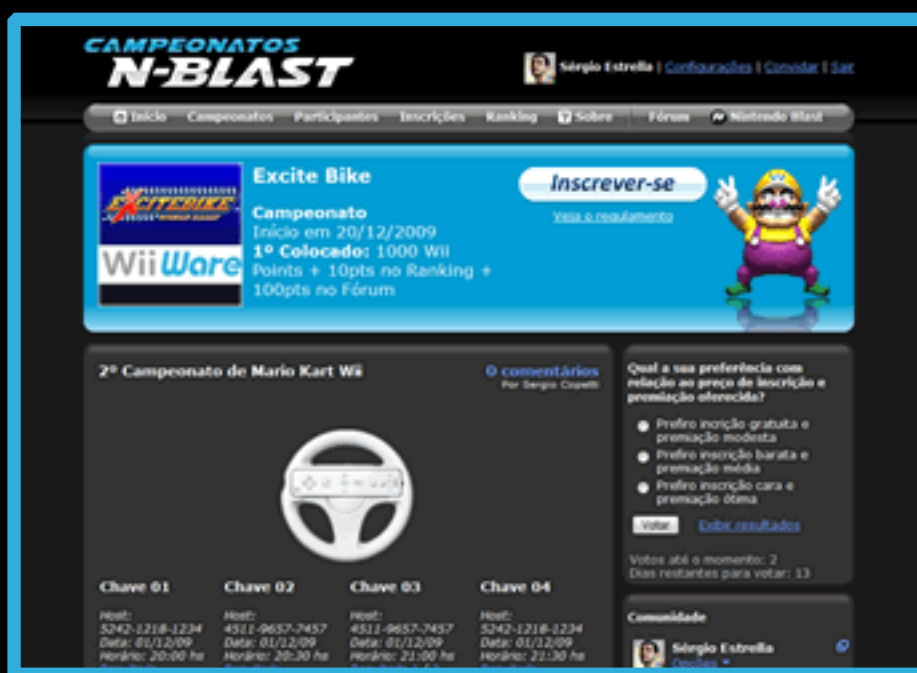


9.5

Monster Hunter Tri (Wii)

GRÁFICOS 10 | SOM 9.0
JOGABILIDADE 9.5 | DIVERSÃO 10

CAMPEONATOS N-BLAST



campeonatos.nintendoblast.com.br

Gosta de campeonatos online?

Então está em casa! O Nintendo Blast orgulhosamente apresenta o Campeonato N-Blast, sua comunidade de jogos online Nintendo. Esse projeto é o resultado da integração da bem sucedida Comunidade Wii Play (que agora passa a se chamar Campeonato N-Blast) com o Nintendo Blast, trazendo para você leitor – e jogador – algo inédito no Brasil para consoles Nintendo, aliada ao nome de um site de games já conceituado.

Análise

Dementium II (DS)

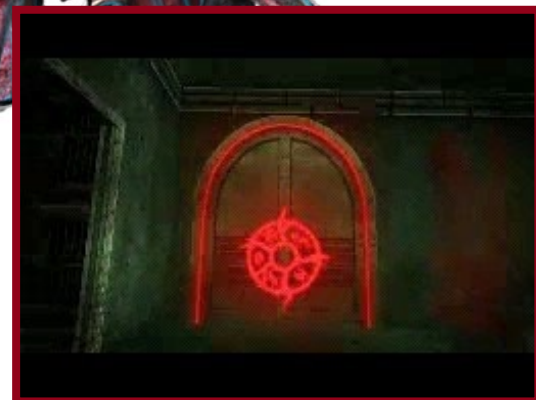
Lançado em 2007, Dementium the Ward foi um inesperado sucesso no DS. Conseguindo oferecer bom visual, trama muito interessante e jogabilidade surpreendente o game deu uma sobrevida para os títulos do gênero no console. Agora chegou a vez de saber como se saiu a continuação, Dementium II, que acaba de aportar no console da Nintendo.

Games de FPS bons no DS são extremamente raros. A tela limitada e a falta de botões do portátil acabam prejudicando a jogabilidade de forma decisiva, tornando os títulos do gênero limitados e problemáticos. Esse (felizmente) não foi o caso de Dementium, e não é o caso de Dementium II.

Desenvolvido pela Renegade Kid (que já havia criado o ótimo Moon), Dementium II é um game que conta com todos os elementos de um bom título do gênero. Para quem não sabe, antes de ser um FPS, Dementium é um game de terror, com uma trama extremamente bizarra e muito mais viçada do que a do primeiro game da série.

Terror de filme B, sustos de menos

Dementium II se passa no Centro de Tratamento Bright Dawn, um lugar calmo e tranquilo criado pelas cabeças mais pensantes da América. Tranquilo e calmo pelo menos para quem não é admitido no local. Métodos extremos, amarras e injeções de substâncias misteriosas são apenas a ponta do iceberg da crueldade com que os pacientes são tratados.



O jogador controla um dos pacientes do Bright Dawn que já sofre transtornos decorrentes do tratamento que vem recebendo. Tanto que sua cabeça visualiza alucinações, dando a idéia de dois mundos: um de repressão (Bright Dawn) e outro de puro terror (Hell Land). É em Hell Land que as coisas realmente ficam imprevisíveis... e você vai perceber que apesar do quê de Silent Hill, essa ambientação não funciona tão bem assim...

Para um game com foco nos sustos e em uma trama aterrorizante, o que dá pra adiantar é que os objetivos não foram totalmente alcançados. A trama não possui qualquer relação com o primeiro Dementium, as passagens da narrativa são confusas e pouco inspiradas e a todo momento o sentimento é de que o enredo poderia ter ganhado um trabalho mais minucioso. As informações são jogadas sem muito cuidado, deixando o jogador meio perdido com relação ao que está acontecendo.

Uma crítica importante que pode ser feita é que dessa vez o game utiliza cenários mais claros e não há mais a necessidade do uso da lanterna (como acontecia no primeiro Dementium), o que claramente fez com que o título se perdesse e acabasse se tornando muito menos assustador e impactante. O design dos monstros e inimigos beirando o grotesco também contribui para que o clima assustador seja deixado de lado. Resumindo: o game é mais violento que assustador, embora esse não tenha sido o objetivo inicial.

Tecnicamente perfeito

Se a concepção visual de Dementium II deixou a desejar, isso não é culpa da parte técnica. Talvez esse seja um dos games mais bonitos do DS do ponto de vista do uso das capacidades do portátil. O game roda a 60 fps constantes, possui construções poligonais muito interessantes, efeitos de luz surpreendentes e cenários repletos de detalhes. Mais um grande trabalho da Renegade Kid, sem sombra de dúvidas.

O que parece não seguir essa fórmula é a previsibilidade da trilha sonora e dos efeitos sonoros, que em nada contribuem para diferenciar o game de outros títulos do gênero. O trabalho de voz é confuso e parece deslocado. Efeitos sonoros e a trilha sonora ainda cumprem bem o seu papel, mas ficam abaixo de outros trabalhos em games do gênero. O que merece muitos elogios são os controles e a



jogabilidade em si. Apesar de básico, o controle da movimentação na tela de toque é bastante funcional e a presença de um mapa na tela inferior e movimentos de esquiva contribuem muito para o balanceamento da estrutura de jogo.

Mas há dois problemas mais sérios. Um deles é recorrente do primeiro Dementium: o sistema de save. Se no original a cada nova área o game era salvo automaticamente, aqui a presença de save points foi reduzida ao extremo, causando um grande desconforto ao jogador. O segundo defeito é a falta de enigmas mais profundos. Tudo parece meio repetitivo após a metade da jornada, deixando uma sensação de que as ideias da equipe de desenvolvimento se estagnaram.

Se o primeiro Dementium surpreendeu, o mesmo não acontece com essa continuação que, apesar de manter o padrão técnico de qualidade, não consegue oferecer nada de novo – e nem ao menos repete a inspiração do primeiro. Uma pena.



7.0

DEMENTIUM II - NINTENDO DS

GRÁFICOS 8.5 | SOM 7.0

JOGABILIDADE 8.0 | DIVERSÃO 7.0

SIN & PUNISHMENT

STAR SUCCESSOR

por Gustavo Assumpção

O primeiro Sin and Punishment passou quase despercebido pelo Nintendo 64. Lançado no fim de 2000, época em que o console dava seus últimos suspiros, o game se tornou uma grande surpresa, agradando boa parte da crítica especializada e os poucos jogadores que desfrutaram da experiência que o título ofereceu. Agora é hora de se preparar para Star Successor, continuação com lançamento marcado para o próximo mês, que você confere em detalhes agora:

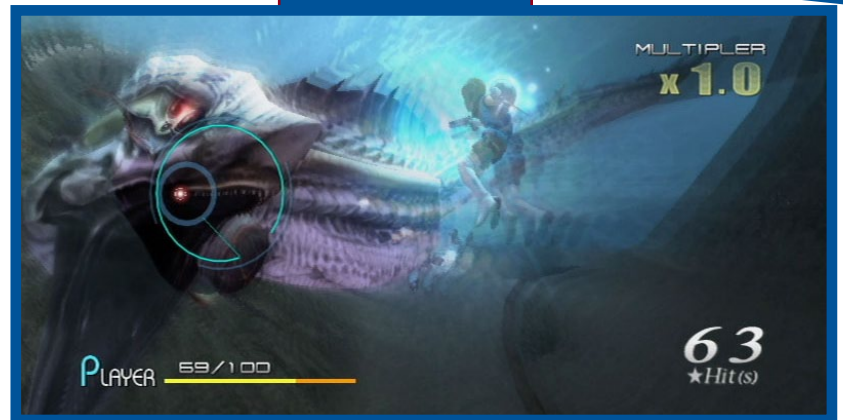
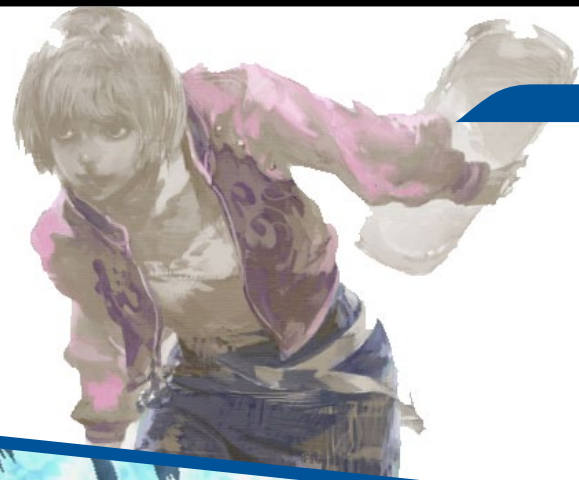
Sin and Punishment é mais um game diferenciado, fruto da mente criativa e pulsante da Treasure. Idealizadores de grandes obras-primas como Radiant Silvergun (Saturn), Bangai-O (Nintendo 64 e Dreamcast), Gunstar Heroes (Mega Drive), Silhouette Mirage (Playstation), Alien Soldier (Mega Drive) e Ikaruga (GameCube e Dreamcast), a equipe é uma das mais respeitadas quando o assunto são games de tiro desafiadores - e olha que muitas vezes esse desafio é tão grande que beira a deslealdade.

Mas dentre todos esses grandes games, acho que nenhum atingiu o nível de excelência de Sin and Punishment, lançado em 21 de novembro de 2000 para o Nintendo 64 em terras japonesas. Dirigido por Hideyuki Suganami (de Buck'ô Hare e Radiant Silvergun), o game atingiu um nível técnico incrível para a época.

Mas talvez o grande trunfo do game tenha sido sua ambientação incrível de uma Tóquio surrealista e irreconhecível. Num futuro próximo, a capital japonesa enfrentou um crescimento vertiginoso de sua população, levando a uma crise de alimentos. Nesse cenário, cientistas tentam criar em laboratório uma raça exclusivamente para a alimentação humana. Só que é lógico que algo sai errado; essa tal raça se volta contra seus criadores e começa a atacar a população da capital japonesa. Lutas políticas, conspirações e muita disputa por poder acabam entrando em jogo nesse momento de intensas emoções.

Um degrau acima

Se o primeiro game era grandioso e ousado, Star Successor é um salto considerável de qualidade, profundidade e grandiosidade para a série. O desenvolvimento, mais uma vez, ficou nas mãos da Treasure, só que agora a Nintendo acompanhou de perto o processo, sendo responsável pela distribuição do game ao redor do mundo.



Nessa continuação, o jogador vai controlar Isa (o filho de Saki e Airan, os personagens principais do primeiro jogo) e uma menina chamada Kachi. A trama relembra momentos do original e a história é muito mais tecnicista, elemento que se reflete no design dos personagens e ambientes. Engraçado é que agora a história é muito mais tecnicista, elemento que se reflete no design de personagens e ambientes.

Ambientes que são gigantescos, muito bem trabalhados e com efeitos de explosão, luz e animações bem acima da média – características que inclusive já tinham sido um grande destaque do primeiro Sin and Punishment. Vale destacar ainda a fluência que a Treasure conseguiu para esse visual. O game roda em 60 fps constantes, logo, nada de serrilhados ou slowdowns em meio à toda a ação.

Rail shooter sim, com muito orgulho

A jogabilidade de Sin and Punishment pode ser resumida em uma palavra: desafio. A movimentação dos personagens e da câmera acontece em um esquema pré-determinado – o super falado rail shooter. Porém, resumi-lo a esse tipo de experiência é errôneo, pois há muita variedade de ações, movimentações e angulações de câmera, levando o gênero a um outro nível.

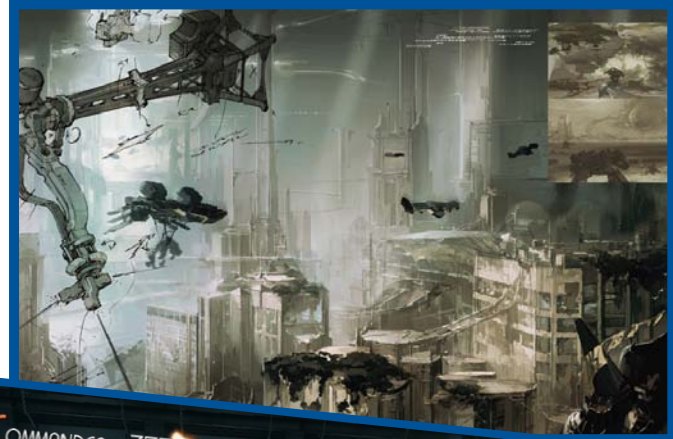
Atingir os inimigos é tarefa das mais fáceis: basta apontar o Wiimote para a tela e disparar toneladas de tiros para cima deles. Mas enganase quem pensa que a jogabilidade é apenas apertar botões rapidamente. Se você adotar esse estilo de jogo, as chances de êxito são pequenas. É preciso dosar a hora certa de atirar, saber se esquivar e ser ágil e preciso nas decisões. Lembra aqueles games que tanto nos encantavam em épocas passadas, só que ampliado pelas possibilidades que a geração atual permite.

PRÉVIA



Existem algumas funcionalidades bastante interessantes. Por exemplo, com o botão B é possível travar a mira em um inimigo, embora isso cause uma perda de potência nos tiros. A presença do jetpack além de permitir o jogador planar durante as fases, também adiciona uma certa variedade aos controles, otimizando o uso do espaço na tela. Essa característica chegava a ser um problema no primeiro game da série e foi solucionada nessa continuação.

Assim como todo game de ação, os chefes possuem um papel crucial. Gigantescos e baseados no esquema de possuírem um ponto fraco, cada um deles tem relação com a fase determinada em que o jogador está naquela hora. Star Sucessor ainda possui suporte ao Classic Controller, o que por si só é uma grande adição – não que precise, já que a jogabilidade original funciona muito bem. Há a presença de um modo cooperativo para duas pessoas que apesar de simples, pode ser uma boa opção para superar os desafios absurdos que o game propõe com um ajudante.



Se você procura um game realmente hardcore no Wii, talvez esteja próximo de alcançar esse objetivo. O lançamento de Sin and Punishment: Star Sucessor está marcado para 26 de junho.



Monster Hunter G (Wii)



Dá até para apostar que a maioria dos leitores da revista está ocupada demais jogando o belíssimo e incrível *Monster Hunter Tri* - e enquanto jogam, se perguntam: “minha nossa! Como um jogo tão bom quanto esse ainda não existia?” A verdade é que um jogo desse tipo existiu antes sim, desde o dia 23 de abril de 2009.

Como alguns já suspeitam, estou falando de *Monster Hunter G*, um título exclusivo do Japão que foi lançado em 2005 para o PlayStation 2. Opa, espera aí, Playstation 2? Isso mesmo. E somente em 2009 é que a Capcom percebeu o tamanho do sucesso que a franquia estava fazendo no Japão. Sucesso o suficiente para relançar uma versão do título para o Wii.

Monster Hunter G funciona basicamente como uma expansão do *Monster Hunter* original de PlayStation 2. Ou seja, estamos falando de um hack and slash, com pitadas de aventura e estratégia até não poder mais.

O que é, e quais são as diferenças?

Para aqueles que não estão familiarizados com o estilo do jogo, em *Monster Hunter* você cria o personagem com o qual irá se aventurar num mundo habitado por diversos dinossauros e dragões. O modo singleplayer conta com uma história para acompanhar, mas geralmente não é tão importante, servindo apenas para entender direito o mecanismo do jogo. O título encontra seu ápice no modo online, onde você se reúne com até três companheiros para dizimar as criaturas mais gigantes enquanto usam diversas armas e estratégias extremamente complexas.

Monster Hunter G, para o Nintendo Wii, saiu um pouco tarde e, apesar de ser considerado uma expansão do primeiro título – cujo qual que teve seu lançamento no Ocidente –, contou com várias novidades, que inclusive ditaram as novas tendências que foram usadas em *Monster Hunter Tri*, por exemplo. Estamos falando de criaturas de mesma espécie, porém com cores e tamanhos diferentes; o novo ranking de missões extremamente difíceis, chamado G (daí o *Monster Hunter G*); a possibilidade de contratar Felynes cozinheiros que aumentam (ou diminuem) os atributos através de refeições e, principalmente, um sistema de habilidades, baseado em números singulares para cada peça de armadura, ao invés de contar somente quando o set estava completo, o que demorava muito tempo. E olha que essas são só algumas novidades.



Aceitação Nipônica

É curioso ver como Monster Hunter G vendeu bem no Japão. Como dito no começo deste artigo, seu lançamento foi realizado dia 23 de Abril de 2009. Ou seja, pouquíssimo tempo antes do lançamento de Monster Hunter Tri no Japão, que aconteceu no dia 1º de Agosto do mesmo ano. Foram 200 mil unidades vendidas somente nas primeiras semanas, alcançando no fim a marca de 250 mil unidades. Isso levando em consideração que o Tri sairia poucos meses depois.

Na verdade, o título foi vendido num bundle que trazia consigo a demo de Monster Hunter Tri e o controle Classic, o que o ajudou a alavancar no mercado, gerando até expectativas de que uma versão pudesse ser lançada no Ocidente, mas o que parece pouco provável.

Monster Hunter G redefiniu e aprimorou o mecanismo da franquia de forma a influenciar diretamente nos outros jogos da série, inclusive em Monster Hunter Tri. Ele obteve um número de vendas consideravelmente bom, ainda por cima levando em consideração que era apenas um port da versão de PlayStation 2, lançado anos atrás e, infelizmente, parece ser mais um título de Wii que entra na lista dos que não jogaremos por aqui.





GameDev

PARTE 7: CONCEITOS DE PROGRAMAÇÃO DE JOGOS NO XNA

Por Sérgio Oliveira

Nos nossos últimos dois encontros aqui na coluna *GameDev* do *Nintendo Blast* conhecemos melhor o XNA, suas camadas, vantagens e possibilidades. A partir de hoje aliaremos a teoria à prática e começaremos a utilizar as ferramentas do XNA, bem como escrever e, principalmente, compreender as linhas de código. Apesar de tomar “ares” de tutorial, não deixaremos de abordar e explicar os aspectos técnicos sobre o desenvolvimento de um jogo com XNA em nossa coluna.

Na coluna de hoje iremos criar um projeto XNA em branco, analisar sua estrutura e entender os conceitos básicos por detrás do jogo.

Para acompanhar a coluna de hoje e aproveitá-la em sua magnitude, precisaremos fazer o download e instalar a última versão do XNA Game Studio e o Microsoft Visual C# Express Edition, ambos gratuitos, que podem ser encontrados na sessão de downloads do XNA Creators Club (<http://creators.xna.com/en-US/downloads>).

Após feito o download dos dois, instale primeiramente o Visual C# Express Edition e logo em seguida o XNA Game Studio. Ao longo das próximas colunas eu utilizarei uma versão diferente do Visual C# que na verdade é o Visual Studio Team System 2008 (versão paga e licenciada) que, para a nossa coluna, faz as mesmas coisas.



O primeiro contato com o Visual Studio

Após as devidas instalações, abra o seu Visual C# Express Edition e uma janela parecida com a Imagem 1 deverá aparecer. Clique em **File->New->Project** (ou aperte Ctrl+Shift+N) para criar um novo projeto. Na janela que aparecer, selecione a opção **XNA Game Studio 3.1** e em templates selecione **Windows Game (3.1)** (Imagem 2).

No campo name você definirá qual o nome desse seu projeto e em Location o nome da pasta onde seu projeto ficará salvo - escolha qualquer nome e a pasta que você quiser para isso e dê OK.

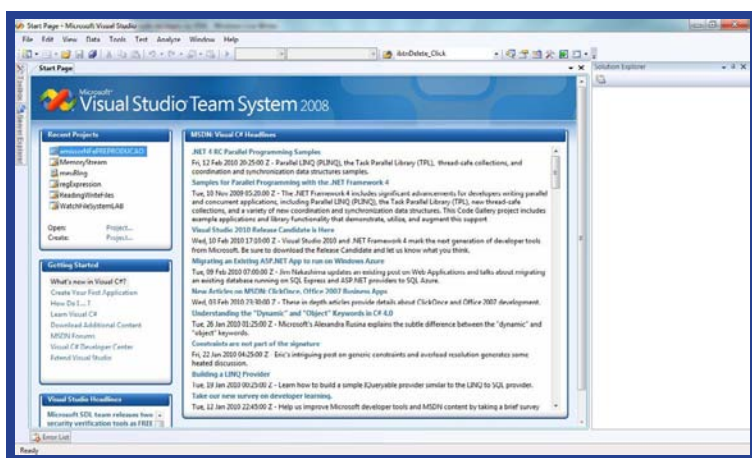


Imagem 1

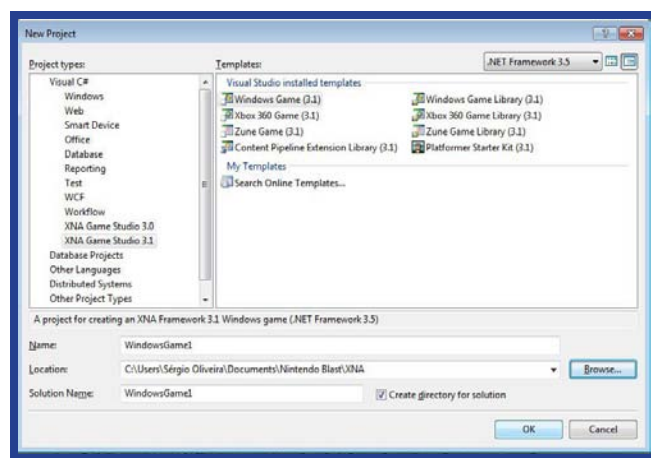
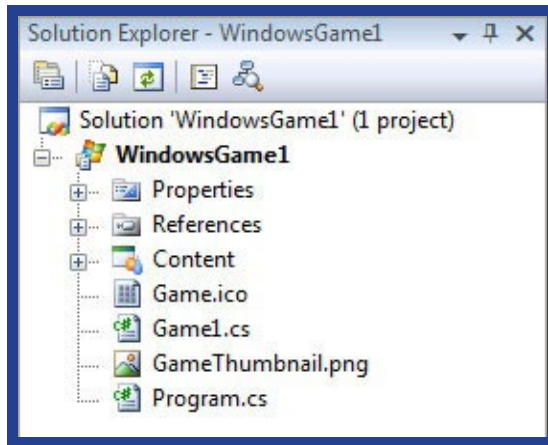


Imagem 2

Após o projeto ter sido criado, aperte F5 no seu teclado para compila-lo e colocar o jogo para funcionar. É claro que não aparecerá jogo nenhum – ao invés disso aparecerá uma entediante tela azul que indica que você está pronto para desenvolver o seu primeiro jogo com XNA. Feche a janela azul e volte ao Visual C# Express Edition e perceba



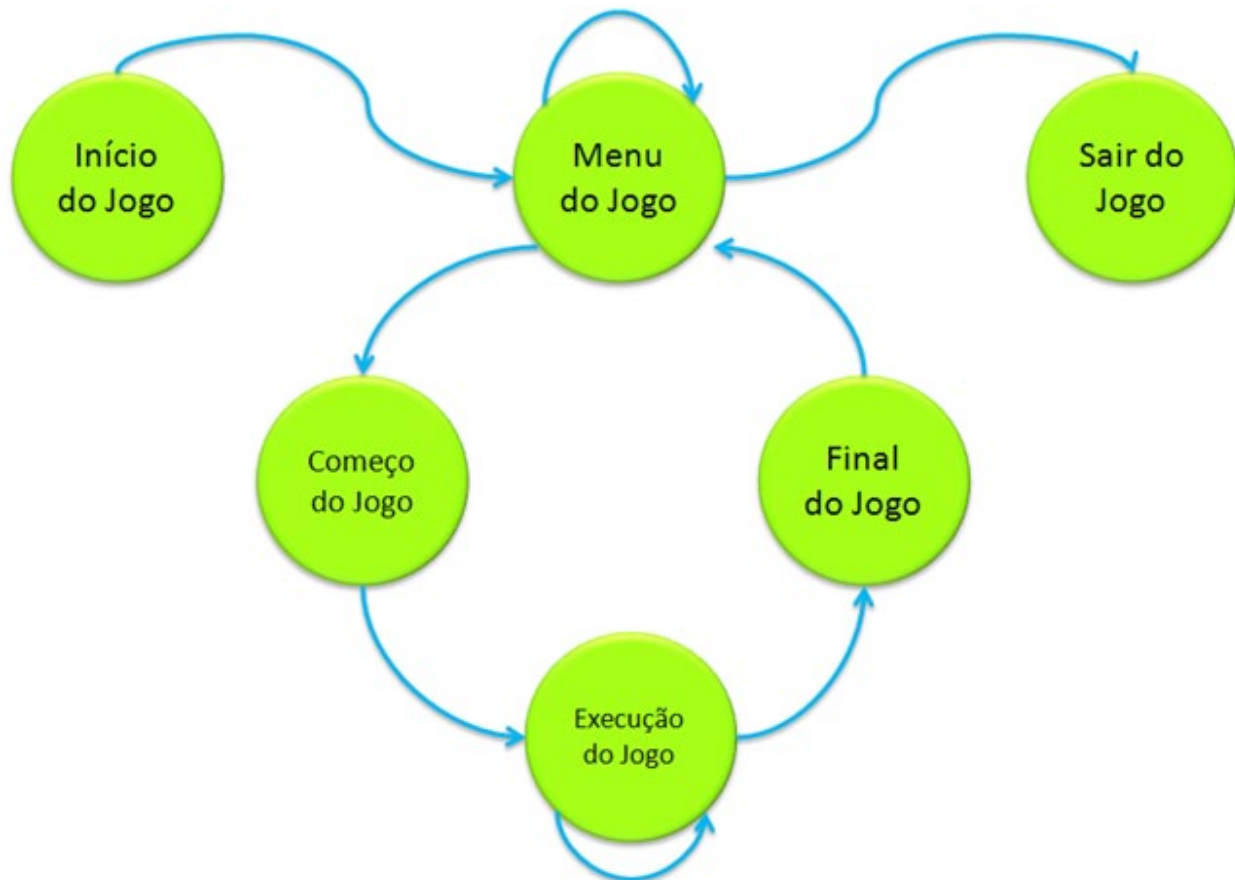
que, à direita, a aba do **Solution Explorer** foi “povoada” com vários arquivos e pastas (Imagem 3).

Perceba que com os arquivos de imagem **Game.ico** e **GameThumbnail.png**, seu projeto foi criado com dois arquivos de extensão .cs (**Program.cs** e **Game1.cs**) que são arquivos de código C#. Também foi criada uma pasta chamada **Content** que armazenará todo o conteúdo do jogo: sons, imagens, modelos 3D, texturas e por aí vai.

Antes de começarmos a entender a estrutura dos arquivos de código do XNA, vamos recapitular um pouco a estrutura básica de um jogo.

Estrutura genérica de um jogo

No nosso último encontro apresentamos um fluxograma básico sobre como um jogo funciona e citamos que uma vez dentro dele, este fica “dando voltas” dentro de si próprio verificando se cumprimos todos os objetivos para alcançarmos o final e, finalmente, saímos do jogo de uma vez por todas. Agora teremos essa visão de um lado mais técnico da coisa.



A lógica central de qualquer jogo inclui a preparação do ambiente onde ele será executado, a execução do jogo em loop até que o critério de fim dele seja alcançado e, finalmente, a limpeza desse ambiente. Se formos representar tecnicamente essa estrutura, podemos construir o seguinte pseudocódigo:

- Inicialização dos gráficos, dispositivos de entrada de dados e som
- Carregar os recursos
- Iniciar o loop do jogo. A cada loop:
- Capturar os comandos do jogador
- Executar os cálculos necessários (IA, movimentos, detecção de colisões, etc.)
- Testar se o critério de fim de jogo foi alcançado – em caso positivo, o loop para
- Renderizar gráficos em tela, gerar sons e respostas aos comandos do jogador
- Finalizar os gráficos, dispositivos de entrada e som
- Liberar os recursos (placa de vídeo, memória e CPU)

Claro, essa é uma visão bastante simplista, mas que já nos dá uma boa visão de como devemos “construir” o código do nosso jogo. Antes do XNA, os desenvolvedores não se preocupavam somente com essa estrutura, mas também em como tudo isso iria funcionar – todos os mínimos detalhes que não faziam parte do projeto de desenvolver o jogo em si deveriam ser pensados também.

Se voltarmos ao Visual C# Express Edition e observarmos atentamente os arquivos **Game1.cs** e **Program.cs** perceberemos que eles apresentam métodos que nos permitem afirmar que a estrutura dos jogos em XNA é bastante similar a estrutura do pseudocódigo que mostramos anteriormente. Sem entrar em maiores detalhes por enquanto, temos o seguinte pseudocódigo nos dois arquivos:

- **Game1()** – Inicialização geral (Game1.cs)
- **Initialize()** – Inicialização do jogo (Game1.cs)
- **LoadContent()** – Inicializa e carrega recursos gráficos (Game1.cs)
- **Run()** – Inicia o loop do jogo (Program.cs). A cada loop:
- **Update()** – Captura os comandos do jogador, realiza cálculos e testa o critério de fim de jogo (Game1.cs)
- **Draw()** – Renderiza os gráficos em tela (Game1.cs)
- **UnloadContent()** – Libera os recursos gráficos

A inicialização do jogo

Observemos o seguinte pedaço de código:

```
public class Game1 : Microsoft.Xna.Framework.Game
{
    GraphicsDeviceManager graphics;
    SpriteBatch spriteBatch;
    public Game1()
    {
        graphics = new GraphicsDeviceManager(this);
        Content.RootDirectory = "Content";
    }
}
```

Nele podemos perceber que a classe **Game1** está sendo definida e criada, tendo como classe pai a **Microsoft.Xna.Framework.Game**, que nos permitirá trabalhar com gráficos, sons, controles e cálculos de lógica para o nosso jogo.

Logo em seguida definimos e inicializamos dois objetos do tipo **GraphicsDeviceManager** e **SpriteBatch** – esse nos permitirá desenhar textos e imagens 2D na tela do nosso jogo, enquanto aquele nos dará acesso aos recursos do

gerenciador de dispositivos gráficos para que trabalhemos com os gráficos do nosso jogo. Além disso, ainda temos a definição da pasta raiz onde ficará todo o conteúdo do nosso jogo – isso é importante pois é a partir daqui que teremos acesso ao gerenciador de conteúdo do nosso jogo.

Observação: daqui para frente entraremos em um linguajar mais técnico da coisa. Sempre que possível, tentarei esmiuçar o palavreado e explicar da forma mais simples possível. Entretanto, é importante que haja um conhecimento básico sobre programação e o paradigma de Orientação a Objetos. Caso se interessem pelo assunto de verdade, aconselho o download e leitura de algumas apostilas de algoritmos, técnicas de programação e orientação a objetos.

O Graphics Device Manager (gerenciador de dispositivo gráfico)



Como seu nome sugere, o gerenciador de dispositivo gráfico nos permitirá lidar diretamente com a camada gráfica do nosso dispositivo. Ele inclui métodos, propriedades e eventos que nos permitem consultar e alterar essa camada. Trocando em miúdos, é o gerenciador que nos dará acesso aos recursos da nossa placa de vídeo.

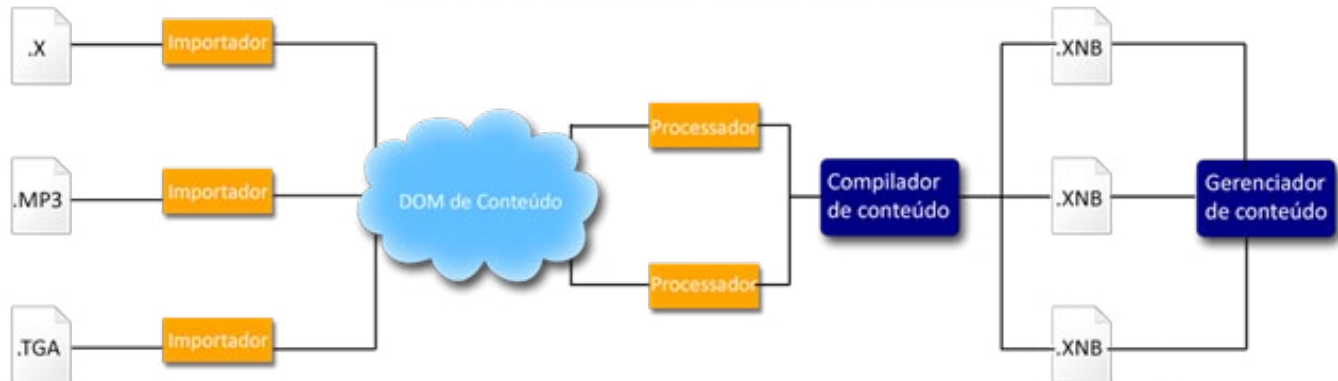
Mais uma vez, antes do XNA havia a preocupação de programar a placa de vídeo para que essa trabalhasse corretamente com o jogo. Com a classe `GraphicsDeviceManager` do XNA, toda a preocupação, complexidade e detalhes de se trabalhar com uma placa de vídeo se esvai. Por ora, o que precisamos ter em mente é que sempre utilizaremos o objeto criado a partir da classe `GraphicsDeviceManager` para realizar qualquer operação gráfica.

O Content Pipeline (gerenciador de conteúdo)

O gerenciador de conteúdo é um dos recursos mais interessantes do XNA. A forma como foi desenvolvido permite que importemos para o jogo qualquer tipo de conteúdo de ferramentas diferentes. Em jogos que não utilizam XNA os desenvolvedores precisam se preocupar em como carregar tais conteúdos, onde esse conteúdo está armazenado, se existem as bibliotecas corretas para importação e execução e mais uma série de dores de cabeça desnecessárias.

O gerenciador de conteúdo do XNA dá conta de todo o recado, fazendo com que esse doloroso processo se torne bastante agradável. Basicamente o que ele faz é importar os conteúdos, processá-los com o compilador de conteúdo e gerar um arquivo de conteúdo (extensão .XNB) que será utilizado pelo jogo.

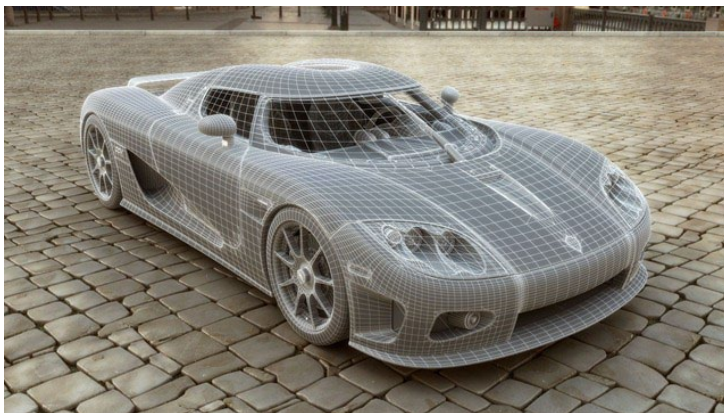
FLUXOGRAMA DO CONTENT PIPELINE



O mais interessante é que, apesar de abranger uma enorme quantidade de formato de arquivos, você ainda poderá desenvolver seus próprios compiladores para trabalhar com um conteúdo específico de alguma ferramenta que o XNA ainda não dê suporte (inclusive já existe uma boa quantidade dessas bibliotecas no XNA Creators Club – <http://creators.xna.com>). Discutiremos mais sobre isso em outros encontros.

Os métodos de inicialização do jogo

Se voltarmos um pouco ao começo desse artigo e analisarmos o pseudocódigo que contém os métodos gerados e utilizados pelo XNA, perceberemos que temos duas inicializações do jogo e que cujos propósitos ainda não foram devidamente explicados. Antes de começarmos a trabalhar no nosso jogo, precisamos entender o porquê dessas duas rotinas de inicialização.



O método **Initialize()** é chamado apenas uma vez quando o método **Run()** (que inicia o loop do jogo) é executado. É no método Initialize() que devemos trabalhar as rotinas de inicialização de componentes não-gráficos do nosso jogo, como, por exemplo, preparar o conteúdo de áudio.

Os gráficos precisam ser carregados em um método diferente porque às vezes o jogo precisa recarregá-los. Para que o jogo rode em ambiente ótimo, os gráficos são carregados de acordo com as configurações da sua placa de vídeo. Quando as configurações da placa são alteradas, a resolução mudada, ou coisas desse tipo, os gráficos precisam ser recarregados. Como o método **Initialize()** é chamado apenas uma vez, antes do loop do jogo ser iniciado, os gráficos são inicializados no método **LoadContent()**, que é chamada toda vez que o jogo precisar carregar ou recarregar os gráficos.

O loop do jogo

A maior parte do processamento do jogo acontece dentro do seu loop. É no loop que o jogo verifica os comandos do jogador, processa-os e dá o feedback no personagem do jogo, calcula a inteligência artificial, os movimentos são calculados e executados, as colisões entre objetos detectadas, a vibração do controle é ativada, o som é tocado, os gráficos desenhados na tela e os critérios para alcançar o fim do jogo checados. Enfim, praticamente tudo ocorre no loop do jogo.

No XNA, o loop do jogo depende de dois métodos derivados da classe **Microsoft.Xna.Framework.Game: Update()** e **Draw()**. No **Update()** são incluídos todos os cálculos do jogo, enquanto no **Draw()** você desenha os componentes do jogo em tela. Para compreender melhor, observemos os dois métodos:

```
protected override void Update(GameTime gameTime)
{
    if (GamePad.GetState(PlayerIndex.One).Buttons.Back == ButtonState.Pressed)
        this.Exit();
    base.Update(gameTime);
}

protected override void Draw(GameTime gameTime)
{
    graphics.GraphicsDevice.Clear(Color.CornflowerBlue);
    base.Draw(gameTime);
}
```

Em relação as demais classes que já vimos, o que primeiro podemos denotar é que tanto o método Update(), quanto o Draw() recebem o parâmetro gameTime. No momento, o que precisamos saber é que esse parâmetro é crucial para a lógica do jogo. A partir dele é que o nosso jogo saberá quanto tempo passou desde que o último loop foi executado. Isso é extremamente importante para que possamos fazer os cálculos corretos – por exemplo, calcular a posição correta dos componentes do jogo de acordo com suas velocidades no jogo. Sem esse parâmetro, seria impossível organizar e desenvolver o jogo.

Outra coisa a ser observada é que, no método Update(), há um código pré-definido para sair do jogo quando se aperta o botão “Back” no controle do Xbox 360:

```
if (GamePad.GetState(PlayerIndex.One).Buttons.Back == ButtonState.Pressed)
    this.Exit();
```

A classe GamePad nos permite acessar o controle e permite ativar a vibração dele. A captura dos comandos dados pelo usuário é em tempo real e não há nenhum atraso quanto a isso. Se observarmos atentamente, podemos ver que o trabalho com a interface do controle é bem simples e intuitiva, pois as propriedades são auto-explicativas:

- **GamePad:** classe de acesso ao controle
- **GetState:** pegue o estado do controle
- **PlayerIndex.One:** apenas do primeiro controle
- **Buttons:** acessa os botões do controle
- **Back:** especificamente o “Back”
- **ButtonState.Pressed:** quando o botão for pressionado
- **this.Exit():** sair do jogo

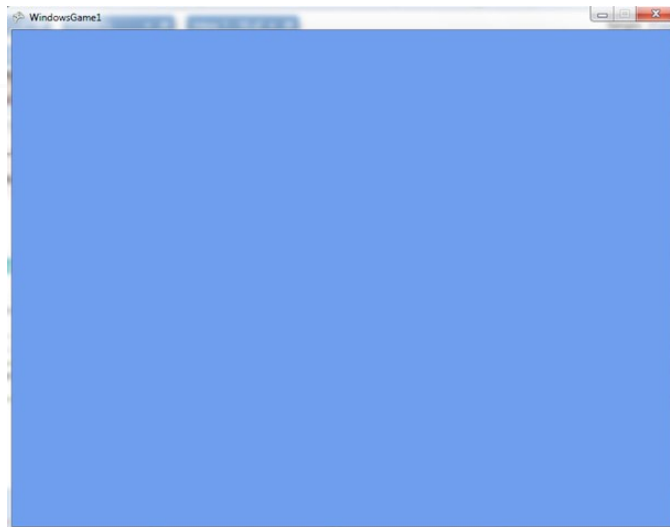


Logo, sem nenhuma dificuldade, podemos dizer que esse código está dizendo ao jogo que “quando o botão Back do controle 1 for pressionado, saia do jogo”. Simples, não?

Já o método `Draw()` inclui uma linha para limpar o dispositivo gráfico e preencher a tela do jogo com uma única cor – **CornflowerBlue**:

```
graphics.GraphicsDevice.Clear(Color.CornflowerBlue);
```

Como falamos anteriormente, o gerenciador de dispositivo gráfico (representado aqui pela variável `graphics`) é o nosso principal meio de acesso às propriedades do nosso dispositivo gráfico. Nesse exemplo, em específico, utilizamos a propriedade **GraphicsDevice**, que expõem as propriedades e métodos que nos permitem ler e configurar os vários detalhes da renderização do nosso jogo.



A finalização do jogo

Se formos fazer uma analogia bem rasteira de como as coisas são, podemos comparar desenvolver um jogo em XNA à organização de uma festa. Para realizarmos uma festa de sucesso, primeiro temos que planejá-la e preparar o ambiente em que a festa será realizada. Ao término da festa, precisamos limpar o ambiente para “devolvê-lo” nas condições que o pegamos. E assim acontece com o XNA também.

Vimos que antes de começarmos o jogo em si, precisamos preparar o ambiente em que o jogo será executado (classes **Game1()**, **Initialize()** e **LoadContent()**), para começarmos a festa (métodos **Run()**, **Update()** e **Draw()**). Ao término do jogo, também precisamos dar uma geral no ambiente. Precisamos esvaziar a memória da plataforma em que o jogo está sendo executado e tudo o mais. O grande diferencial aqui está no método **UnloadContent()** do XNA.

É como se tivesse uma pessoa exclusivamente para executar a faxina no nosso sistema ao término do jogo. O `UnloadContent()` possui o “coletor de lixo” (**garbage collector**) oriundo do .NET Framework que simplifica bastante o desenvolvimento e execução de rotinas de encerramento em jogos XNA. E, o melhor de tudo, ele não atua somente quando o jogo é finalizado, mas também enquanto está em execução. Dessa forma, não precisamos nos preocupar muito com essa coleta de lixo dentro do jogo, a não ser que passemos a desenvolver jogos realmente complexos. Mas isso é assunto para depois.

Hoje tivemos o nosso primeiro contato com o Visual C# Express Edition e com a estrutura de um jogo em XNA. Pudemos sentir de perto como as coisas funcionam e já temos consolidadas as bases para começarmos a desenvolver o nosso jogo em XNA. No nosso próximo encontro iremos ver como funciona o esquema de gráficos 2D no XNA, desenharemos nossos primeiros elementos em tela, daremos vida a eles e os faremos colidir um com os outros. Fiquem atentos que o nosso próximo encontro promete. Até lá!

Tempo de Jogar



Vivemos em uma grande época.

Enquanto pedras no chão serviam como peças em algum jogo da antiguidade ou apresentações teatrais ocorriam nos castelos medievais, um elemento sempre permaneceu igual: divirta-se, oras!

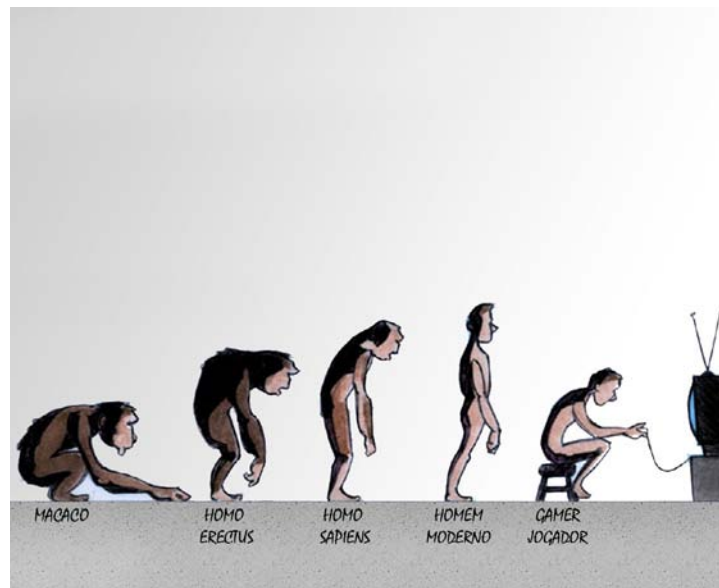
Evolução tecnológica, cultural e comportamental

O estudo do comportamento animal em relação ao ato do Jogo, e que vai muito além das fronteiras da raça humana, reside em lugares mais apropriados do que nesse mero artigo. Mas é a sua essência facilmente percebida que abre caminho para enxergarmos, com orgulho, o mundo ao nosso redor.

Nas últimas décadas evoluímos de forma acelerada em vários segmentos da sociedade - tecnológico, cultural, comportamental. É impossível negar o papel da tecnologia durante esse processo, mas um olhar atento mostra que o termo 'tecnologia' nada mais é do que outro nome para a o centro de tudo: **As ferramentas.**

Sejam físicas, como um tabuleiro de xadrez ou uma bola sobre o gramado do Maracanã, ou fictícias, na mente de toda criança, as ferramentas para o Jogo escondem os reais segredos da diversão plena. Porém, essa diversão não virá até que as ferramentas desapareçam do cenário e entre em cena a imaginação 'pós-surpresa'.

Quando uma nova atividade é introduzida na vida humana nossa primeira preocupação é perguntar 'Como isso funciona? Do que é feito? Como se faz?' - tanto com profetas explicando um cometa ou pais dando aula sobre a internet aos filhos.



Passado o momento de exaltação da ferramenta (ou da tecnologia, como preferir) embarcamos na exploração das possibilidades; começamos a experimentar versões, produtos, opções; passamos a falar com os amigos sobre isso, a procurar outras opiniões, a 'viver' com aquilo dentro de nós.

Trazendo isso para o mundo prático, pense no Twitter. À primeira vista um sistema sem propósito, mas que em pouco tempo esconde sua face óbvia de 'aplicativo' e torna-se parte do comportamento social.

Viajemos rapidamente até o início do cinema. Sempre muda e calada, a mídia fazia sucesso justamente devido à curiosidade que o ser humano tinha pela ferramenta - audaciosamente projetando cenas reais em uma tela de pano.



Charles Chaplin teve a chance de utilizar a fala em seus filmes, mas se negou prontamente. Não que o avanço técnico fosse desprezível. Porém, o mestre havia encontrado uma fórmula que atendia perfeitamente à imaginação do público e não via nas falas algo que justificasse abandonar seu conceito original.

Claro que esse ainda era um período em que não existia a globalização social - e econômica - de hoje. Vivendo em um mundo conectado temos a chance e o privilégio de nos manter informados a respeito dos novos conteúdos que vem por aí.

Mais e as ferramentas?

Os videogames vêm sofrendo uma evolução semelhante à das grandes mídias em seu período de nascimento: avanço técnico, procura por excelência numérica, comparações meramente racionais.

A experiência de vivenciar esse processo deve ser vista como um privilégio por todos nós. Sejam os mais velhos que brincaram com Pong ou os mais novos que começaram com God of War, nossa cross-generation tem a chance de acompanhar a procura de um significado para essa nova ferramenta chamada game.

Assim como em toda a história, a evolução não está nas mudanças pontuais e de incremento mecânico. Ela situa-se em um plano superior, que afeta comportamentos da sociedade de maneiras imprevisíveis. Antes da narrativa, dos gráficos, da computação de nuvem, olhemos para a importância que os videogames vêm ganhando no mundo atual.

Linha do tempo dos consoles.



NINTENDO BLAST

Confira outras edições em:
revista.nintendoblast.com.br