

BLOODMOON

TRANSIÇÃO PARA O MOTOR FÍSICO uBODE

*Compreender as vantagens, as mudanças e as soluções
para os objetos mesh antigos*

OpenSimulator 0.9.3.1

GUIA PARA RESIDENTES E PROPRIETÁRIOS DE REGIÕES

CPU

PRINCIPAL BENEFÍCIO

Redução significativa observada em
vários simuladores

RAM

BENEFÍCIO ADICIONAL

Diminuição constatada em várias
regiões

uBODE

MOTOR PREDEFINIDO

Escolha atual da configuração do
OpenSimulator

Informações sobre esta evolução

Por ocasião da atualização da BloodMoon para o OpenSimulator 0.9.3.1, o motor físico BulletSim foi substituído pelo uBODE nas regiões hospedadas pela grid. Esta evolução traz melhorias muito visíveis à nossa infraestrutura, sobretudo uma redução significativa da utilização do processador.

A RETER

A passagem para o uBODE não danifica os objetos. Pode, contudo, revelar uma forma física antiga, imprecisa ou demasiado complexa, anteriormente mais bem tolerada pelo BulletSim.

Índice

1	Compreender o papel do motor físico
2	Por que razão a BloodMoon escolheu o uBODE
3	Por que alguns meshes antigos podem causar problemas
4	Reconhecer os problemas mais comuns
5	Corrigir as formas físicas no Firestorm
6	Soluções práticas para pisos, escadas e edifícios
7	Veículos e objetos físicos
8	Boas práticas e erros a evitar
9	Como comunicar um problema
10	Conclusão: uma evolução positiva para a BloodMoon

1. Compreender o papel do motor físico

No OpenSimulator, o motor físico gere tudo o que diz respeito a contactos, movimentos e colisões numa região. Não determina a aparência dos objetos: calcula a forma como avatares e objetos interagem com eles.

- impedir que um avatar atravessasse uma parede;
- permitir caminhar sobre um piso;
- gerir quedas e obstáculos;
- mover objetos físicos;
- gerir determinados veículos;
- calcular colisões entre avatares e objetos;

- permitir que alguns scripts detetem um contacto;
- gerir escadas, estradas, pontes e edifícios.

A aparência visual e a física são duas coisas diferentes

Um objeto mesh pode conter duas representações distintas: o modelo visível, apresentado no viewer, e uma forma física invisível, utilizada pelo simulador.

MODELO VISÍVEL	FORMA FÍSICA
• paredes e janelas • relevos e decorações • puxadores, telhas, molduras • detalhes do mobiliário	• pisos onde se pode caminhar • paredes que devem bloquear • portas e passagens abertas • superfícies que provocam colisões

PRINCÍPIO ESSENCIAL

A forma física deve, em geral, ser muito mais simples do que o modelo visível. Uma mesa muito detalhada não precisa de uma colisão que reproduza cada moldura.

2. Por que razão a BloodMoon escolheu o uBODE

O uBODE é agora o motor físico predefinido

A configuração atual do OpenSimulator utiliza o uBODE como motor físico predefinido, associado a um sistema de cálculo de meshes concebido para funcionar com ele. Esta escolha permite à BloodMoon acompanhar as evoluções recentes do software e preparar melhor as futuras atualizações.

Uma redução significativa do consumo de CPU

Este é o principal benefício observado desde a mudança. Com conteúdos equivalentes, vários simuladores BloodMoon utilizam muito menos processador do que com o BulletSim.

- reduzir a carga geral dos servidores;
- manter mais recursos disponíveis;
- limitar alguns picos de utilização do processador;
- distribuir melhor os recursos entre as regiões;
- evitar que uma região muito carregada penalize os outros simuladores;
- facilitar a hospedagem de um grande número de regiões.

OBSERVAÇÃO BLOODMOON

A redução da CPU é o benefício mais visível. Também foi observada uma diminuição da memória, mas geralmente menos acentuada.

Uma utilização da memória mais bem controlada

Desde a passagem para o uBODE, várias regiões também utilizam menos memória. Esta melhoria ajuda a manter uma margem de segurança para eventos, avatares e scripts.

Uma gestão mais rigorosa das colisões

O uBODE interpreta as formas físicas de modo diferente e, por vezes, mais rigoroso do que o BulletSim. Nos objetos corretamente concebidos, as colisões podem ser mais coerentes. Em alguns meshes antigos, esta maior precisão pode revelar um defeito que já existia na forma física.

3. Por que alguns meshes antigos podem causar problemas

Muitos objetos mesh foram criados há vários anos, por vezes com métodos de importação diferentes dos utilizados atualmente. Alguns foram testados sobretudo com o BulletSim.

- sem um verdadeiro modelo físico separado;
- utilizando diretamente o modelo visível como forma física;
- com uma colisão excessivamente detalhada;
- com um invólucro que envolve completamente o objeto;
- com portas e janelas ausentes da forma física;
- com triângulos inúteis ou mal posicionados;
- com uma forma física desalinhada em relação ao modelo visível.

O BulletSim podia, por vezes, tolerar ou interpretar estes objetos de forma a torná-los utilizáveis. O uBODE pode interpretá-los de modo diferente e fazer surgir colisões invisíveis, passagens fechadas ou pisos desalinhados.

IMPORTANTE

Um mesh antigo não contém uma « física BulletSim ». Contém uma forma física que pode ser interpretada de maneira diferente consoante o motor utilizado.

4. Reconhecer os problemas mais comuns

PROBLEMA	Uma porta ou abertura está bloqueada O objeto utiliza frequentemente um invólucro convexo que fecha a abertura, como se o edifício fosse um volume maciço.
O QUE FAZER?	Testar a opção « Prim » na parte afetada. Verificar se apenas a parte ligada correta está selecionada. Como último recurso, desativar a colisão incorreta e recriar as paredes com prim invisíveis.
PROBLEMA	O avatar caminha acima do piso A forma física está colocada acima do modelo visível, dando a impressão de que o avatar flutua.
O QUE FAZER?	Testar a opção « Prim ». Verificar a parte do piso num objeto ligado. Adicionar um prim plano invisível à altura correta se a colisão integrada continuar incorreta.

PROBLEMA	O avatar atravessa o piso A forma física pode estar ausente, desativada ou mal posicionada.
O QUE FAZER?	Verificar se o tipo de forma física não está definido como « Nenhum ». Testar a opção « Prim ». Criar uma superfície invisível simples sob o piso visível.
PROBLEMA	Surgem paredes ou obstáculos invisíveis A colisão pode estender-se para além do modelo visível ou incluir uma zona que deveria permanecer livre.
O QUE FAZER?	Selecionar as partes ligadas uma a uma. Desativar a colisão dos elementos puramente decorativos. Substituir a colisão complexa por alguns prims simples.
PROBLEMA	As escadas tornam-se difíceis de utilizar Cada degrau pode ser interpretado como um obstáculo separado, sobretudo quando a física é muito detalhada.
O QUE FAZER?	Manter a escada pela sua aparência. Desativar a colisão incorreta, se necessário. Adicionar uma rampa inclinada invisível sobre os degraus.

PROBLEMA	Um veículo torna-se instável O veículo depende da sua forma, massa, scripts e do motor físico. Algumas definições foram concebidas apenas para o BulletSim.
O QUE FAZER?	Rezar uma cópia limpa. Reiniciar os scripts. Testar a baixa velocidade num espaço livre. Contactar o criador se o script precisar de adaptação.

5. Corrigir as formas físicas no Firestorm

Primeira precaução: guardar sempre uma cópia

- Verifique se o objeto pode ser copiado.
- Guarde uma cópia no seu inventário.
- Anote a posição e a rotação.
- Sempre que possível, faça os testes numa cópia.
- Não elimine o objeto original antes de validar a correção.

Verificar o tipo de forma física

- Clique com o botão direito do rato no objeto.
- Selecione « Editar ».
- Abra o separador « Funções ».
- Procure a opção « Tipo de forma física ».

INVÓLUCRO CONVEXO	Forma simplificada adequada a pequenos objetos maciços. Pode fechar portas, arcos, túneis ou edifícios ocios.
PRIM	Utiliza a forma física fornecida durante a importação do mesh. Pode restaurar portas e passagens se essa forma tiver sido corretamente criada.
NENHUM	Desativa a colisão. Adequado para plantas, cortinas, quadros e decorações que os avatares podem atravessar.

ATENÇÃO	Não defina sistematicamente todos os meshes como « Prim ». Se a forma integrada for muito complexa ou mal concebida, poderá causar outros problemas e aumentar a carga do simulador.
----------------	--

Modificar apenas a parte afetada

Um edifício mesh é frequentemente composto por várias partes ligadas: piso, paredes, telhado, janelas, escadas e decorações. Nem sempre é necessário modificar o objeto inteiro.

- Ative « Editar partes ligadas ».
- Clique apenas na parte problemática.
- Verifique o respetivo tipo de forma física.
- Modifique apenas essa parte.
- Teste o resultado antes de continuar.

6. Soluções práticas para pisos, escadas e edifícios

Substituir uma física incorreta por prims invisíveis

Quando um mesh possui uma forma física inutilizável, a solução mais fiável consiste em separar a aparência visual da colisão. O mesh permanece visível, a colisão incorreta é desativada e prims simples representam as superfícies físicas.

- um prim achatado para o piso;
- alguns prims para as paredes;
- uma rampa invisível para a escada;
- espaços deixados livres nas portas.

CONSELHO DE CONSTRUÇÃO

Mantenha temporariamente os prims visíveis durante a instalação. Torne-os transparentes apenas depois de verificar a posição e as passagens.

Corrigir escadas com uma rampa invisível

- Mantenha a escada mesh pela sua aparência.
- Desative a colisão se esta causar bloqueios.
- Adicione um prim inclinado sobre os degraus.
- Ajuste a inclinação e a altura.
- Torne o prim transparente depois dos testes.

O avatar subirá assim por uma superfície regular, embora os degraus continuem visíveis. Este método melhora o conforto de deslocação e reduz o risco de bloqueio.

Corrigir um piso ou uma estrada

Se o avatar flutuar acima do piso:

- verifique o tipo de forma física;
- teste a opção « Prim »;
- verifique se o mesh é composto por várias partes;
- desative a colisão incorreta se o problema persistir;
- adicione um prim plano invisível à altura correta.

Se o avatar atravessar o piso:

- verifique se a física não está definida como « Nenhum »;
- teste a opção « Prim »;

- confirme que a parte selecionada é realmente o piso;
- adicione uma superfície invisível se a física integrada não funcionar.

Aligeirar os objetos decorativos

Os objetos puramente decorativos nem sempre precisam de participar nos cálculos físicos. Quando dispõe de direitos de modificação, a opção « Nenhum » pode ser considerada para:

- plantas, erva e flores;
- cortinas, quadros e decorações de parede;
- pequenos objetos colocados sobre móveis;
- elementos fora de alcance;
- detalhes do telhado ou partes exclusivamente visuais.

IMPACTO NO DESEMPENHO

Eliminar colisões desnecessárias reduz o número de cálculos efetuados pelo motor físico e pode ajudar a limitar a carga de CPU da região.

7. Veículos e objetos físicos

Os veículos utilizam simultaneamente o motor físico, a sua forma, a massa e os scripts. Um veículo concebido para o BulletSim pode, portanto, necessitar de adaptação ao uBODE.

- Guarde uma cópia do veículo.
- Reze uma cópia limpa num espaço livre.
- Reinicie os scripts.
- Verifique se apenas as partes necessárias estão físicas.
- Confirme que as decorações não possuem colisões desnecessárias.
- Teste o veículo a baixa velocidade.
- Compare o comportamento com o observado anteriormente.

POSSÍVEL LIMITAÇÃO

Se o veículo continuar instável, o script pode conter parâmetros configurados especificamente para o BulletSim. A equipa BloodMoon nem sempre poderá corrigir um veículo não modificável ou protegido.

8. Boas práticas e erros a evitar

O que não deve fazer

- modificar um objeto sem guardar uma cópia;
- definir todos os meshes da região como « Prim » sem os testar;
- definir todos os objetos como « Nenhum »;
- desligar um edifício complexo sem cópia de segurança;
- eliminar um objeto antes de encontrar uma solução;
- tornar físicos objetos que não precisam disso;

- utilizar centenas de pequenos prims para substituir um piso simples;
- modificar um veículo sem guardar as definições originais;
- reimportar um mesh sem possuir os ficheiros de origem ou os direitos;
- considerar automaticamente cada problema de colisão como uma avaria do simulador.

É necessário substituir todos os objetos antigos?

Não. A maioria dos objetos corretamente criados continua a funcionar normalmente com o uBODE. Só é necessário intervir quando um problema é realmente constatado.

- alterar o tipo de forma física;
- modificar apenas uma parte ligada;
- desativar a colisão de um elemento decorativo;
- adicionar um piso ou uma rampa invisível;
- substituir uma colisão incorreta por alguns prims simples;
- atualizar um veículo antigo.

9. Como comunicar um problema

Se encontrar um problema após a passagem para o uBODE, envie à equipa BloodMoon o máximo de informações possível:

- o nome da região;
- as coordenadas exatas;
- o nome do objeto;
- o nome do proprietário;
- uma descrição precisa do problema;
- uma captura de ecrã, se possível;
- a indicação dos seus direitos sobre o objeto;
- a informação de que o objeto é ou não modificável.

Exemplos de descrições úteis

	EXEMPLO « A porta principal está aberta, mas uma colisão invisível impede a passagem. »
	EXEMPLO « O avatar caminha cerca de cinquenta centímetros acima da estrada. »
	EXEMPLO « O piso está visível, mas caímos através dele. »
	EXEMPLO « Existe um obstáculo invisível diante da escada. »

EXEMPLO

« O veículo funcionava antes, mas agora capota imediatamente. »

O que a equipa poderá verificar

- o tipo de forma física;
- a presença de um invólucro convexo;
- uma colisão incorreta integrada no mesh;
- uma parte ligada específica;
- um script do veículo;
- os direitos de modificação do objeto;
- um eventual problema real do simulador.

10. Uma evolução positiva para a BloodMoon

A passagem do BulletSim para o uBODE representa uma evolução importante para a BloodMoon. As primeiras observações são particularmente encorajadoras, com uma redução muito evidente da carga de CPU de numerosos simuladores.

Também foi observada uma diminuição do consumo de memória, embora o principal benefício continue a ser o processador. Esta melhoria permite manter mais recursos disponíveis para avatares, scripts, eventos e para o conjunto das regiões hospedadas.

Esta transição pode exigir algumas adaptações em determinados meshes antigos, mas estas dificuldades dizem sobretudo respeito a objetos cuja forma física já era imprecisa ou pouco adequada.

**A LONGO PRAZO, O uBODE DEVERÁ PERMITIR À BLOODMOON DISPOR DE REGIÕES
MENOS EXIGENTES EM TERMOS DE PROCESSADOR
MAIS ADAPTADAS ÀS VERSÕES ATUAIS DO OPENSIMULATOR
MAIS FÁCEIS DE GERIR E MAIS ESTÁVEIS
MELHOR PREPARADAS PARA AS FUTURAS EVOLUÇÕES DA GRID**

Obrigado a todos os residentes, proprietários de regiões e criadores pela compreensão e participação na melhoria da BloodMoon.