

# Game Design baseado em Plataforma

*Game Design Platform-based*

Figueirôa, Dino L.; Mestrando; Universidade Federal de Pernambuco  
dinolincoln@gmail.com

Artimã, Williams; Desenvolvedor de Jogos, Layout Informática  
williamsartima@gmail.com

Campos, Fábio; Doutor; Universidade Federal de Pernambuco  
fc2005@gmail.com

Neves, André M. M.; Doutor; Universidade Federal de Pernambuco  
andremneves@gmail.com

## Resumo

A presente pesquisa aborda o paradigma de projeto baseado em plataforma e sua aplicação ao game design, em particular, aos jogos para PC (Personal Computer). Foi realizada uma taxonomia sobre os componentes das plataformas de jogos e realizado um estudo de caso que consistiu na criação de um produto “real”; ao novo jogo gerado denominou-se “P&D MOD”, o qual foi baseado na plataforma “Source”, da Valve Corporation.

**Palavras Chave:** projeto baseado em plataforma, game design, reuso de soluções.

## Abstract

*This research approaches to the platform-based design paradigm and its real application to the game design, talking about PC Games. We has performed a taxonomy about the game platforms components and applied these results into a study of case. As resulting of this process, we conceived a new game called “P&D MOD”, using as base the “Source” game platform, from Valve Corporation.*

**Keywords:** *platform-based design, game design, solutions reuse.*

**Anais do 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design**

8 a 11 de outubro de 2008 São Paulo – SP Brasil ISBN 978-85-60186-03-7

©2008 Associação de Ensino e Pesquisa de Nível Superior de Design do Brasil (AEND|Brasil)

Reprodução permitida, para uso sem fins comerciais, desde que seja citada a fonte.

Este documento foi publicado exatamente como fornecido pelo(s) autor(es), o(s) qual(is) se responsabiliza(m) pela totalidade de seu conteúdo.

## Introdução

A partir da produção em massa do PC (Personal Computer) e de demais artefatos digitais informatizados, nos anos 80, a demanda por metodologias de projeto eficientes em reutilizar e modularizar os designs estava em alta. Destacamos a engenharia de software que modulariza o projeto dos programas de computador permitindo rápida reconfiguração e reuso de soluções (OLIVEIRA, 2006).

É neste âmbito que estreitamos o campo conceitual de nossa pesquisa. Os jogos para PC são, na verdade, modernos programas de computador, herdando os mesmos problemas projetuais de qualquer software. Acrescentando-se, ainda, as particularidades do setor de jogos (BETHKE, 2003).

Neste documento descrevemos como nossa pesquisa mapeou problemas no game design - quanto a reuso de soluções - propondo as plataformas como solução ótima. Desdobrando, desse modo, a pesquisa outrora por nós incursada, na conceituação das plataformas aplicada ao game design (FIGUEIRÔA et al., 2007).

## Procedimentos Metodológicos

Nossa pesquisa consiste na realização de uma taxonomia destrinchando plataformas de jogos para PC. Em seguida realizamos um estudo de caso projetando e desenvolvendo um novo jogo. Demonstrando, desse modo, a aplicação prática dos conceitos sobre plataformas e game design explicitados ao longo do documento.

## O Design Baseado em Plataforma

As plataformas representam uma estrutura de projeto dividida entre componentes fixos e variáveis, além de características chave, conforme descrevemos a seguir (Figura 1).

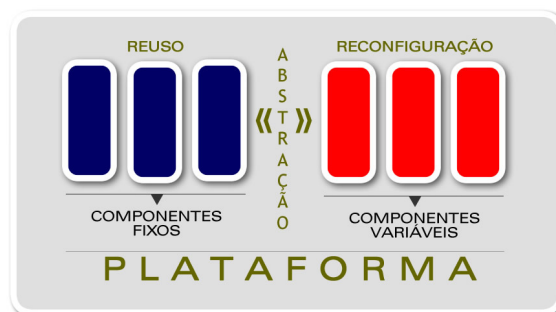


Figura 1 - Estrutura conceitual de uma plataforma

- *Componentes fixos*: Destinados ao reuso e, a princípio, inacessíveis. Nesta seção concentra-se o grande esforço de desenvolvimento inicial de uma plataforma (GOERING, 2002);
- *Componentes variáveis*: Destinados à customização da plataforma. Através da modificação dos mesmos, gera-se projetos distintos (FIGUEIRÔA et al., 2007)
- *Capacidade de abstração*: Quem desenvolve os componentes variáveis abstrai-se da complexidade dos componentes fixos (SANGIOVANNI-VINCENNELLI, 2002);
- *Capacidade de reuso*: Se o projeto não é destinado ao posterior reuso, não é uma plataforma (SANGIOVANNI-VINCENNELLI, 2002).

Baseando-se nestas informações, distinguiremos adiante como este paradigma de projeto pode contribuir efetivamente ao game design.

## Modelo de Plataforma Proposto ao Game Design

Por atender fortes demandas do setor de game design, as plataformas vêm ganhando espaço no mesmo. Resolvendo problemas projetuais como a necessidade por eficiente reuso de código, modularização, redução de riscos do projeto e rápida reconfiguração do programa (RUCKER, 2002; OLIVEIRA, 2006);

Os jogos para PC atuais vêm sendo projetados como plataformas. Como toda ela, possuindo uma sessão de componentes fixos e outra de componentes variáveis. Através da modificação destes variáveis gera-se uma inteira linha de jogos distintos e personalizados, às vezes sem se fazer necessário, sequer, saber programar uma linha de código.



Figura 1 - Estrutura conceitual das plataformas de jogos.

A seguir, a descrição mais detalhada sobre cada componente proposto na Figura 2.

- *Engine*: Representa um conjunto de aplicações responsáveis pelo funcionamento de todo o jogo. Oferecendo programas que trabalham aplicações de rede, detecção de colisões, geração de partículas, inteligência artificial, dentre outros (ROUSE III, 2001). Sendo, portanto, considerado o “centro nervoso” do programa (EBERLY, 2001). A grande vantagem estratégica está justamente nesta composição, que concentra todos os recursos vitais e complexos dos jogos numa unidade central. Onde somente especialistas experientes em suas respectivas áreas podem atuar. Tornando o engine uma seção fechada, o componente fixo da plataforma.
- *Objetos*: Modelos que interagem com o usuário, com o cenário e com outros objetos. Normalmente trata-se de objetos tridimensionais que podem ser modificados ou adicionados.
- *Cenários*: Uma espécie de “objeto grande”, onde atuam os demais objetos. É o componente variável mais estimulado à customização, uma vez que alguns jogos acompanham um programa exclusivo de construção de cenários.
- *GUI*: A interface gráfica com o usuário, também conhecida pelo termo GUI (“*Graphical User Interface*”), divide-se basicamente em interface inicial, onde se encontra o “menu” e demais opções de configuração, e em interface de jogo. (FIGUEIRÔA et al., 2007).
- *Sons*: Este componente variável é um dos principais responsáveis pela imersão dos usuários na temática, dividindo o peso desta tarefa com a GUI.

## Resultados

Nossa pesquisa apresenta “MOD”s, “add-on”s e pacotes de expansão como resultados diretos destas modificações, conforme estaremos detalhando a seguir.

- “MOD”s: A técnica de “*modding*”, como o próprio nome sugere, consiste na modificação dos componentes variáveis da plataforma na geração de novos jogos, que são chamados de “MOD”s (SALEN & ZIMMERMAN). Os “MOD”s podem ser produzidos pelos próprios jogadores (e normalmente o são) na concepção de jogos personalizados, abstraindo-se da complexidade do engine e reusando todos os recursos (Figura 3).



Figura 3 - À esquerda a plataforma “Battlefield 2” e à direita o MOD “STARGATE”

- “*Add-on*”s: Termo comum na comunidade de game design. São, na verdade, um tipo de “*Partial MOD*” (modificação parcial) que, ao invés atribuir ênfase na alteração dos componentes variáveis, atribui na adição em novos componentes similares aos existentes (ROUSE III, 2001). Os simuladores da série “*Microsoft Flight Simulator*”, lideram o uso de “*add-on*”s pela implementação de diversas novas aeronaves (objetos) e cenários.
- *Pacotes de Expansão*: Tratam-se de “*paywares*” (softwares pagos) produzidos e comercializados pelos próprios fabricantes da plataforma, sem alterar a temática do jogo, mas dando continuidade a mesma (Figura 4).



Figura 4 – “Battlefield 2” e “Battlefield 2: Special Forces”: pacote de expansão estende a batalha

## Estudo de Caso

Descreveremos nesta seção como concebemos um jogo, estilo FPS, reusando a plataforma “Source”, esta fora inicialmente desenvolvida para o jogo “Half-Life 2” (HL2). Conforme explicitamos no modelo de plataforma outrora delineado, não modificaremos nada do componente fixo, o engine. Porém, editaremos os componentes variáveis, compreendidos entre objetos, cenários, GUI e sons, gerando, a partir da “Source”, outro jogo, que denominados “P&D MOD”. O mesmo está disponibilizado online, no endereço eletrônico <http://www.dino.eti.br/pedmod>. Para utilizar o mesmo, basta ter a plataforma “Source” instalada, adquirindo o “Half-Life 2” original.

Iniciando a construção do “P&D MOD”, acessamos o programa “Source SDK” onde encontramos o “Hammer Editor” (Figura 5), com o qual criamos um cenário para o nosso jogo.

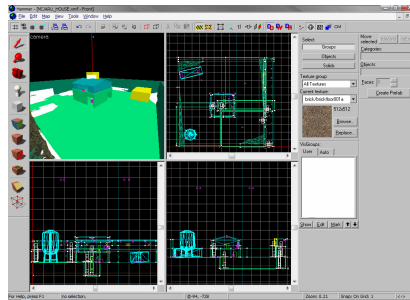


Figura 5 - Construindo o cenário (componente variável) no "Hammer Editor"

A construção de objetos demanda uma equipe competente de modeladores 3D. Em nosso caso, reutilizamos muitos objetos profissionalmente modelados, da riquíssima biblioteca do “*Half-Life 2*”.

Para modificações diversas na GUI, utilizamos um agente facilitador não presente no SDK, chamado “*VTFEdit*”. Modificamos ainda o arquivo “*gameinfo.txt*” para ajustar o menu principal do mesmo, conforme ilustrado na Figura 6.



Figura 6 - Componente variável "GUI" alterado

Para edição do componente variável “som” nesta plataforma acessamos manualmente o arquivo “*game\_sound\_weapons.txt*” e editamos quais ações ou objetos disparariam determinando som. Os que instalarem o nosso perceberão que editamos o áudio da arma “*Stunstick*”. Para conclusão do protótipo do nosso projeto, retornamos às opções do “*Source SDK*” e acessamos “*Create a MOD*” para criar o “*P&D MOD*”. Na Figura 7 temos o jogo em funcionamento.



Figura 7 - Autores jogando o "P&D MOD" em rede local

Finalmente, o “*P&D MOD*” foi concebido no intervalo de 72 horas, sendo 70% do tempo investido na construção do cenário.

## **Conclusões**

Nossa pesquisa trouxe como contribuição aprofundamentos a respeito dos componentes da plataforma, delineados em engine como componente fixo, além de objetos, cenários, GUI e sons como componentes variáveis (FIGUEIRÔA et al., 2007).

A capacidade de reuso e a abstração do paradigma proposto foram demonstrados através do nosso estudo de caso, resultando no jogo para PC “P&D MOD”. No desenvolvimento deste, o tempo e esforço de desenvolvimento foram condensados em largas proporções, reutilizando a plataforma “Source”, abstraindo-se da elevada complexidade computacional do engine.

Concluindo, esperamos ter conseguido explorar a temática dos jogos para PC, transmitindo informações esclarecedoras e contribuindo no detalhamento das características das plataformas atuantes no game design.

## **Referências**

- BETHKE, E. **Game Development and Production**. Plano: Wordware Publishing, 2003.
- EBERLY, D. **3D Game Engine Design**. San Francisco: Morgan Kauffman, 2001.
- FIGUEIRÔA, D., CAMPOS, F., NEVES, A. O Paradigma do Projeto Baseado em Plataformas Aplicado ao Game Design. In: SBGAMES 2007 | VI Simpósio Brasileiro de Jogos para Computador e Entretenimento Digital. São Leopoldo, 7 a 9 nov. 2007. **Anais do VI Simpósio Brasileiro de Jogos para Computador e Entretenimento Digital**. São Leopoldo: UNISINOS, 2007.
- GOERING, R. **Platform-based Design: A Choice, Not a Panacea**. Washington: EETimes, 2002.
- OLIVEIRA, M. **Estimativa e Exploração do Espaço de Projeto Baseados em Modelos para Suporte ao Projeto de Sistemas Embarcados**. Plano de Doutorado. Porto Alegre: Instituto de Informática, UFRGS, 2006.
- ROUSE III, R. **Game Design Theory and Practice**. Plano: Wordware Publishing, 2001.
- RUCKER, R. **Software Engineering and Computer Games**. Boston: Addison-Wesley Longman Publishing, 2002.
- SALEN, K. E ZIMMERMAN, E. **Rules of Play – Game Design Fundamentals**. Cambridge: MIT Press, 2004.
- SANGIOVANNI-VINCENTELLI, A. **Defining Platform-based Design**. In: EEDesign, 2002. (<http://www.eetimes.com/story/OEG20020204S0062>)