

Universidade Federal de Pernambuco
Departamento de Design
Programa de Pós-Graduação em Design



Dino Lincoln Figueirôa Santos

DESIGN DE ARTEFATOS DIGITAIS BASEADOS EM PADRÕES E PLATAFORMAS

Recife
2009



Dino Lincoln Figueirôa Santos

DESIGN DE ARTEFATOS DIGITAIS BASEADOS EM PADRÕES E PLATAFORMAS

Dissertação apresentada ao
Programa de Pós-Graduação em
Design da Universidade Federal
de Pernambuco como requisito
parcial para obtenção do grau de
Mestre em Design

Orientador: Prof. Dr. Fábio Ferreira da Costa Campos

Recife
2009



UNIVERSIDADE FEDERAL DE PERNAMBUCO
PROGRAMA DE PÓS-GRADUAÇÃO EM DESIGN

PARECER DA COMISSÃO EXAMINADORA
DE DEFESA DE DISSERTAÇÃO DE
MESTRADO ACADÊMICO DE
DINO LINCOLN FIGUEIRÔA SANTOS

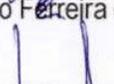
“Design de Artefatos Digitais baseados em padrões e plataformas”

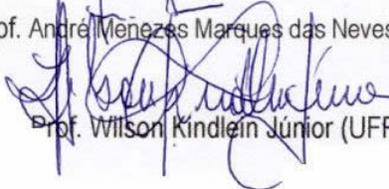
ÁREA DE CONCENTRAÇÃO: DESIGN E ERGONOMIA

A comissão examinadora, composta pelos professores abaixo, sob a presidência do primeiro, considera o candidato DINO LINDOLN FIGUEIRÔA SANTOS **APROVADO.**

Recife, 27 de março de 2009.


Prof. Fábio Ferreira da Costa Campos (UFPE)


Prof. André Menezes Marques das Neves (UFPE)


Prof. Wilson Kindlein Júnior (UFRS)

Ao Mestre dos mestres,
Jesus Cristo.

Agradecimentos

A concepção deste projeto é a materialização de um sonho. Sustenta-se por uma ampla variedade de vetores impulsionadores, sem os quais não seria possível a sua realização. Presto aqui um simples registro de agradecimento a estes colaboradores.

A **Deus**, por não ter me poupado das dificuldades no percurso deste trabalho, mas por usá-las para forjar em mim alguém melhor.

A todos os meus **avôs** e **avós** que me inculcaram desde criança, não com palavras, mas com exemplos, valores de dignidade, esforço pessoal e superação.

Aos meus pais, **Eugênio Lincoln** e **Mirtes**, por terem me impregnado com o gosto pelas ciências através da vida acadêmica e pelo seu sacrifício pessoal em prol de mim, o qual pretendo retribuir em larga escala.

À minha noiva, **Taci**, por manter-se irredutivelmente ao meu lado em formatura fechada, sempre com zelo e carinho diante de todas as dificuldades, em minha busca pelo sucesso (do qual ela faz parte).

Ao meu orientador e amigo, **Fábio**, pela paciência e energia desprendidas em inúmeras reuniões de acompanhamento acadêmico, bem como pela minha orientação profissional e oportunidades a mim direcionadas.

A minha querida tia, Profa. Dra. **Salett Tauk**, pelo ostensivo incentivo para meu desbravamento na carreira acadêmica.

Ao professores **André Neves** e **Rômulo Cesar** pelas suas dicas informais. Bem como ao Prof. Dr. **Wilson Kindlein** (da banca examinadora) por me presentear com tão significativas contribuições.

A todos os **amigos** que contribuíram direta ou indiretamente para este sonho tornar-se realidade, sintam-se direcionado pelo meu humilde agradecimento.

RESUMO

A partir dos anos 1980 presenciamos o advento dos produtos digitalizados que, concebidos em progressão geométrica tornam-se cada vez mais imateriais, abrindo então as linhas de projeto de softwares. No design destes artefatos digitais imateriais, softwares, detectamos um forte “vácuo projetual” de ofertas por reuso de soluções e transferência de conhecimento. Verificamos, portanto, como os paradigmas projetuais baseados em padrões e em plataformas contribuem neste sentido, reduzindo tempo, riscos e custos de projeto através de um extensivo reuso de soluções e transferência de conhecimento. Nos estudos de caso, foram concebidos games e websites utilizando padrões e plataformas. Permitindo ilustrar claramente, na prática, como efetivamente eles contribuem ao design de artefatos digitais, atuando diretamente nas demandas supracitadas.

Palavras-chave: *padrões, plataformas, novos paradigmas de projeto, Game Design, Web Design*

ABSTRACT

From years 1980 we witness the advent of the digital products that, conceived in geometric progression become each time more incorporeal. Opening, this way, the software project frontlines. Over the digital artifacts designing activity, we are talking about softwares, we've detected a huge "projectual empty" for solutions reuse and knowledge transfer. We've verified, therefore, as the pattern-based design and the platform-based design paradigms contribute on this domain, providing reduction of project time, project risks and project cost trough an extensive solutions reuse and knowledge transfer. In our studies of case, we conceive games and websites using patterns and platforms. Allowing to illustrate clearly, in the practical way, as effectively they contribute to digital artifacts designing, acting directly in the above-mentioned demands.

Keywords: *patterns, platforms, new project paradigms, Game Design, Web Design*

SUMÁRIO

LISTA DE FIGURAS	10
LISTA DE QUADROS	11
PREFÁCIO	12
1 PROBLEMÁTICA	13
1.1 Delimitação do Tema	15
1.2 Objetivos da Pesquisa.....	16
1.2.1 Objetivos gerais.....	16
1.2.2 Objetivos específicos	17
1.3 Justificativas.....	17
1.4 Motivações.....	18
1.5 Normatização da Dissertação.....	19
1.6 Metodologia	20
2 DESIGN BASEADO EM PADRÕES	21
2.1 Estrutura dos Padrões	22
2.1.1 Origens na retórica grega	22
2.1.2 Suporte à criatividade	23
2.2 Aplicações dos Padrões	24
2.2.1 Padrões no Design de UI (User Interface)	25
2.2.2 Padrões na Engenharia Eletrônica	25
2.2.3 Padrões na EAD (Educação à Distância).....	25
2.2.4 Padrões de Software	26
2.3 Padrões no Game Design.....	28
2.4 Padrões no Web Design.....	29
2.5 Reuso de Soluções e Transferência de Conhecimento	30
3 DESIGN BASEADO EM PLATAFORMA	32
3.1 Estrutura das Plataformas	33
3.1.1 Benefícios do Design Baseado em Plataforma	35
3.2 Aplicações das Plataformas	35
3.2.1 Plataformas na Indústria Automobilística.....	35
3.2.2 Plataformas na Engenharia Eletrônica.....	36
3.2.3 Plataformas na Engenharia de Software	37
3.3 Plataformas no Game Design.....	37
3.3.1 Modelo de Plataforma de Games	37
3.3.1.1 <i>Engine</i>	38
3.3.1.2 <i>Objetos</i>	39
3.3.1.3 <i>Cenários</i>	40
3.3.1.4 <i>Interface Gráfica (GUI)</i>	40
3.3.1.5 <i>Sons</i>	41
3.3.1.6 <i>SDK</i>	42
3.3.2 Resultados da aplicação das Plataformas no Game Design.....	42
3.3.2.1 <i>“MOD”s</i>	43

3.3.2.2 “Add-on”s.....	44
3.3.2.3 Pacotes de Expansão	44
3.3.2.4 Third Party.....	45
3.4 Plataformas no Web Design	46
3.4.1 Modelo de Plataforma de Websites	46
3.4.2 Resultados da aplicação das plataformas no Web Design	46
3.4.2.1 Blogs, Fotologs e Videoblogs.....	47
3.4.2.2 Templates	47
3.4.2.3 Fóruns.....	47
3.4.2.4 Transmissão de Vídeo.....	48
3.5 Reuso de Soluções e Transferência de Conhecimento	48
4 ESTUDOS DE CASO	50
4.1. Plataformas	51
4.1.1 SBGames MOD.....	51
4.1.1.1 Introduzindo a Plataforma Escolhida: Source.....	51
4.1.1.2 Reusando a Plataforma “Source” para criação do “SBGames MOD”	53
4.1.1.3 Resultados e Breves Considerações	57
4.1.2 P&D MOD	57
4.1.2.1 Reusando a Plataforma “Source” para criação do “P&D MOD”	58
4.1.2.2 Resultados e Breves Considerações	61
4.1.3 Blogger.....	63
4.1.3.1 Resultados e Breves considerações	64
4.1.4 Wordpress	66
4.1.4.1 Resultados e Breves Considerações	69
4.2 Padrões.....	71
4.2.1 Padrão Wizard.....	72
4.2.1.1 Resultados e Breves Considerações	73
4.2.2 Padrão Breadcrumbs (Migalhas de Pão).....	77
4.2.2.1 Resultados e Breves Considerações	79
4.3 Considerações Gerais	81
4.3.1 Discussões sobre Padrões.....	81
4.3.2 Discussões sobre Plataformas	82
4.3.3 Proposta de Integração entre Padrões e Plataformas	83
5 CONCLUSÕES.....	85
5.1 Limitações e Desdobramentos	86
REFERÊNCIAS	87
ANEXO A - PRODUÇÃO ACADÊMICA E TÉCNICA	92
Artigos Completos Publicados	92
Artigos Resumidos Publicados	92
Softwares	93
APÊNDICE A – Padrão “Breadcrumbs”	94

LISTA DE FIGURAS

Figura 1 – Representação da reta infinita dos números reais	13
Figura 2 – Reta dos números reais com referências	13
Figura 3 – Infinitas possibilidades dentro de espaços limitados	13
Figura 4 – Padrões provendo transferência de conhecimento	31
Figura 5 – Ecosport e Fiesta: uma plataforma (Foto: Dino Lincoln Figueirôa).....	32
Figura 6 – Estrutura de uma Plataforma	34
Figura 7 – Ilustração da Plataforma “V.E.N.I.C.E.”.....	36
Figura 8 – Modelo proposto de Plataforma de games	38
Figura 9 – Objetos 3D do game "Ghost Recon"	39
Figura 10 – Interface Gráfica do game "Star Wars Battlefront"	41
Figura 11 – SDK do game "Commande & Conquer: Generals"	42
Figura 12 – O MOD futurista “Galatic Conquest” derivado de “Battlefield 1942”	43
Figura 13 – Os "add-on"s para a Plataforma "Microsoft Flight Simulator"	44
Figura 14 – “Battlefield 2” e "Battlefield 2: Special Forces"	45
Figura 15 – Plataformas provendo abstração	49
Figura 16 – “Insurgency”: Batalhas entre americanos e iraquianos	52
Figura 17 – "Mario Kart: Source": game de corrida.....	52
Figura 18 – "Dragonball: Source"	53
Figura 19 – "Enterprise" do seriado "Startrek"	53
Figura 20 – Tela principal do "Source SDK"	54
Figura 21 – Construindo cenário no "Hammer Editor"	54
Figura 22 – Importando e posicionando objetos no "Hammer Editor".....	55
Figura 23 – GUI inicial do "SBGames MOD"	56
Figura 24 – Jogando o "SBGames MOD" em rede	56
Figura 25 – Efeitos de explosão do "Source Engine" (componente fixo).....	57
Figura 26 – Construindo o cenário no "Hammer Editor"	59
Figura 27 – Selecionando e inserindo objetos no cenário.....	59
Figura 28 – GUI (componente variável) alterado.....	60
Figura 29 – Jogando o "P&D MOD" em rede local	61
Figura 30 – Reutilizando a física do "Source Engine".....	61
Figura 31 – Blog de Michelson Borges: Plataforma "Blogger"	63
Figura 32 – Assistente de Modificação da Plataforma "Blogger”	64
Figura 33 – Websites reusando a Plataforma "Blogger".....	65
Figura 34 – Website do Ministério da Cultura: Plataforma “Wordpress”	67
Figura 35 – Website experimental gerado pelo reuso da Plataforma "Wordpress".....	69
Figura 36 – Websites concebidos pelos alunos reusando a Plataforma "Wordpress"	70
Figura 37 – Formulário original do "Yahoo!"	74
Figura 38 – Redesign do formulário do "Yahoo!".....	75
Figura 39 – Percentual de Respostas para Pergunta 1 – Formulário do “Yahoo!”	76
Figura 40 – Percentual de Respostas para Pergunta 2 – Formulário do “Yahoo!”	76
Figura 41 – Página do website original da UFPE.....	79
Figura 42 – Página reprojeta da UFPE	79
Figura 43 – Resultados mediante apresentação da página original da UFPE	80
Figura 44 – Resultados mediante apresentação da página da UFPE reprojeta da	80
Figura 45 – Resultados do redesign do website da UFPE	81

LISTA DE QUADROS

Quadro 1 - Recorte conceitual de nossa pesquisa.....	16
Quadro 2 - Avaliação da Plataforma "Source"	62
Quadro 3 - Avaliação da Plataforma "Blogger"	66
Quadro 4 - Avaliação da Plataforma "Wordpress"	71
Quadro 5 - Comparativo das avaliações das Plataformas	71

PREFÁCIO

Antes do início do discurso da dissertação propriamente dita, convém que seja relatado como o presente projeto de pesquisa foi iniciado. A princípio, a proposta possuía o intuito de desenvolver uma metodologia de projeto para auxiliar o designer em decisões projetuais com embasamento matemático. Valendo-se de um artefato recém concebido, denominado “Lateo” (FIGUEIRÔA et al., 2008), utilizado para extrair e ilustrar a incerteza envolvida na seleção de alternativas dentro de espaços projetuais. Os paradigmas de projeto que escolhemos para adaptar ao manuseio do “Lateo” eram os padrões (ALEXANDER, 1977) e as plataformas (FIGUEIRÔA et al., 2007).

Entretanto, o aprofundamento realizado nestes paradigmas nos revelou fortes demandas presentes no design de artefatos digitais (necessidade de suplantação de reuso de soluções e transferência de conhecimento). O trabalho realizado para adequação dos padrões e plataformas no cumprimento destas demandas demonstrou-se substancialmente denso.

Deste modo, os estudos sobre padrões e plataformas aplicados ao design de artefatos digitais tomou proporções inesperadas, abrangendo um espaço considerável da pesquisa e realinhando-a a um novo objeto de estudo. Ao longo deste documento será explicitado como estes paradigmas de projeto contribuem ao design de artefatos digitais oferecendo reuso de soluções e transferência de conhecimento.

1 PROBLEMÁTICA

Diante de um dado problema o designer conta com uma ampla gama de métodos, técnicas, processos criativos, experiências alheias e demais recursos que propõem-se a oferecer um trilha mais sistêmico e inovador possível rumo à solução demandada.

É possível, analogamente, comparar este espaço projetual (amostra de todas as possibilidades de projeto possíveis) à reta dos números reais. Sem nenhum tipo de referência nos perdemos na mesma, pois pode ser tão extensa e detalhada o quanto desejarmos (Ver Figura 1).



Figura 1 – Representação da reta infinita dos números reais

A partir do momento que atribuímos pontos referenciais à nossa “reta dos reais”, diga-se espaço projetual, apesar de continuar com a mesma complexidade, a mesma apresenta-se de forma substancialmente mais clara, conforme observamos na Figura 2.



Figura 2 – Reta dos números reais com referências

Estes pontos de referência são, em geral, as limitações e requisitos do projeto tais como tempo disponível para desenvolvimento do mesmo, investimentos financeiros, tecnologia, especificações do problema, etc. Percebamos que, apesar de existir a possibilidade das referências serem claras, temos ainda inúmeras possibilidades de paradigmas de projeto possíveis (Ver Figura 3), desafiando a capacidade investigativa e a criatividade do designer.

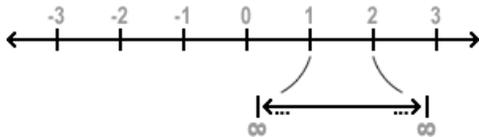


Figura 3 – Infinitas possibilidades dentro de espaços limitados

Este “vácuo projetual” (infinitas possibilidades de projeto diante das referências) tem sido o combustível da contínua criação de novas metodologias de design e aperfeiçoamento das mesmas ao longo de sua recém-criada história.

Em paralelo a este debate, observamos o advento e crescimento exponencial da indústria de artefatos digitais a partir dos anos 80. Este fato traz à humanidade mudanças tão expressivas quanto à sua comunicação, estilo de vida e compartilhamento de conhecimento que contribuições de tamanhas proporções só foram observadas com o advento da imprensa no século XV (BÜRDEK, 2006).

Os produtos passam por um processo acelerado de desmaterialização e nos dias atuais existem softwares embutidos nas mais diversas áreas de nossas vidas. É possível jogar xadrez remotamente com adversários em qualquer parte do mundo, com um tabuleiro virtual na internet, num jogo de computador (que será convencionado daqui por diante pelo termo em inglês “game”, tratando-se de jogos digitais computadorizados, uma vez que “jogo” pode ser confundido com “jogos de azar”, “jogos de tabuleiro”, ou outras modalidades de jogos).

Isto revela aos designers novos pontos de referência em nosso espaço projetual agregando demandas por paradigmas projetuais completamente originais. Afinal de contas, se ampliarmos o detalhamento sobre nossa “reta dos reais” percebemos com facilidade que não é do mesmo jeito que projeta-se uma cadeira que projetaríamos um software.

... programas são tão “impalpáveis” (software) que qualquer tentativa de agarrá-los com as mãos fracassa. Essas não-coisas são, no sentido da palavra, “inapreensíveis”. São apenas decodificáveis.

(FLÜSSER, 2007. O Mundo Codificado, p. 54)

Uma vez que as espécies de produtos digitalizados multiplicam-se dramaticamente, torna-se cada vez mais difícil valer-se de soluções de projetos anteriores, reaproveitando-as com o intuito de encurtar tempo e demais esforços projetuais. E tempo é algo ainda mais crítico no design de artefatos digitais uma vez que as novas tecnologias, recém concebidas, tornam-se obsoletas prematuramente.

Mas não é a primeira vez que deparamo-nos com circunstâncias desta natureza. Nos anos 60, quando os produtos informatizados eram praticamente inexistentes e os designers estavam começando a receber mais responsabilidades dentro da atividade industrial, Christopher Alexander (1964), aclamado como um dos pais da metodologia de design, reportou que:

- A quantidade de informações necessárias para a resolução de problemas de projeto elevou-se de tal forma que o designer por si só não consegue coletar nem manipular;
- A quantidade de problemas de projeto aumentou rapidamente;
- A espécie de problemas de projeto, comparada a épocas anteriores, vem se modificando em um ritmo acelerado, de forma que se torna cada vez mais difícil se valer de experiências anteriores.

Em outras palavras, existia (assim como hoje), uma demanda por metodologias de projeto que suplantassem o reuso de soluções e transferência de conhecimento. A inexistência de metodologias eficazes nestas referências de nosso espaço projetual resultou no que Bürdek(2006) chamou de “mudanças de paradigma de projeto”.

Nas forças deste contexto, o próprio Alexander(1977; 1979) concebeu o design baseado em padrões (“*Pattern-based Design*”). Posteriormente, nas linhas industriais, também observamos o nascimento do paradigma de projeto baseado em plataforma (“*Platform-based Design*” - PBD). Ambos os paradigmas de projeto idealizados no apogeu de suas demandas.

No decorrer deste documento é relatado como foram investigados estes novos paradigmas de projeto, bem como o efetivo funcionamento dos mesmos no auxílio do *reuso de soluções e transferência de conhecimento* dentro design de artefatos digitais.

1.1 Delimitação do Tema

Para comportar a pesquisa dentro das limitações de um projeto de mestrado, foi-se restringido campo amostral investigando dentre os artefatos digitais (que compreendem qualquer produto com informatização embutida) especificamente os imateriais. Em outras palavras, estaremos lidando com design de softwares. Mais detalhadamente, nosso objeto de estudo concentra-se no Web Design e no Game Design, por considerar áreas específicas de expressiva importância dentro do nosso recorte conceitual, abrangendo assim a maior seção possível do segmento de design em estudo (artefatos digitais) (Ver Quadro 1).

“Design para a Web é a nova área mais significativa de prática do design nos últimos 20 anos”.

(MACDONALD, 2003)

Assim sendo, a descrição desta pesquisa sobre os padrões e plataformas, dentre os paradigmas que poderiam emergir em nossa “reta dos reais”, incurvou-se partindo da hipótese de que os mesmos apontam para melhor eficácia disponível na solução do problema em mãos: reuso de soluções e transferência de conhecimento no design de artefatos digitais.



Quadro 1 - Recorte conceitual de nossa pesquisa

Neste capítulo é explicitado com maior exatidão os objetivos da presente pesquisa. É exposto ainda justificativas que representam densas argumentações para a necessidade do presente projeto de pesquisa. Além de motivações para que o atual trabalho fosse desenvolvido. Paulatinamente, ao longo dos demais capítulos tornar-se-á claro exatamente como foi conduzido todo processo de pesquisa.

1.2 Objetivos da Pesquisa

1.2.1 Objetivos gerais

Num contexto mais amplo, o presente projeto propõe-se a contribuir em nossa linha de pesquisa alcançando os objetivos abaixo:

- Levantar o estado da arte, através de revisão bibliográfica, relativo aos tópicos abordados: padrões e plataformas;
- Contribuir com a comunidade científica que discute os paradigmas de projeto propostos, fornecendo resultados de nossos aprofundamentos, incluindo conceituações, softwares gerados e taxonomias;
- Verificar a contribuição dos paradigmas de projeto propostos (padrões e plataformas) no design de artefatos digitais, atentando quanto a sua real utilidade em

prover reuso de soluções e transferência de conhecimento. Sendo este nosso objetivo central.

1.2.2 Objetivos específicos

Descrevendo mais efetivamente do que esta pesquisa realizou, esforços foram concentrados em:

- Realizar aprofundamentos epistemológicos quanto ao paradigma de projeto baseado em plataforma bem como quanto ao paradigma de projeto baseado em padrões, verificando detalhadamente – e especificamente – sua conceituação, estrutura e funcionamento;
- Realizar taxonomias quanto à atuação dos padrões e das plataformas quer no Web Design, quer no Game Design, objetivando disponibilizar uma visualização panorâmica dos mesmos no design de software;
- Incursar estudos de caso no Web Design e no Game Design, visando a validação prática dos conceitos levantados ao longo da pesquisa. Isto inclui o projeto de games e sites baseados em padrões e/ou plataformas.
- Publicar o material gerado por esta pesquisa em congressos de porte expressivo, propiciando uma depuração do nosso trabalho pela comunidade científica bem como maior sustentabilidade das informações por nós levantadas.

1.3 Justificativas

As homepages são o patrimônio mais valioso do mundo. São investidos milhões de dólares em um espaço com menos de um metro quadrado.
(NIELSEN & TAHIR, 2002)

O projeto de artefatos digitais imateriais está supervalorizado em nosso tempo. De acordo com Flüsser(2007): "... o hardware está se tornando cada vez mais barato, ao passo que software, mais caro". O mundo informatizado é um fenômeno presente e em crescimento, de modo que até os porta-retratos são digitais.

Só no ano de 2003 obtivemos um crescimento de 30% no fluxo de novas informações em meios eletrônicos como rádio, telefone, televisão (BERKELEY, 2005). Implicando num agregado crescimento de produtos e softwares que suportam estes canais de comunicação.

Tudo isto embasa investimentos de recursos financeiros e humanos no desenvolvimento do setor. É destacado, portanto, pontualmente abaixo diversos aspectos os quais justificam a nossa pesquisa.

- **Aspecto metodológico:** o design de artefatos digitais imateriais carece de ferramentas metodológicas que aperfeiçoem o reuso de soluções e transferência de conhecimento (BURDEK, 2006).
- **Aspecto comercial:** o mercado de games tem um notório faturamento que supera o da indústria cinematográfica e os websites são considerados, muitas vezes, como um dos patrimônios mais caros da humanidade, pois além de tornar-se uma indústria milionária, estes produtos imateriais atingem milhões de milhões de pessoas (consumidores) (NIELSEN & TAHIR, 2002). Além do mais, as empresas perdem tempo e dinheiro refazendo ou reajustando projetos nestes segmentos, pelo ineficiente reuso de soluções e transferência de conhecimento.
- **Aspecto científico:** devido ao recém advento dos artefatos digitais, diversos aprofundamentos epistêmicos neste domínio são demandados pela comunidade científica, oscilando desde conceituação a taxonomias.
- **Aspecto social:** os produtos digitalizados, os games e a internet mudaram, em diversas fatias da população mundial, a forma de se comunicar, entreter e obter acesso ao conhecimento livre. Portanto, contribuições nestes seguimentos representam benefícios compreendidos ao cotidiano de milhões de pessoas.

1.4 Motivações

Durante mais de uma década desenvolvendo profissionalmente websites, observamos nitidamente, na prática, as demandas que aqui estão sendo destacadas por uma abordagem científica. O contratempo de refazer soluções ao invés de reusar, de buscar aprender por diversos meios como solucionar um problema, ao invés de obtê-lo por um sistema aperfeiçoado de transferência de conhecimento, são motivações ao desenvolvimento deste projeto de pesquisa.

Durante os 2 (dois) anos que trabalhamos no PEDEM (Pesquisa e Desenvolvimento Marista) desenvolvendo aplicações Web para os Irmãos Maristas, participamos da pesquisa descrita por Melo & Neves (2006), onde os autores relatam problemas metodológicos no Web design. Esta pesquisa aponta como reduzimos o tempo de desenvolvimento de

websites através do reuso de aplicativos CSS. Observando assim, de perto, o quanto os Web designers se beneficiam pelo ostensivo reuso.

Motivação esta que nos move a destrinchar, desenvolver e publicar ferramentas metodológicas que, empiricamente falando, auxiliem Web designers de diversas expertises, tanto novatos como experientes.

Por outro lado, também fazendo o viés de *gamer*, é motivador poder contribuir com a comunidade de Game Design, expondo como, através principalmente do paradigma de projeto baseado em plataforma, amadores podem projetar games próximo do limiar profissional.

O interesse pelas amostras do recorte conceitual de nossa delimitação de tema (Web Design e Game Design) é transferida ao presente projeto de pesquisa. Propiciando assim, o mesmo, uma atenção especial. Além da consciência de que o conhecimento disponível no presente documento beneficiará a comunidade do setor.

Nos capítulos seguintes (2 Design Baseado em Padrões e 3 Design Baseado em Plataforma), está exposto construtivamente a fundamentação teórica do tema desta pesquisa.

1.5 Normatização da Dissertação

O presente documento está pautado nas normas abaixo explicitadas, regidas pela ABNT (Associação Brasileira de Normas Técnicas).

- Figuras e Quadros: NBR14724:2001.
- Referências: NBR6023:2000.
- Sumário: NBR6027:1989-NB85:1987.
- Numeração progressiva das seções de um documento: NBR6024:1989-NB69:1987.
- Resumos (errata, julho 1991): NBR6028:1990-NB88:1987.
- Citações em documento: NBR10520:2001;
- Referências: NBR6023:2000;
- Títulos de lombada: NBR12225:1992;

1.6 Metodologia

Partindo da fundamentação teórica, a presente pesquisa baseia-se na seguinte metodologia.

- Revisão bibliográfica no que tange os paradigmas de projeto baseados em padrões e plataformas, destacando suas inflexões ao reuso de soluções e transferência de conhecimento;
- Levantamento e exposição de taxonomias das aplicações dos padrões e plataformas no design de artefatos digitais;
- Experimentação prática (estudos de caso), desenvolvendo games e websites concebidos nos paradigmas de projeto propostos, verificando os benefícios agregados pelo reuso de soluções e transferência de conhecimento providos pelos mesmos.

2 DESIGN BASEADO EM PADRÕES

Bem vindo aos padrões. Alguém já resolveu seus problemas ...
(adaptado de FREEMAN et al., 2004).

O paradigma de projeto baseado em padrões emergiu nos anos 70 quando o arquiteto e matemático Christopher Alexander(1977; 1979) publicou um conjunto de descrições sobre problemas corriqueiros na arquitetura e suas soluções dentro dos respectivos contextos. Cada padrão representa o registro de uma solução num dado contexto, transmitindo didaticamente a experiência para aplicação da mesma em outros contextos. Permitindo assim que esta experiência de projeto seja reusada.

O mais importante tipo de reuso para designers é o reuso de experiência de design.
(adaptado de BOLCHINI, 2000, p. 8)

Os padrões são soluções ótimas para problemas comuns (GRANLUND et al., 2001). Na medida em que problemas comuns são colocados em discussão na comunidade da área e são resolvidos, soluções comuns espontaneamente emergem. Finalmente, a melhor delas se destaca, se auto-identifica e refina-se até atingir a condição de design padrão.

Encontraremos repositórios de padrões, chamados de Linguagem de Padrões, relatando soluções para problemas corriqueiros dentro de um determinado domínio. Algumas das linguagens de padrões são disponibilizadas publicamente, outras, no entanto, são destinadas a uso corporativo, restrito aos projetistas de uma empresa representando um diferencial estratégico em relação à concorrência. De forma mais clara, o conhecimento – que passa a ser transferido internamente na empresa – é uma espécie de patrimônio da mesma.

A seguir, foi detalhado com maior exatidão o que é um padrão através do estudo de sua estrutura e as raízes da mesma na comunicação. Posteriormente encontra-se exposto o estado da arte sobre este paradigma de projeto, explicitando os domínios onde o mesmo faz contribuições expressivas. Por fim, são apresentadas discussões sobre exatamente como os padrões oferecem reuso de soluções e transferência de conhecimento na atividade projetual. Permitindo, deste modo, uma compreensão mais densa a respeito de sua real contribuição ao nosso objeto de estudo além de abranger alguns objetivos específicos de nossa pesquisa.

Todo este procedimento fora realizado através de revisão bibliográfica e estas informações estão posteriormente reaplicadas em nossos estudos de caso.

2.1 Estrutura dos Padrões

Cada padrão consiste numa série de tópicos encadeados descrevendo de forma ampla problema, solução e contexto. A informação é distribuída, a princípio, nos componentes (tópicos) abaixo apresentados.

- **Nome:** deve ser expressivo e conduzir semanticamente o projetista a entender a solução que o padrão descreve;
- **Problema:** descrição do problema em mãos, destacando o porquê de usar este padrão;
- **Contexto:** descrição das pré-condições que em que o designer normalmente encontrará o problema, permitindo uma rápida identificação em outros contextos;
- **Forças:** relacionamentos, contrastes, conflitos e limitações dos elementos permeando a atuação do padrão;
- **Solução:** dado o problema e contexto, trata-se da descrição sobre como o designer interagiu com cada uma das variáveis acima descritas até chegar a uma solução ótima.

A quantidade de componentes e a própria composição deles pode variar dependendo da área cujos padrões estão falando. Mesmo dentro do design de software é possível termos padrões de design com um formato e padrões de programação com outro, visivelmente diferente. Entretanto, habitualmente encontraremos todos os padrões com o mesmo formato dentro de uma mesma Linguagem de Padrões.

Convém enfatizar que, qualquer que seja o formato, o objetivo é claro, transmitir uma experiência de projeto. Em outras palavras, transmitir conhecimento sobre uma dada solução permitindo o reuso da mesma noutro contexto.

2.1.1 Origens na retórica grega

Davide Bolchini(2000) realizou uma pesquisa identificando o porque os padrões são tão eficazes na transferência de experiência de projeto (conhecimento). Na verdade seus trabalhos abrangem o domínio da comunicação e é neste aspecto que ele destaca o poder dos padrões. Por ser uma linguagem facilmente compreendida entre profissionais de um mesmo domínio, disseminando com maior facilidade uma informação útil e sintetizada.

O autor identificou as origens dos padrões na retórica de Aristóteles. Amplamente empregada em discursos, sermões e demais atividades de oratória, a retórica grega consegue de modo imbatível prender a atenção dos expectadores e transmitir-lhes a mensagem desejada com maior possibilidade de aprovação dos mesmos.

A retórica vem sendo estudada há séculos pela lógica, filosofia, literatura e lingüística. Sua persuasão envolve todo o ambiente entre orador e público, incluindo os gestos, interlocutores, os argumentos e etc. De modo que cria-se uma expectativa no público de “ser levado a verdade” pelo orador.

Um dos principais artifícios da retórica trata-se da criação de Tópicos. Aristóteles desenvolveu a técnica permitindo uma correta articulação dos mesmos, permitindo a construção de um discurso persuasivo, obtendo a fé do público para os argumentos transmitidos pelo orador. E, obviamente, temos mais facilidade em absorver a informação pela qual estamos interessados do que aquela que não queremos ouvir.

Convém destacar um fato curioso. Coincidentemente ou não, Aristóteles definia seus Tópicos através de um nome, um contexto, forças envolvidas, exemplos e tópicos relacionados. Estrutura retórica muito próxima da estrutura dos padrões propostos por Alexander (1977).

Dado que o uso da retórica é amplamente aceito em qualquer escola de oratória e a mesma tem sido aplicada, aprimorada, estudada e disseminada há séculos. Possivelmente uma das grandes chances do sucesso atribuído à eficácia dos padrões seja proveniente das suas raízes na retórica grega. Em outras palavras, os padrões possuem um embasamento de séculos de desenvolvimento.

2.1.2 Suporte à criatividade

A princípio, observado um padrão, pode-se equivocadamente subentender que se trata de uma solução pronta, pré-formatada e nada mais. Isto seria um limitador à criatividade uma vez que dado um problema, existiria uma dada solução e ponto final, não restando espaço a inovações.

Entretanto, não é bem assim que as coisas funcionam. O padrão descreve uma experiência de projeto num determinado contexto. Aprendendo com a experiência alheia sintetizada, reduzimos esforços de tempo e recursos em problemas corriqueiros. Abrimos,

assim, espaço projetual para processos criativos no contexto presente. Os padrões não visam abranger todos os possíveis problemas de projeto, mas atingem os problemas comuns, com os quais o projetista não quer perder tempo (mas em muitos casos perde, deixando de aplicar esforços nas inovações para perder energia em problemas que já estão resolvidos).

Reuso consiste em obter vantagem de qualquer esforço de trabalhos anteriores para reduzir o esforço necessário para concluir um novo trabalho.
(adaptado de BOLCHINI, 2000)

Com o intuito de ilustrar isto tomemos, por exemplo, o projeto de um novo carro. O designer automotivo, na concepção de um veículo inovador chega à conclusão de que não utilizará rodas no esquema de trilho (como um tanque de guerra), nem nenhum sistema anti-gravidade, mas optará pela roda automotiva convencional. Não será necessário, portanto, reinventar a roda. Teríamos dentro desta indústria documentos (padrões) descrevendo com detalhes solução de problemas corriqueiros no design de rodas anteriores, os aperfeiçoamentos, problemas, soluções. Isto descrito por designers renomados, com larga experiência. O designer automotivo, portanto, aproveitará esta experiência no seu contexto atual e passará a concentrar esforços nas múltiplas outras complexidades do seu projeto, em vez de dedicar uma maior fração do seu espaço de projeto com um problema corriqueiro cuja depuração está em milhares de outros projetos. Estamos falando de reusar o que é conhecimento empírico, e abrir margem para a criatividade no que venha demandar inovação.

Apesar do exemplo acima ser meramente ilustrativo, é possível imaginar a parcela de contribuição que os padrões oferecem não só a criatividade, mas a muitos outros requisitos do projeto por oferecer mais tempo disponível para o desenvolvimento dos mesmos, além da redução dos riscos de projeto por reutilizar soluções empíricas.

2.2 Aplicações dos Padrões

Por se tratar de uma ferramenta projetual, um conceito, o paradigma de projeto baseado em padrões passou a ser recorrido com grande sucesso por outras áreas além da sua ciência raiz, a arquitetura. Encontram-se descritos a seguir diversos domínios que aproveitam os padrões, permitindo uma visão mais ampla de sua empregabilidade.

2.2.1 Padrões no Design de UI (User Interface)

Os congressos de UI recebem regularmente publicações sobre padrões de projeto (ERICKSON, 1998; GRANLUND et al., 2001; GRIFFITHS & PEMBERTON, 2001). O livro de Jennifer Tidwell (2005), “Designing Interfaces”, é amplamente referenciado neste âmbito, como um bom repositório de padrões para interfaces informatizadas. O que já era de se esperar, uma vez que com a expansão da desmaterialização de produtos o design de interfaces ganhou tremenda valorização (BÜRDEK, 2006).

Erickson (1998) enfatiza o poder dos padrões por se tratarem de orientações destinadas aos profissionais de um mesmo setor (no caso UI), com possibilidade de aplicação em múltiplos contextos dentro do mesmo domínio. Chegando a chamar este paradigma de projeto de “*A língua franca do design*”.

Em contrapartida, Granlund et al. (2001) alega que os padrões podem ser lidos e entendidos por todos, seja da área ou não. Além disto, este autor faz várias outras contribuições relevantes, por exemplo, esclarecendo que os padrões são uma ferramenta para solução do problema ao invés de suporte à criatividade. Destaca ainda que os meios formais de descrição de documentos de UI são fracos, contrabalanceando os padrões como um eficiente método para capturar e transferir conhecimento. O autor define que “padrões descrevem soluções genéricas para problemas comuns num contexto”.

2.2.2 Padrões na Engenharia Eletrônica

Encontraremos, ainda, ampla aceitação dos padrões na engenharia eletrônica. Pont (2001) relata em seu livro diversas soluções para sistemas integrados, utilizando a experiência de projeto com uma família específica de microcontroladores.

2.2.3 Padrões na EAD (Educação à Distância)

Alegando que os projetos de EAD envolvem domínios desconexos, distintos, e que seus profissionais geralmente não são treinados para preparar aulas para este fim, nem tampouco as interfaces das mesmas, Frizell (2001) sugere o desenvolvimento de padrões descrevendo soluções corriqueiras neste tipo de projeto. Isto solucionaria, a princípio, a má qualidade de muito conteúdo didático atualmente em tráfego nos sistemas de EAD via internet, fato que vem a dificultar o aprendizado dos alunos.

Detectando esta mesma demanda, Talarico Neto (2005) desenvolve na sua dissertação de mestrado, em computação, uma Linguagem de Padrões para dar suporte ao projeto de EAD. Ele realça que muitos professores são leigos na preparação deste tipo de material, sendo os padrões desenvolvidos por professores experientes no assunto um método ótimo para transferência de conhecimento neste íterim.

2.2.4 Padrões de Software

Todo desenvolvedor de software deveria ser capaz de usar padrões de forma efetiva. Quando isso for conseguido, seremos capazes de comemorar a inteligência humana que os padrões refletem, tanto em cada padrão individual quanto em todos os padrões na sua plenitude. (BUSCHMANN & MEUNIER, 1995)

O objeto de estudo da presente pesquisa concentra-se no design de software e as informações levantadas nesta seção vem a ser especialmente esclarecedoras para as sessões seguintes, dentro deste mesmo capítulo.

Nos anos 1980, com uma publicação sobre padrões para Programação Orientada a Objetos (POO), Beck & Cunningham (1987) torna-se pioneiros na introdução dos padrões à área de projeto de software. Entretanto, foi partir dos anos 1990 que este paradigma de projeto consolidou-se no setor, iniciando pelas publicações de Coplien (1992) sobre atuação dos padrões na programação em linguagem C++ (COPLIEN & SCHMIDT, 1995).

Em 1995 um grupo conhecido como “Gang of Four” (GoF), desenvolveu e publicou uma série de 23 padrões tratando problemas comuns no projeto e programação em linguagem JAVA. Levando o paradigma de projeto baseado em padrões a ter ressonância sem igual entre os profissionais da área. Para Erich Gama (1995), pertencente ao referido grupo: “Os padrões de projeto são descrições de objetos que se comunicam e classes que são customizadas para resolver um problema genérico de design em um contexto específico”. No livro “Padrões de Projeto” (GAMMA et. al., 2000) os padrões são agrupados por tarefas: criação, estrutura, comportamento. Também encontraremos dentre os padrões de software, padrões de arquitetura de software, descrevendo em caráter de projeto a sua estrutura (SHAW, 1995).

Desse modo, convém destacar duas frentes principais de atuação dos padrões se formam no setor de software, cada qual com suas estruturas e singularidades:

- Projeto (GAMMA et al., 2000; SHAW, 1995); e
- Programação (COPLIEN, 1992)

Entretanto, independente de se tratar de padrões de projeto ou de software, Appleton (1997) descreve quais componentes pertencentes em cada padrão são mais comumente encontrados nos repositórios dos padrões de software:

- **Nome:** algo que lembre o que este padrão é, pode ser orientado ao problema ou a solução.
- **Problema:** o que o padrão propõe-se a resolver.
- **Contexto:** condições nas quais corriqueiramente o problema reincide.
- **Forças:** descrição das influências e restrições do problema dentro do contexto.
- **Solução:** instrução detalhada de como o problema é resolvido, dentro do contexto descrito, podendo envolver além de texto, figuras e esquemas.
- **Exemplos:** aplicações deste padrão no contexto descrito.
- **Contexto resultante:** estado do contexto após a aplicação do padrão.
- **Rationale:** uma explicação de como o padrão funciona e o porque de sua eficácia, descrevendo seu relacionamento com as forças acima citadas.
- **Padrões relacionados:** outros padrões que possuam relações com este, como compartilhar as mesmas forças, ou contexto.
- **Ocorrências:** situações em que este padrão foi usado.

Esse nível de detalhamento de informações reafirma a comparação outrora realizada por Granlund et al. (2001) quando relata que os padrões de software possuem mais componentes, mais detalhes e mais descrições sobre como eles interagem do que os padrões de arquitetura. De fato, foi no design de software que os padrões se estabeleceram com maior sucesso. É neste seguimento que encontraremos o maior número de publicações sobre o assunto.

Appleton (1997) define ainda o conceito de “Anti-padrão” (“*Anti-pattern*”), que representam padrões descrevendo o que não deve ser feito. Em outras palavras, em vez de

descrever uma solução de sucesso, em uso pela comunidade da área, os Anti-padrões descrevem soluções que são corriqueiramente empregadas e falham. Apontam um caminho já traçado pela comunidade e as conseqüências negativas do mesmo, podendo em alguns casos indicar um padrão mais aconselhável para o problema em mãos.

2.3 Padrões no Game Design

O conceito do projeto baseado em padrões está bem difundido na indústria de games. O Game Design (como denomina-se o projeto de jogos digitais) demanda por metodologias que reduzam o tempo de produção de games, através do reuso de soluções. Kabashi (2006) recorre ao projeto baseado em padrões comparando-o como mais eficiente do que as metodologias mais conceituadas no Game Design, incluindo a “*Rouse III*”, no que tange a redução do tempo de produção (como resultado do reuso). Entretanto, a observação do referido autor que serve de maior contribuição à nossa pesquisa é sua justificativa da escolha deste paradigma de projeto, abaixo descrita.

- **Padrões são provados:** refletem a experiência, conhecimento e idéias e desenvolvedores que usaram com sucesso esses padrões em seu próprio trabalho.
- **Padrões são reusáveis:** soluções prontas que podem ser adaptadas a diferentes problemas, se necessário.
- **Padrões são expressivos:** vocabulário comum de soluções que podem expressar amplas soluções de forma sucinta.

A publicação mais difundida sobre padrões no setor é a Linguagem de Padrões de Björk & Holopainen (2005) composta por 200 padrões de Game Design. Conforme observamos na seção 2.2.4 Padrões de Software, neste domínio existem padrões de projeto e padrões de programação. Neste caso tratam-se de padrões de projeto, o que é predominante no Game Design.

Os padrões recebem destaque entre os game designers por serem úteis tanto a projetistas experientes quanto a novatos. Também por atuarem em diversas camadas de desenvolvimento ou mesmo por ser usados em múltiplos projetos envolvendo games distintos. Os padrões prestam ainda comunicação de profissionais de várias áreas correlatas neste contexto, como engenharia de software e design de Interface Homem-Máquina, dentre outros. Afinal, o Game Design é uma mescla destas áreas inter-relacionadas.

2.4 Padrões no Web Design

O Web Design (Como se conhece o “design de websites” ou aplicativos para Web) tem a possibilidade de encapsular todos os setores informatizados acima descritos, dos quais observamos atividades envolvendo o uso dos padrões com grande sucesso.

Os padrões de UI são normalmente utilizados no Web Design em projetos de interfaces e no desenvolvimento das mesmas no ambiente de hipermídia (ADKISSON, 2007). Substantial fatia dos padrões para o design de UI são perfeitamente aplicáveis as interfaces informatizadas dos websites, reduzindo o tempo de produção, pelo reuso, e transferindo conhecimento sobre bom design (TIDWELL, 2005).

Quanto às análises de usabilidade para websites, elas são tão complexas que o Jakob Nielsen & Tahir (2002) lançam uma publicação descrevendo 113 diretrizes de avaliação das mesmas. Este é um processo caro, pela necessidade de larga experiência na área. Algumas agências chegam a cobrar 10 mil dólares pela análise de usabilidade de um website. Encontraremos, portanto, diversas publicações na tentativa de atender uma forte demanda em descentralizar, transferir, o conhecimento sobre usabilidade na Web através dos padrões (GRAHAM, 2003).

Contradizendo alguns equívocos emergidos na comunidade de desenvolvedores de software, de que os padrões só serviriam para atuar em Programação Orientada a Objetos, existem publicações esclarecedoras no sentido de que, obviamente, este paradigma de projeto não limita-se a isto (VLISSIDES, 2007). Por sinal, tratando-se de padrões de software, mesmo aqueles disponibilizados para desenvolvedores JAVA já são atualmente adaptados para PHP (“*Hipertext Pre Processor*”), uma linguagem de programação nativamente online (MELO, 2007). Com isso destaca-se o fato de que o Web Design está sendo beneficiado pelos padrões produzidos para outros estilos de design de software.

Neste domínio encontram-se discussões interessantes relativa a uma deficiência dos padrões. Trata-se de que, apesar de todas as vantagens sobre os padrões, algumas vezes é demasiadamente demorada a procura por um que solucione o problema desejado. Adkisson (2007) afirma ironicamente que às vezes é mais vantajoso investir tempo desenvolvendo a solução do que procurando o padrão através de repositórios inteiros. Em contrapartida esta mesma autora revela que organizar uma biblioteca pessoal de padrões

torna o processo muito mais rápido, destacando a “*Blink Design Library*”, como alternativa para o designer armazenar seus padrões preferidos. Este é apenas um exemplo dentre diversos sites especializados em padrões para Web Design que estão disponíveis online, como “*welie.com*”, “*Yahoo! Pattern Library*” e o “*Desining Interfaces*” (TIDWELL, 2005; ADKISSON, 2007). A difusão dos padrões entre os web designers é tamanha que já existem metodologias de Web Design articuladas a padrões (ISAKOWITZ et al., 1995). Bolchini (2000) categoriza os padrões de projeto para web como: estrutura, navegação, interface e funcionais.

Por fim, os conceitos levantados sobre os padrões, incluindo na comunidade do Game Design, são perfeitamente aplicáveis ao contexto da Web, principalmente no que tange o design de interfaces e engenharia de software. Em suma, encontraremos no Web Design desdobramentos dos padrões presentes em todas as áreas acima descritas, com foco no problema principal destacado neste documento: reuso de soluções e transferência de conhecimento.

2.5 Reuso de Soluções e Transferência de Conhecimento

Experiência, conhecimento, soluções. Estes são recursos que estão implícitos nas mentes de designers veteranos (BOLCHINI, 2000). Como os demais designers (novatos ou experientes) irão ter acesso a estes recursos? Como os detentores deles conseguirão compartilhar com a comunidade de modo que todos entendam sem distorções? Como registrar esta experiência e disponibilizá-la para reuso?

Estamos diante de um problema essencialmente comunicativo. Atuando como ponte, o paradigma de projeto baseado em padrões preenche esse canal de comunicação entre os designers fazendo com que o conhecimento circule na comunidade, seja compartilhado e a solução seja reusada em diferentes contextos. Isto utilizando-se da eficácia e a persuasão da retórica grega para que exista uma otimização na transferência de conhecimento, economizando tempo. Ciente de que Linguagem de Padrões é o local onde procurar soluções para problemas corriqueiros, reduz-se substancialmente energia que seria empregada “reinventando a roda”.

Acrescenta-se a estes fatores a questão de que os padrões são geralmente lidos por profissionais de um mesmo domínio, otimizando-se assim o aspecto semiótico dos conceitos envolvidos entre quem cataloga os padrões no repositório e quem reusa esta solução. Em

outras palavras, com uma *língua franca* o conhecimento é transmitido de forma mais organizada.

Faz-se necessário enfatizar que os padrões transmitem o conhecimento empírico, diferentemente do conhecimento científico. Isto aumenta em muito a aplicabilidade dos mesmos. Mais uma vez, reduzindo o tempo. Uma vez que é empírico, significa que já funciona. Afinal, o padrão é o resultado da *expertise* avançada de um designer ou uma equipe de designers. O que vem a ser especialmente útil para novatos que estão se deparando com problemas que para os veteranos é algo corriqueiro.

É possível afirmar, portanto, que os padrões provêm uma ostensiva transferência de conhecimento a respeito de uma solução num dado contexto de modo que esta experiência empírica de projeto (conhecimento) possa ser reusada em outros contextos, conforme ilustrado na Figura 4.

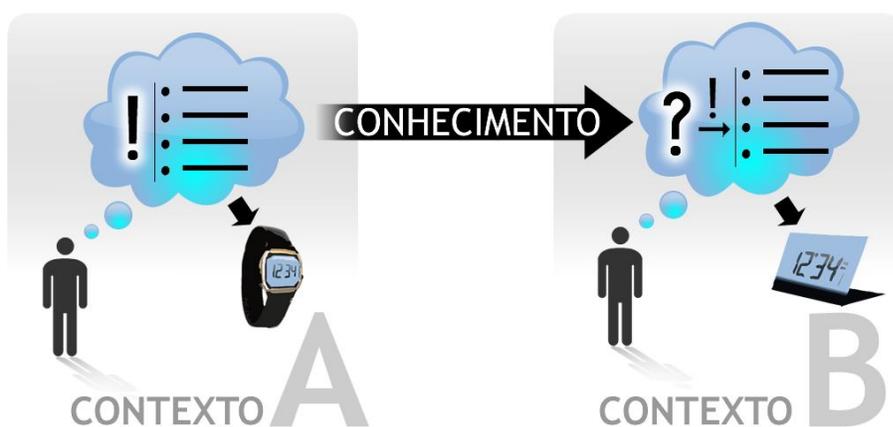


Figura 4 - Padrões provendo transferência de conhecimento

No capítulo a seguir encontra-se descrito em detalhes as informações levantadas pelo levantadas no Estado da Arte a respeito de outro paradigma de projeto completamente distinto, mas que também auxiliam o designer nos mesmos problemas os quais os padrões são eficazes: as plataformas.

3 DESIGN BASEADO EM PLATAFORMA

Conforme outrora mencionado, as novas metodologias de projeto catalizadas pela globalização e pelo crescimento industrial delineado desde o pós-guerra, são categorizadas como “mudanças de paradigma de projeto” (BÜRDEK, 2006). As linhas industriais deparam-se com novas demandas metodológicas uma vez que os projetos tornaram-se por demais complexos, em grande volume e difíceis de serem reusados (ALEXANDER, 1964). Recaindo sobre os designers a responsabilidade de aperfeiçoar o processo industrial.

Nestas circunstâncias, nasce a “plataforma”, um design reutilizável mediante customização de alguns componentes intencionalmente dispostos para este fim (GOERING, 2002). O paradigma de design baseado em plataforma, também conhecido por “*platform-based design*” (PBD), emergiu nos ambientes industriais como alternativa modularizar e reaproveitar projetos eficientemente (DE MARINIS et al., 2003). Através da mesma uma única plataforma de chassis poderia servir para uma inteira linha de carros, com a modificação de apenas algumas seções configuráveis, intencionalmente dispostas para este fim (Ver Figura 5) (CARDOSO et al., 2006).



Figura 5 - Ecosport e Fiesta: uma plataforma (Foto: Dino Lincoln Figueirôa)

A engenharia eletrônica absorveu fortemente este paradigma projetual, principalmente no design de componentes de hardware que, atuando como plataformas, atendiam a um amplo número de projetos (BOULDIN, 2003); dessa maneira evitando que fosse desenvolvido um chip para cada atuação diferente (SANGIOVANNI-VINCENTELLI, 2007).

Outra área onde as plataformas são aceitas com grande sucesso é a engenharia de software (GRAAF et al., 2003) (sobre esse campo nos aprofundaremos no decorrer desse trabalho), seguindo o mesmo conceito predominante nas outras áreas, onde uma única plataforma de software pode ser produzida para servir a uma grande variedade de aplicações, proporcionando aos desenvolvedores reutilização de código e redução do *time-to-market* (tempo até o mercado).

Chegamos então ao projeto de games e websites. Nesse ínterim, se observarmos estes produtos digitais, veremos que se tratam de sofisticados programas de computador, herdando as mesmas demandas e problemas de projeto encontrados na engenharia de software (CLAYPOOL & CLAYPOOL). Estes segmentos estão rapidamente incorporando as plataformas, produzindo games e websites configuráveis.

No presente capítulo encontra-se detalhadamente o que é uma plataforma, haja visto que a mesma ainda possui uma conceituação controversa (GOERING, 2002). Estes estudos renderam informações sobre a atuação das plataformas no Game Design e no Web Design, resultando em taxonomias. O procedimento consiste em analisar a estrutura das plataformas, identificar sua atuação na atividade projetual através de um levantamento literário e pela geração de taxonomias dos seus resultados. Verificando, por fim, com base nestas informações, como as plataformas oferecem reuso de soluções e transferência de conhecimento.

3.1 Estrutura das Plataformas

Considerando que existem muitas conceituações para o “design baseado em plataforma” (WOLF, 2006; GOERING, 2002), convém adotarmos por alicerce o conceito de Sangiovanni-Vicentelli (2002), o qual conceitua este paradigma de projeto como uma camada de abstração com duas visões. Visões essas que representam a seção com componentes fixos e a seção com componentes configuráveis. Neste paradigma, quem desenvolve/configura os componentes da seção configurável se abstrai da complexidade da seção fixa; sendo essa abstração uma das suas maiores vantagens.

O Grupo de Desenvolvimento em Projeto Baseado em Plataforma (*“Platform-based Design Development Working Group”* – PBD DWG) sugere que esse paradigma seja utilizado na forma de “uma abordagem de design integrado enfatizando reuso sistemático para desenvolvimento de produtos complexos sob Plataformas... ..com a intenção de

reduzir custos de desenvolvimento, riscos do projeto e tempo até o mercado” (GOERING, 2002).

Apesar das discussões, o paradigma de projeto baseado em plataforma ganha consistência por apresentar características constantes mesmo sendo muitas vezes mencionado em áreas distintas (vide Figura 6). Abaixo, as características que esboçam uma plataforma.

- **Componentes fixos:** onde se encontram a maior parte dos recursos de disponibilizados e onde se concentra o maior esforço de desenvolvimento inicial, também definido como “sessão de arquitetura” (CHEN et al., 2003);
- **Componentes variáveis:** através das modificações dos quais é possível gerar produtos distintos (mas pertencentes a uma mesma plataforma), a chamada “sessão de aplicações” (CHEN et al., 2003);
- **Capacidade de abstração:** os desenvolvedores dos componentes variáveis não precisam se aprofundar na complexidade de desenvolvimento dos componentes fixos, mas sim em como interagir com os mesmos (SANGIOVANNI-VINCENTELLI, 2007)
- **Capacidade de reuso:** os componentes fixos são reutilizados em outros projetos, além disso, se o projeto não é destinado ao reuso, não é uma plataforma.

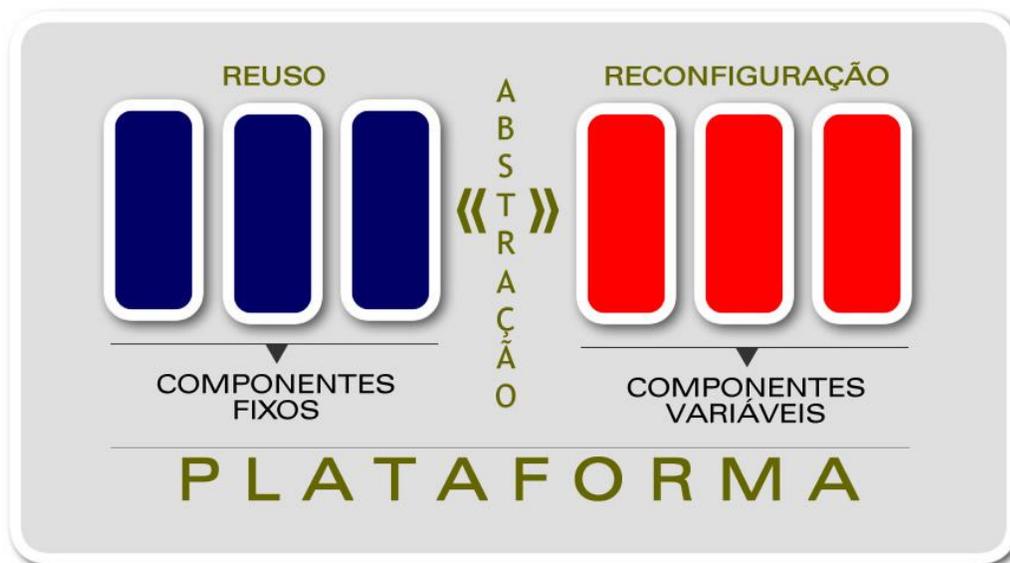


Figura 6 - Estrutura de uma Plataforma

3.1.1 Benefícios do Design Baseado em Plataforma

Design baseado em plataforma é um poderoso conceito de cópia na acrescida pressão por redução de “tempo até o mercado”, além de oferecer redução de custos de projeto e manufatura.

(adaptado de SANGIOVANNI-VINCENTELLI, 2007)

Os objetivos deste paradigma de projeto consistem em reduzir o tempo de lançamento até o mercado e prover capacidade de abstração otimizando o sistema cartesiano do processo de design (NSAME et al., 2001). Sob este paradigma, a redução de tempo é tamanha que, de acordo com Augusto Oliveira, arquiteto chefe da Philips, alguns projetos caminham para 50% de economia no tempo de desenvolvimento. Isto se dá porque a plataforma limita as escolhas de modo organizado, provendo um intenso reuso de design (GOERING, 2002). Isto implica também em redução nos custos do projeto.

Apesar do reuso da plataforma reduzir drasticamente o tempo de desenvolvimento de um projeto, a construção da mesma pode ser algo demorado e isto precisa ser levado em conta na hora de balancear com outras opções de processo de design (GOERING, 2002). Entretanto, os benefícios se estendem ainda à qualidade dos produtos, uma vez que uma plataforma é o resultado de um projeto já testado, refinado. De acordo com a IBM: “os clientes já começam a experimentar as melhorias devido ao design baseado em plataforma” (NSAME et al., 2001).

A grande capacidade de abstração permite que as plataformas operem em condições de design multidisciplinares, uma vez que os desenvolvedores dos componentes variáveis abstraem-se da complexidade dos componentes fixos (FIGUEIRÔA et al., 2007).

3.2 Aplicações das Plataformas

Nesta seção encontram-se expostos os resultados de do Estado da Arte, concebido por revisão bibliográfica, onde é possível observar panoramicamente as diversas áreas de atuação onde as plataformas estão sendo aplicadas.

3.2.1 Plataformas na Indústria Automobilística

A indústria automobilística *FIAT* vem utilizando a técnica das plataformas em diversas divisões dos projetos de seus carros. Tanto motores e chassis quanto em sistemas eletrônicos. O projeto “*V.E.N.I.C.E.*” (ou “*Rede Veicular com Controle Eletrônico*”) é uma

resposta baseada em plataforma diante uma problemática internacional. Um mesmo modelo de automóvel tem configuração diferente do sistema de luzes em cada país comercializado, de acordo com a legislação local. Entretanto para evitar contratempos tais como produzir diferentes sistemas, componentes e instruções de instalação para as fábricas, além de dificuldades em vender carros fabricados para um país em outro, a *FIAT* produziu uma única plataforma do sistema eletrônico que administra as luzes dos veículos para diversas configurações.

Como toda plataforma, esse sistema possui a sessão de componentes fixos, tais como cabos, lâmpadas, circuitos eletrônicos, códigos de software e microprocessador. E possui a sessão de componentes variáveis, que, nesse caso particular, trata-se apenas de um elemento: a informação proveniente da leitura de um código ou chip de memória. Dependendo da informação inserida nesta área, o sistema se informa acerca de qual configuração de luzes deve ser assumida, modificando suas características para atender a legislação de um determinado país (vide Figura 7). Caso pretenda-se registrar o veículo em outra jurisdição, basta encaminhar-se a uma manutenção autorizada *FIAT* para, novamente, ajustar qual configuração o sistema deve assumir.

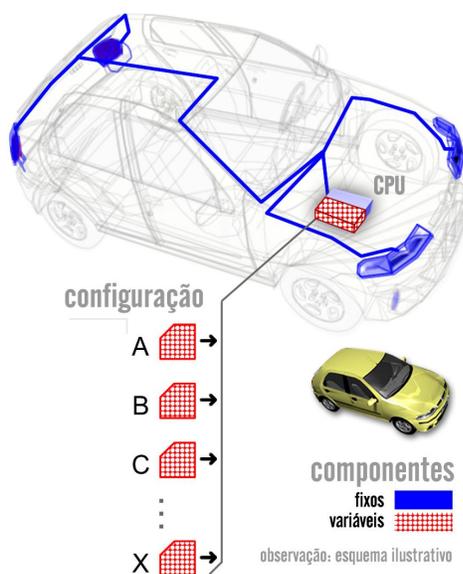


Figura 7 - Ilustração da Plataforma “V.E.N.I.C.E.”

3.2.2 Plataformas na Engenharia Eletrônica

O design de “SoCs” (“System on Chip”) é o setor de aplicação das plataformas que mais produz publicações descrevendo suas atividades e resultados. Sangiovanni-Vincentelli

(2002) é aclamado por muitos como um dos originadores do conceito (CHEN et al., 2003; GOERING, 2002). Suas publicações, entretanto, mostram exclusivamente seus experimentos e pesquisas com plataformas nos projetos de engenharia eletrônica, especialmente envolvendo “SoCs”. Os componentes nos designs baseados em plataforma de Sangiovanni-Vincentelli (2002) normalmente são sistemas eletrônicos (compostos por software e hardware) e circuitos integrados.

Por todos os benefícios já descritos, e pesquisas desenvolvidas no setor, o paradigma de projeto baseado em plataformas consolidou-se no design de “SoCs”, conquistando a atenção de instituições como IBM e NASA (BOULDIN, 2003; NSAME et al., 2001).

3.2.3 Plataformas na Engenharia de Software

Na engenharia de software, onde a abstração exerce uma forte demanda não só no produto final, mas também no seu desenvolvimento, as plataformas são usadas em diversas vertentes. Por exemplo, a programação orientada a objetos que emprega o paradigma em questão utilizando os argumentos, ou variáveis de entrada, como componentes variáveis; e toda a estrutura restante do objeto, como componentes fixos. Visualizando os “métodos” ou “classes” como pequenas plataformas dentro do programa (GRAAF et al., 2003).

É importante destacar que podemos ter objetos dentro de objetos, como métodos que possuem outros métodos dentro de sua estrutura. Além da recursão, onde uma função computacional pode ter ela mesma como componente interno. Desdobrando-se em diversos encapsulamentos de objetos e demonstrando assim a capacidade das plataformas de ter outras plataformas dentro de sua estrutura. Atuando como diferentes tipos de componentes. Em outras palavras, uma plataforma pode ser um componente, fixo ou variável, de outra plataforma.

Este campo de aplicação das plataformas aproxima-se do objeto de estudo da presente pesquisa (Game Design e Web Design). Encontra-se descrito este aprofundamento adiante.

3.3 Plataformas no Game Design

3.3.1 Modelo de Plataforma de Games

Os games 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 games distintos e personalizados,

às vezes sem se fazer necessário, sequer, saber programar uma linha de código. Na Figura 8 ilustramos a disposição dos componentes de uma plataforma de games.



Figura 8 - Modelo proposto de Plataforma de games

A seguir, a descrição mais detalhada sobre cada componente proposto na Figura 8 e ainda a apresentação de um tipo de ferramenta de edição de plataformas (SDK – “*Software Development Kit*”, ou “Conjunto de Desenvolvimento de Software”), comumente encontrada nos games atuais para customização de componentes variáveis.

3.3.1.1 Engine

É a central nervosa do programa do game, sendo responsável pela geração de gráficos 2D e 3D (ou “*rendering*”), aplicações de rede, animações, detecção de colisões, inteligência artificial, execução de sons, geração de partículas, dentre outros (ROUSE III, 2001). Chegando muitas vezes a ser modularizado, por tarefa, tais como “*render engine*” para gráficos, “*physics engine*” para cálculos de animação de objetos (EBERLY, 2001; 2003) e assim por diante. Em outras palavras, o engine principal pode ser entendido como o encapsulamento de vários engines.

Um engine completo pode ser desenvolvido na forma de plataforma, para atender linhas inteiras de games. A *Valve Corporation*, fabricante da série de games de tiro em primeira pessoa mais jogados do mundo, o *Half-Life*, desenvolveu recentemente o “*Source Engine*”, para o *Half-Life 2*. Utilizando Visual Studio 7.1 e linguagem de programação C/C++. O

“Source”, como é conhecido, trata de recursos extremamente avançados como o inovador “digital muscles”, tecnologia de reprodução de expressão facial humana e não-humana em personagens 3D. Capaz de rodar em diferentes ambientes (Microsoft Windows, Xbox 360 e Playstation 3) este mesmo engine é utilizado para os games *Counter Strike: Source*, *Team Fortress: Source*, *Day of Defeat: Source*, e centenas de outros. Justificando assim o seu custo de desenvolvimento, uma vez que terá as vendas multiplicadas pelos diversos segmentos de games que utilizam seus recursos. Este custo de desenvolvimento varia bastante, dependendo obviamente dos requisitos que o projeto visa atender. Se o game vai possuir multiplayer (múltiplos jogadores simultâneos) tornando necessário desenvolvimento de aplicações de rede, ou se disponibilizará campanha, propiciando uma necessidade de level design. Contudo, para se ter uma idéia da dimensão deste setor, o desenvolvimento do engine do bem sucedido “*Warcraft III*”, da Blizzard Entertainment, chegou à marca de 3,7 milhões de dólares.

3.3.1.2 Objetos

São modelos, geralmente 3D, que interagem com o usuário e o cenário (FULLERTON et al., 2004). Atuados automaticamente pelo engine, algumas vezes articulado com IA (inteligência artificial), outras vezes com movimentos programados, ou ainda pela intervenção do usuário. Compõem os personagens, armas, veículos e etc.

Enquanto que em alguns games pode-se trocar a textura dos modelos 3D apenas, em outros é possível inserir objetos completamente novos, assim como suas características de comportamento, modelados em programas de edição tridimensionais externamente (vide Figura 9). Tudo dependendo da flexibilidade da plataforma em questão.



Figura 9 - Objetos 3D do game "Ghost Recon"

3.3.1.3 Cenários

Ambientes virtuais dentro dos quais atuam os objetos. Pode ser encarado como um “objeto grande” que interage com os demais objetos, porém com algumas características bastante particulares, como tecnologia para geração de céu virtual (“skybox”), ou efeitos de reflexo na água (BETHKE, 2003). O design de cenários vem ganhando aperfeiçoamentos nos últimos anos com o progresso das tecnologias de gráficos vetoriais 3D, que possibilitam a “renderização” de cenários com mais objetos e detalhes.

A maioria dos SDK's para games possuem apenas construtores de cenários, os quais são, na verdade, um administrador de um conjunto de objetos posicionados sobre um terreno atuando com um seletor de texturas (paredes, terrenos, águas, etc.) para superfícies de objetos e terrenos.

3.3.1.4 Interface Gráfica (GUI)

A interface gráfica (“*Graphical User Interface*” - *GUI*) é a principal responsável por imergir o usuário na temática que os designers projetaram para o game. Segue a idéia do desenho de conceito, transmitindo informações que cognitivamente estimulem o jogador a sentir-se envolvido pelo ambiente proposto. Conceitualmente, podemos subdividir o sistema em duas interfaces.

Na primeira situam-se as configurações preliminares ao game, tais como: ajustes gráficos, opções de dificuldade, seleção de estágios e etc. Exemplificando, um game de guerra pode ter como tela de fundo imagens de armas, enquanto um simulador de cidades reproduz um centro comercial. Em alguns casos o usuário interage com uma interface tridimensional, como em *Star Wars Battlefront*, onde é passada a impressão de alta tecnologia (veja Figura 10).



Figura 10 – Interface Gráfica do game "Star Wars Battlefront"

A segunda é a interface de game, onde adota-se o termo técnico “HUD” (“*Head Up Display*”). Trata-se da mira, quantidade de munição, radares, tempo restante do estágio, etc. Estes elementos dinâmicos são conhecidos como “*widjets*” ou “*gauges*”. Por serem efetivamente interligados ao engine, tornam esta interface mais trabalhosa para manuseio. Podendo vir a exigir alguma intervenção em linguagem de programação.

3.3.1.5 Sons

Este componente variável é um dos principais responsáveis pela imersão dos usuários na temática, dividindo o peso desta tarefa com a GUI. Corriqueiramente encontraremos games para PC disponibilizando os sons para customização, mesmo que compactando os arquivos num formato específico, que não seja “.wav” ou “.mp3”, mas conhecido apenas pelas produtores da plataforma. Entretanto, rapidamente a comunidade do setor descobre como acessar estes arquivos. Em alguns casos acesso é facilitado, sendo possível modificar os sons através de um programa que acompanha o game, desenvolvido para este fim. Como podemos encontrar no clássico “*Warcraft III*”, da *Blizzard Entertainment*. Este game acompanha um “*Sound Editor*” e, dentre as opções do mesmo, é possível importar diferentes formatos de arquivos de áudio digital. Sendo ainda possível configurar diversos efeitos como abrangência de um determinado som dentro do cenário, repetições (*loops*), “*fade in*”/“*fade out*”, e outros. Isto permite ao usuário instalar sua própria voz nos personagens do game.

3.3.1.6 SDK

Muitos games acompanham esta ferramenta gráfica para desenvolver recursos adicionais (NIEBORG, 2005), como o “*World Builder*” do “*Command & Conquer: Generals*”, EA Games, onde o usuário pode criar novos campos de batalha e adicionar à biblioteca de cenários (vide Figura 11).



Figura 11 - SDK do game "Commande & Conquer: Generals"

Algumas empresas disponibilizam programas para desenvolvimento de games acompanhados de um engine para que o desenvolvedor usufrua de seus recursos se abstraindo da complexidade das aplicações que rodam no mesmo. O “*Torque Game Engine*” da *Garage Games* é um exemplo bastante difundido. Utilizado para produzir, dentre outros, o game “*Wildlife Tycoon: Venture África*”. Ele constitui-se num engine repleto de recursos acompanhado de uma API (“*Application Programming Interface*” – Interface de Programação de Aplicações) para administrar os componentes variáveis da plataforma.

3.3.2 Resultados da aplicação das Plataformas no Game Design

As alterações dos componentes variáveis dos games em plataformas são tão comuns hoje em dia que já ganharam até nomes. “MOD”s, “add-ons”, e pacotes de expansão são, na prática, quase tão conhecidos quanto os próprios games para os quais eles são produzidos. Websites, grupos e até empresas utilizam a estrutura de plataforma para criá-los, possibilitando a implementação novos mapas, armas, cenários, interfaces e até narrativas, criando um game totalmente novo a partir da plataforma principal.

Encontram-se descritos abaixo os conceitos dos principais tipos de artefatos gerados a partir das plataformas de games.

3.3.2.1 “MOD”s

Como o próprio nome sugere, “MOD” é o termo abreviado para “modificação”. Os componentes variáveis, de acordo com o modelo de plataforma, são alterados para transmitir a idéia de que se está em outro game. Atividade também conhecida por “modding” (SALEN & ZIMMERMAN, 2004).

Podem ser produzidos tanto por empresas, com fins comerciais, quanto pelos usuários sem fins lucrativos. Mesmo sendo “freewares” em sua maioria, certas vezes causam um verdadeiro *frenesi* no mercado. Pois para utilizar um “MOD”, é necessário ter a plataforma principal comprada, para então modificá-la (BETHKE, 2003). Por exemplo, muitos fãs da trilogia cinematográfica “*Star Wars*” utilizam “MOD”s “freeware” para transformar diferentes games no ambiente “*Star Wars*” (veja Figura 12).



Figura 12 - O MOD futurista “Galatic Conquest” derivado de “Battlefield 1942”

O “MOD” mais famoso até então é o “*Counter-Strike*” (também conhecido por “CS”), que reproduz batalhas entre policiais e terroristas. Muitos dos milhões de jogadores deste game ao redor do mundo, que atinge a marca de mais de 94 mil servidores online, nunca ouviram falar do “*Half-Life*”, que acumula cerca de 800 servidores (VALVE, 2007). Enquanto que na verdade, o “CS” é um “MOD” do “*Half-Life*”. O “MOD” fez mais sucesso do que a própria plataforma em que foi criado, tornando-se o “*First Person Shooter*” (FPS) multiplayer mais jogado do mundo.

Basicamente, estes subprodutos se categorizam entre “*Partial MOD*” e “*Total MOD*” (este também conhecido por “*Total Conversion*”), representando uma modificação parcial ou total do game, respectivamente.

3.3.2.2 “*Add-on*”s

Consistem em “*Partial MOD*”s, com objetos ou cenários adicionais, mas sem alterar o drasticamente a temática do game nem a GUI (ROUSE III, 2001) (vide Figura 13). É uma categoria amplamente empregada em simuladores. O “*Microsoft Flight Simulator X*”, também conhecido por “FS”, na versão mais completa (“*Deluxe Edition*”) acompanha 24 aeronaves disponíveis para pilotagem. Cada um destes objetos são compostos de modelo 3D externo, painel 2D, painel 3D (*cockpit*), texturas, sons e realismo de voo (arquivos de extensão *.*cfg*). Entretanto um dos fatores associados ao grande sucesso do “FS” é a possibilidade de adicionar, literalmente, milhares de novas aeronaves disponibilizadas pela internet.



Figura 13 - Os “add-on”s para a Plataforma “Microsoft Flight Simulator”

Apesar da maioria ser “*freeware*”, algumas empresas especializadas em “add-on”s atendem a demanda pelos mesmos, como a *World Sceneries* que fornece cenários de “FS” para uso da *Força Aérea Brasileira*.

3.3.2.3 Pacotes de Expansão

Constituem-se de “*paywares*” (softwares pagos) lançados pelo próprio fabricante da plataforma. São uma continuação do game sem alterar sua temática, particularmente comuns em games de campanha, estendendo a mesma (Ver Figura 14). Pode ser considerado um super pacote de “add-on”s, mas que altera drasticamente o tamanho do game (SALEN & ZIMMERMAN, 2004).



Figura 14 – “Battlefield 2” e "Battlefield 2: Special Forces"

Comercialmente falando, servem de manutenção do ciclo de vida do produto, após sua fase de maturação. Permitindo que o mesmo continue sendo atrativo no mercado quando começaria a ter um declínio natural de suas vendas, inerente a qualquer produto. Depois de um tempo de depreciação, torna-se comum a venda do game e a expansão, ou expansões, num único pacote com o preço reduzido.

3.3.2.4 Third Party

Iniciar um projeto da estaca zero é um dos problemas que as plataformas se propõe a resolver e o faz com grande eficácia, conforme temos ilustrado ao longo desse trabalho. A criação de muitos games é provida pela cópia autorizada de componentes de outras plataformas, reaproveitando-os num novo projeto. Por exemplo, o engine do “*Star Wars Gallatic Battlegrounds*” da *LucasArts* declara na caixa do produto: “Rodando com o engine do *Age of Empires II*”.

O “*Genie Engine*”, desenvolvido pela *Ensemble Studios* para o game “*Age of Empires II*”, foi comprado e aplicado em outro game com todos os seus recursos; como as aplicações multiplayer, reprodução de níveis (level design) e música de fundo, dentre outros. Mais um caso é o “*The Battle for Middle-earth II*”, lançado em 2006 que utiliza o “*Sage Engine*” desenvolvido para o “*Command & Conquer: Generals*”, todos da *EA Games*. Um reproduz batalhas épicas da trilogia cinematográfica “*Senhor dos Anéis*”, enquanto o outro simula guerra moderna com direitos a tanques, caças e até bombas atômicas. Contudo, ambos os games são bastante parecidos quanto à jogabilidade. No primeiro, conforme o jogador enfrenta inimigos, recebe pontos de comando que lhe permitem chamar reforços como as “águias gigantes” ou “exércitos dos mortos”. Já no segundo o general ganha patente, permitindo-lhe chamar apoio aéreo de pára-quedistas ou lançar minas terrestres na base inimiga, dentre outros. Entretanto, tecnicamente falando, são games iguais com interfaces (GUI) e sons diferentes.

Com isso, encerra-se a presente taxonomia da aplicação das plataformas no Game Design. Adiante encontra-se exposto o aprofundamento quanto a este paradigma de projeto no Web Design.

3.4 Plataformas no Web Design

3.4.1 Modelo de Plataforma de Websites

Encontra-se em destaque esta do documento na tentativa de explicar o porquê não é trivial definir uma plataforma de websites. Ao contrário dos claros resultados de nossa taxonomia no Game Design, as plataformas encontradas na Web são demasiadamente diversificadas.

Entretanto, isto não anula o fato de existirem plataformas e de estarem sendo aplicadas a plena produção no Web Design. Reunidos adiante, estão alguns aplicativos na Web que comportam-se claramente como plataformas (através da modificação de alguns componentes variáveis, resultam em novos projetos). Entretanto, uma análise mais detalha dos componentes de plataformas específicas de Web Design, ficará alocada nos estudos de caso.

3.4.2 Resultados da aplicação das plataformas no Web Design

O crescimento em escala geométrica da Web e o advento de novas tecnologias concebidas no espaço cada vez mais curto de tempo representam uma mistura “explosiva” que gera múltiplas novas espécies de plataformas continuamente. Diante das limitações do presente projeto de pesquisa, torna-se inviável, portanto, uma completa taxonomia das estruturas da Web que comportam-se como plataformas. Todavia, existem categorias de websites concebidos neste paradigma de projeto e que estão consolidados. Os esforços na presente pesquisa concentraram-se em identificar algumas estruturas da Web com relativa expressividade, com o objetivo de ilustrar que o design baseado em plataforma está consolidando-se no Web Design e trazendo ao setor as contribuições que este paradigma de projeto agrega.

Faz-se necessário destacar que existe um conceito de projeto no Web Design denominado CMS (“*Customer Management System*”). Tratam-se de plataformas nas quais é possível o “cliente” (usuário) construir websites completamente customizáveis por ele

mesmo, abstraindo-se de toda a complexidade das aplicações rodando em *background*, alterando os componentes variáveis da plataforma, como ocorre nos Blogs que estão considerados a seguir. Entretanto, não são todas as plataformas da Web que são empregadas como CMS. Outras objetivam prover um recurso específico, de complexidade tecnológica relativamente alta, encapsulá-lo numa plataforma e disponibilizá-lo para encaixe no website do usuário. Por como exemplo os aplicativos de Transmissão de Vídeo.

Com isto pretende-se, também, ilustrar o quão diversificado podem ser as estruturas e categorias de plataformas presentes na Web, conforme detalhado abaixo.

3.4.2.1 Blogs, Fotologs e Videoblogs

São uma alternativa rápida para quem deseja publicar textos, fotos e vídeos, respectivamente. Em algumas plataformas são disponibilizados apenas alterações nas cores da tipografia, textos e plano de fundo, como no *Fotolog* (<http://www.fotolog.com>). Em outros casos, aplicações avançadas permitem projetos mais diversificados, como é o caso do *Wordpress* (<http://www.wordpress.org>). Sobre esta plataforma específica incursaremos uma análise em nossos estudos de caso.

3.4.2.2 Templates

Por um lado detentores de uma considerável má fama entre os web designers, por outro atendem uma grande demanda de solicitações, possuindo diversos portais dedicados à comercialização dos mesmos. Os *templates* são a expressão mais pobre das plataformas na Web. Tratam-se de websites prontos, com toda a estrutura disponível para venda. Alterando-se alguns componentes variáveis como fotos, textos e títulos do *menu*, tem-se o website configurado para a empresa que comprou e poucas vezes será reutilizado. É uma alternativa expressa para quem quer ver o produto pronto antes de comprar.

3.4.2.3 Fóruns

Existe uma febre de interatividade entre os internautas da Web 2.0 através dos fóruns, que são regiões no ciberespaço para discussões divididas por temas. Convém destacar a plataforma *phpBB* (<http://www.phpbbforuns.com>), que é “*freeware*”. Esta, compõe-se de um fórum onde é possível abstrair-se do design e implementação do banco de dados, bem como das centenas de aplicações PHP e *javascript* rodando em “*back-end*”, para disponibilizar ao designer os componentes variáveis que permitem, da mesma plataforma, criar diversos fóruns através da modificação de tipografia, cores, formatos, ícones e demais recursos.

3.4.2.4 Transmissão de Vídeo

O nome já esclarece bem o que este tipo de aplicativo faz. Seria impossível não citar o *Youtube* (<http://www.youtube.com>), o maior fenômeno em transmissão de vídeo “*freeware*” da Web atualmente. Do ponto de vista de projeto, o *Youtube* oferece um código referente à plataforma modificada, com o vídeo do usuário. Adaptando este código ao website final, tem-se uma aplicação de transmissão de vídeo pronta. Toda a complexidade envolvida neste aplicativo está abstraída em seus componentes fixos e o usuário tem em seu website uma avançada aplicação de vídeo, como se ele próprio a tivesse concebido.

3.5 Reuso de Soluções e Transferência de Conhecimento

Se o projeto não é destinado ao posterior reuso, não é uma plataforma.
(FIGUEIRÔA et al., 2008)

Diferentemente dos padrões, que transferem o conhecimento através de um procedimento didático no qual o designer que o lê aprende sobre aquela experiência de projeto, propiciando seu reuso noutro contexto, as plataformas atendem esta demanda de modo bastante peculiar. Ao projetar uma plataforma, o designer “embute seu conhecimento no projeto”, mais precisamente nos componentes fixos. Quem reutilizar esta plataforma, abstraindo-se do conhecimento utilizado para a construção dos componentes fixos, produz outro projeto.

A impressão, ao ver o produto final, muitas vezes é de que o designer que o projetou o concebeu por inteiro. Enquanto que, na verdade, parte do conhecimento foi “embutido e transferido no produto”. Desse modo, podemos afirmar que é através da abstração que o conhecimento é transferido. Os componentes fixos, com os quais o designer que reaproveita o projeto não precisa ter conhecimentos aprofundados, são reutilizados.

Recorramos ao já citado exemplo do Ford Ecosport e do Ford Fiesta (Vide Figura 5). A princípio são carros completamente distintos, mas reutilizaram a mesma plataforma. Não podemos afirmar, baseado em mera observação, qual dos dois projetos foi concebido primeiro. Foi o designer do Ford Ecosport quem construiu a plataforma e alguém reaproveitou esse conhecimento para conceber o Ford Fiesta, ou será que foi ao contrário? Ocorrência semelhante observaremos na plataforma que concebeu o game “*Half-Life*”. Milhões de jogadores do “*Counter-Strike*” (modificação desta plataforma) desconhecem o game primeiramente concebido, conforme ilustrado anteriormente.

Estes exemplos reforçam a afirmação de que o paradigma de projeto baseado em plataforma comporta o conhecimento transferindo-o no produto. Permitindo que o designer que reutiliza a plataforma crie projetos completamente originais como se ele tivesse todo o conhecimento necessário para fazê-lo, enquanto que na verdade só possui no que tange a modificação dos componentes variáveis (Vide Figura 15).

A conceituação de plataforma, portanto, aproxima-se dos conceitos de “know how” (componentes variáveis) e “know why” (componentes fixos). Uma vez que quem projeta os componentes variáveis entendem pra que função estão fazendo isso mas abstraem-se do funcionamento dos componentes variáveis. Quem projeta os componentes variáveis não sabe pra que função final irão modificar sua plataforma, mas sabe como funciona sua estrutura interna.

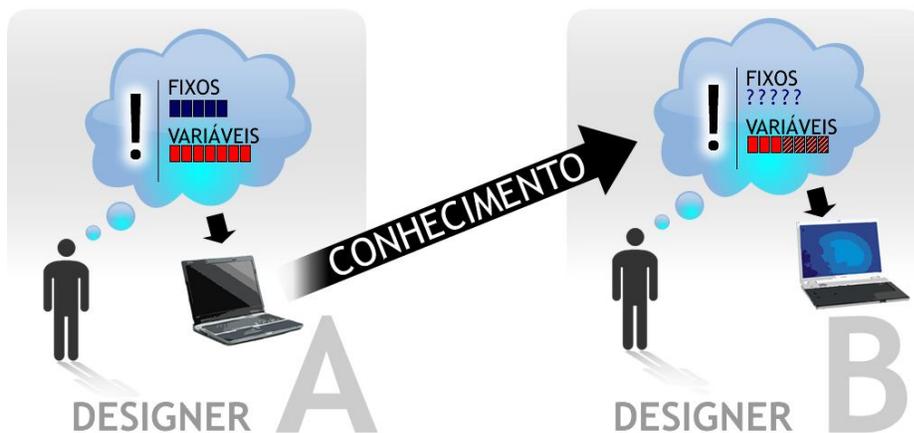


Figura 15 - Plataformas provendo abstração

4 ESTUDOS DE CASO

Após conhecer mais detalhadamente o que são padrões, plataformas e como os mesmos atuam em diversos domínios, foram aplicados os conhecimentos adquiridos desta revisão bibliográfica em experimentos. Validando, na prática, as informações coletadas e permitindo um contato mais efetivo com os paradigmas projetuais em estudo. Gerando, assim, informações mais ricas sobre a aplicação dos mesmos no reuso de soluções e transferência de conhecimento.

Haja visto que encontram-se apresentados múltiplos estudos de caso em áreas com uma série de particularidades, encontra-se disponibilizada uma seção de exposição sintetizada de resultados e/ou breves considerações ao final de cada um deles. Vinculando, desta forma, as particularidades de cada experimento aos objetivos pontuais de nossa pesquisa. Convém destacar que cada estudo de caso, individualmente, possui em sua introdução os objetivos específicos e o procedimento adotado. O destrinchamento detalhado de cada um destes atributos de pesquisa tornaria a seção de estudos de caso demasiadamente longa, extrapolando as limitações do nosso projeto. Assim sendo, foi-se atribuído o presente formato visando transmitir as informações essencialmente úteis de cada estudo de caso sem desviar-nos do foco da pesquisa.

Foram realizados 4 estudos de caso sobre plataformas (destes, 2 são no Game Design e 2 no Web Design) e 2 estudos de caso sobre padrões (todos no Web Design). Este contraste se dá por duas razões em evidência. Primeiramente, já existe uma vasta bibliografia sobre padrões, enquanto que sobre plataformas detectei uma carência informativa que vai desde a conceituação a resultados de experimentos (uma vez que nenhum livro sobre plataformas foi encontrado em na revisão bibliográfica). Em segundo lugar, considerou-se as limitações temporais do presente projeto de pesquisa, concentrando, portanto, esforços com o intuito de contribuir onde existe maior demanda. Isto não desaprecia, de maneira alguma, a importância dos resultados com os padrões, uma vez que está sendo exposta especificamente sua eficácia no reuso de soluções e transferência de conhecimento, uma abordagem relativamente nova a este paradigma de projeto consolidado.

4.1. Plataformas

Os estudos de caso sobre plataformas consistem numa avaliação das mesmas no Game Design e no Web Design, através de um estudo que engloba o destrinchamento e a exposição dos seus componentes, bem como a demonstração prática disto manipulando estes componentes na geração de games e websites experimentais. Ao final do estudo de cada plataforma observada, considerando todas as informações levantadas e expostas, encontra-se apresentado um quadro com a avaliação da mesma na tentativa de expressar sua disponibilidade em prover reuso de soluções e transferência de conhecimento.

4.1.1 SBGames MOD

Durante esta fase da pesquisa ainda não havia sido considerado os sons do game como um componente variável. O objetivo principal é demonstrar que, através do manuseio dos componentes variáveis, obtem-se reuso de das soluções dos componentes fixos e que “conhecimento embutido no projeto” foi transferido pela abstração oferecida pela plataforma “Source”. O objetivo secundário deste estudo de caso é demonstrar a viabilidade da estrutura de plataforma proposta na seção 3.1 Estrutura das Plataformas.

Para o experimento foram utilizados os programas “Source SDK” e “VTFEdit” para modificar a plataforma “Source”, produzida inicialmente para o game *Half-Life 2*, um dos maiores sucessos do mundo em seu segmento. Conforme o modelo de plataforma proposto, não foi modificado nada do engine, o qual representa o componente fixo. Entretanto seus recursos, tais como aplicações de rede e física dos objetos foram explorados pelos componentes variáveis: objetos, cenários e GUI. Neste procedimento concebeu-se um novo game de categoria “MOD”, a partir dessa plataforma.

4.1.1.1 Introduzindo a Plataforma Escolhida: Source

A plataforma “Source”, que foi desenvolvida pela *Valve Corporation* por cerca de \$40 milhões de dólares, foi inicialmente projetada para o clássico da indústria de games para PC denominado “*Half-Life 2*” (BRAMWELL, 2007). Contudo, atualmente ela pode ser comprada por \$54 dólares, comprando-se o referido game. Este, é um sucesso desde sua primeira versão, “*Half-Life*”, que vendeu cerca de 8 milhões de cópias desde seu lançamento em 1998. Travou algumas batalhas judiciais contra hackers que roubaram o código-fonte de seções restritas do programa (componentes fixos), com direito a intervenção do FBI sobre os infratores. Dando continuidade à saga, o HL2 vendeu 1.7 milhão de cópias em menos de 2 meses de lançamento.

Pra se ter uma idéia da grande variedade de “MOD”s gerados a partir da plataforma “Source”, apresenta-se sinteticamente alguns exemplos nas Figuras 16, 17, 18 e 19:



Figura 16 – “Insurgency”: Batalhas entre americanos e iraquianos

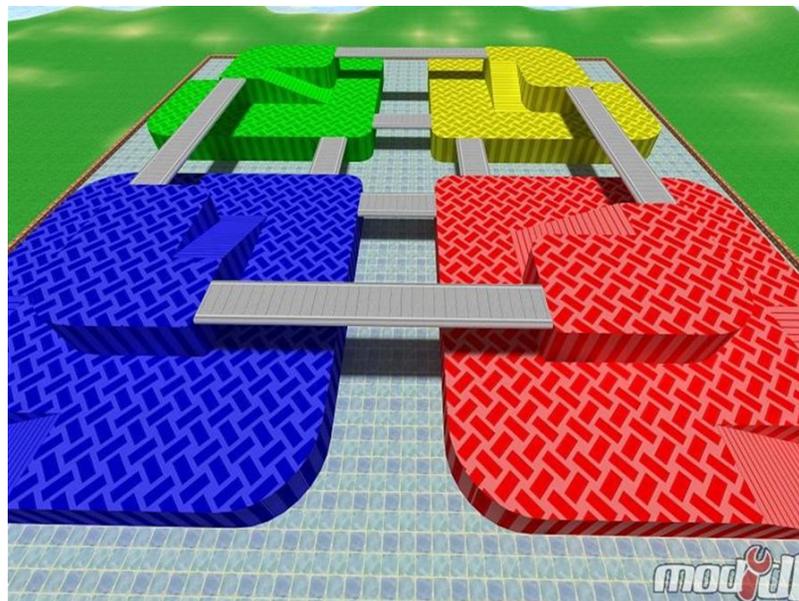


Figura 17 - "Mario Kart: Source": game de corrida



Figura 18 - "Dragonball: Source"



Figura 19 - "Enterprise" do seriado "Startrek"

Com esses poucos exemplos fica explícita a grande flexibilidade da plataforma “Source”, bem como os benefícios do paradigma de projeto baseado em plataforma. Chega-se o ponto em que um FPS é transformado num game de corrida, manuseando os componentes variáveis intencionalmente disponibilizados pela plataforma.

Milhares de outros “MOD”s “freeware” estão disponibilizados, ou em desenvolvimento, sendo compartilhados em websites especializados na internet.

4.1.1.2 Reusando a Plataforma “Source” para criação do “SBGames MOD”

“SBGames MOD” é o nome do game “criado” neste estudo de caso. Por questões de limitação de tempo de desenvolvimento, ele não possui tantos detalhes e refinamentos quanto os exemplos mostrados acima (que demandam meses de desenvolvimento de uma

equipe competente). O propósito aqui restringe-se ao acesso e modificação de componentes variáveis, apenas, ilustrando exatamente como isso é possível.

Iniciando o desenvolvimento, acessou-se o "Source SDK" (vide Figura 20) e utilizou-se o programa "Hammer Editor", no qual é possível construir cenários e importar objetos para plataforma.



Figura 20 - Tela principal do "Source SDK"

Foram construídos alguns cenários utilizando a biblioteca de texturas já presente no programa, conforme ilustrado na Figura 21.

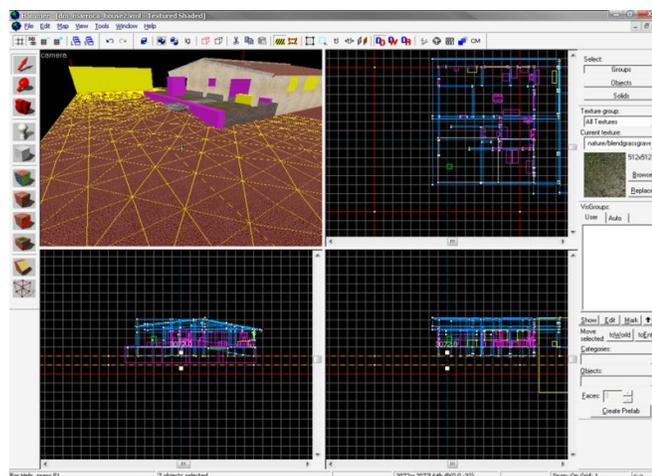


Figura 21 - Construindo cenário no "Hammer Editor"

Quanto aos objetos, existem duas categorias: objetos espalhados pelo cenário e personagens. São construídos em softwares 3D externos, como o “3D Studio Max” ou “Maya”, e então importados pelo “Hammer Editor”. Contudo, reutilizou-se neste experimento a riquíssima biblioteca de objetos do “Half-Life 2” (Figura 22), posicionando os mesmos pelo cenário com relativa facilidade.

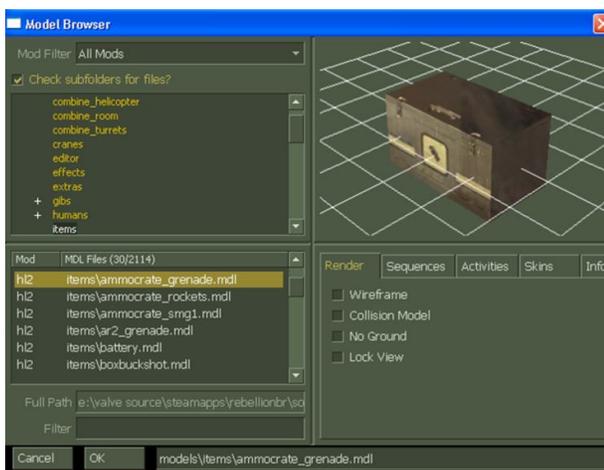


Figura 22 - Importando e posicionando objetos no "Hammer Editor"

Para edição do comportamento dos personagens, foi utilizado o software “Face Poser”, incluso no SDK. Este programa também é empregado como visualizador de objetos e seus movimentos.

Recorrendo ao software “VTFEdit”, não incluso no citado SDK, foram realizadas no mesmo alterações diversas na Interface Gráfica, como plano de fundo. Acessado o arquivo “gameinfo.txt”, presente no pacote de aplicações, foram realizados ajustes os itens do menu do game, apresentado na Figura 23.



Figura 23 - GUI inicial do "SBGames MOD"

Finalmente, foi-se acessada a opção "Create a MOD" do "Source SDK" para criar o "SBGames MOD", definindo sua pasta de arquivos e posicionando os arquivos acima desenvolvidos nela. Nas Figuras 24 e 25 encontram-se ilustrados os testes do game concluído.



Figura 24 - Jogando o "SBGames MOD" em rede



Figura 25 - Efeitos de explosão do "Source Engine" (componente fixo)

4.1.1.3 Resultados e Breves Considerações

Neste estudo de caso, foi ilustrado como em menos de 48 horas de trabalho é possível produzir-se um game de relativa sofisticação utilizando o paradigma de projeto baseado em plataforma. Ao desenvolver este projeto, tornaram-se mais claras as contribuições deste paradigma ao Game Design, pois os “*stakeholders*” diretamente envolvidos são beneficiados de diferentes formas. Os usuários usufruem de grande variedade de games, os desenvolvedores ganham uma eficiente reutilização de projeto, e os comerciantes (“*publishers*”) fermentam as vendas de suas plataformas de games com os novos “MOD”s, “add-on”s ou pacotes de expansão disponibilizados.

Espera-se ter ficado evidente o quanto um projeto concebido como plataforma auxilia o reuso de soluções através o reuso do componente fixo e ao mesmo tempo oferece a transferência de conhecimento “embutindo-a no produto”. Afinal, o “*SBGames MOD*” foi desenvolvido abstraindo-se de toda a extrema complexidade do engine de quase \$40 milhões de dólares.

Uma avaliação relativa à plataforma “*Source*” está disponibilizada no estudo de caso seguinte, que desdobra os experimentos na mesma.

4.1.2 P&D MOD

Foi neste desdobramento do “*SBGAMES MOD*” que chegou-se à estrutura de plataforma de games descrita na seção 3.3.1 Modelo de Plataforma de Games (veja Figura 8). Nele, foi

identificado outro componente variável comumente presente nas plataformas dos games estilos FPS e RTS (“*First Person Shooter*” e “*Real-time Strategy*”): os sons.

Para realização deste experimento, que objetiva demonstrar a viabilidade da estrutura de plataforma com sons proposta, utilizou-se novamente a “*Source*”, bem como os programas “*Source SDK*” e “*VTFEdit*”. Mais uma vez, não foi modificado nada do componente fixo, o engine, porém, editados os componentes variáveis, compreendidos entre objetos, cenários, GUI e sons, gerando outro game. 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.

Não encontra-se nesta seção a descrição da plataforma “*Source*” em datalhes, haja visto que isto já fora realizado na seção 4.1.1.1 Introduzindo a Plataforma Escolhida: *Source*. Os procedimentos de desenvolvimento do novo game também possuem semelhanças com o anterior, porém alguns incrementos como o desenvolvimento de um cenário totalmente novo e a modificação de sons.

Ao final da descrição deste experimento, somando aos resultados obtidos no estudo de caso que gerou o “*SBGames MOD*”, encontra-se apresentada uma avaliação da plataforma “*Source*”.

4.1.2.1 Reusando a Plataforma “*Source*” para criação do “*P&D MOD*”

Seguindo o roteiro desenvolvido pela pesquisa do “*SBGames MOD*”, iniciou-se o desenvolvimento do “*P&D MOD*” acessando o programa “*Source SDK*” onde obtem-se acesso a uma série de aplicativos para modificação da plataforma “*Source*” (Figura 20). Então, com o programa “*Hammer Editor*” (Figura 26), foi criado um cenário para o game.

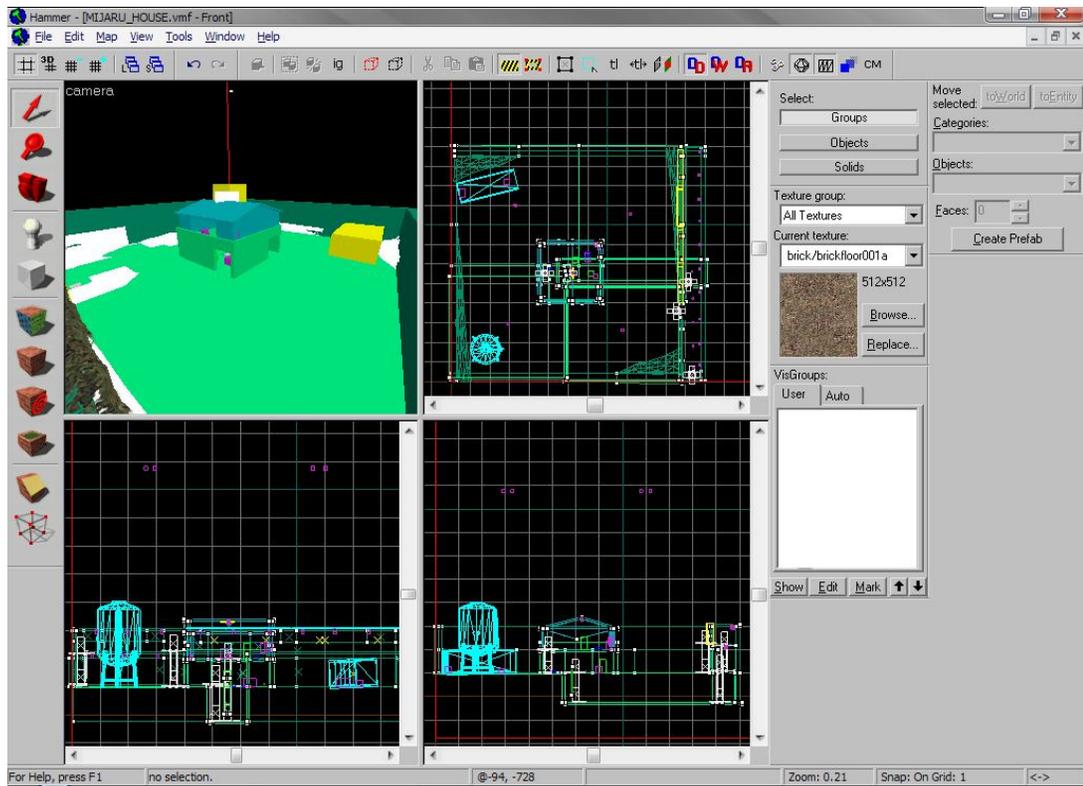


Figura 26 - Construindo o cenário no "Hammer Editor"

Novamente, foi reutilizada a biblioteca de objetos do "Half Life 2", posicionando este tipo de componente variável por dentro do cenário com auxílio dos softwares do SDK (Figura 27).

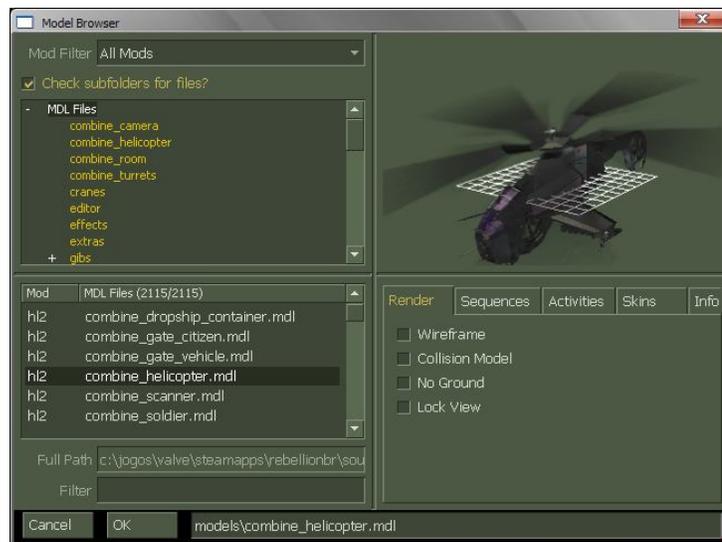


Figura 27 - Selecionando e inserindo objetos no cenário

Através do “VTFEdit” realizou-se modificações na Interface Gráfica. Editando o “background”, ou tela de fundo, da interface inicial do “P&D MOD”. Modificou-se ainda o arquivo “gameinfo.txt” para ajustar o menu principal do mesmo, conforme ilustrado na Figura 28.



Figura 28 - GUI (componente variável) alterado

A edição do componente variável “som” nesta plataforma não contou com um programa de auxílio, como encontraríamos incluído na plataforma “*Warcraft III*” outrora mencionado. Fez-se necessário entrar nas pastas de arquivos dos objetos que interagiriam com o som modificado e alocar lá o arquivo de áudio desejado. Assim, editando manualmente o arquivo “*game_sound_weapons.txt*” foi ajustado quais ações ou objetos disparariam determinando som. Os que baixarem e instalarem o “P&D MOD” (no endereço eletrônico citado no início desta seção) perceberão que o áudio diferenciado da arma “*Stunstick*”.

Para conclusão deste protótipo, foi acessado as opções do “*Source SDK*”, em específico “*Create a MOD*” para criar o “P&D MOD”, ajustando sua pasta de arquivos. Nas Figuras 29 e 30 as telas novo game personalizado em funcionamento.



Figura 29 - Jogando o "P&D MOD" em rede local



Figura 30 - Reutilizando a física do "Source Engine"

4.1.2.2 Resultados e Breves Considerações

O "P&D MOD" foi concebido no intervalo de 72 horas, sendo 70% do tempo investido na construção do cenário (incluindo o posicionamento dos objetos). O tempo seria estendido caso os objetos também fossem desenvolvidos, ao invés de reutilizar a biblioteca do "Half-Life 2". A atividade de modelação gráfica, portanto, é o que demanda maior tempo e, de certa forma, é o que restou de atividade exigindo maiores habilidades, uma vez que o SDK facilita a modificação dos componentes da plataforma. O esforço de edição do "novo" componente, o som, foi ainda menor do que os objetos e cenário, demandando apenas alguns minutos.

Este experimento vem reafirmar que o tempo investido no desenvolvimento dos componentes variáveis está em extremo contraste com o esforço de desenvolvimento do engine, o componente fixo, demonstrando o potencial do paradigma de projeto baseado em plataforma para rápido reuso de soluções.

Com base no experimento do “SBGames MOD” e do “P&D MOD” foi gerada uma avaliação sintetizada da plataforma “Source”. No Quadro 2 encontram-se disponibilizados valores viabilizando expressar o *feedback* quanto a esta plataforma. Neste quadro, e nos demais que seguem nas avaliações das outras plataformas, os “Recursos Disponíveis” estão diretamente relacionados com o porte dos componentes fixos, ou seja, à quantidade de aplicações que a plataforma disponibiliza para reuso. Quanto ao valor atribuído à “Abstração”, pontua-se a abstração provida ao designer diante dos componentes fixos – que são reutilizados através do manuseio dos componentes variáveis. Quanto maior a abstração, maior a quantidade de conhecimento “embutido no produto” disponível para ser transferido de projeto para projeto.

Assim sendo, os quadros disponibilizados nos estudos de caso sobre plataformas (Vide Quadro 2, Quadro 3 e Quadro 4) referem-se a uma avaliação das mesmas quanto ao reuso de soluções e transferência de conhecimento, dentro do modelo apresentado na Figura 6.

<u>DADOS DA PLATAFORMA</u>	
Plataforma	Source
Fabricante/Proprietário	Valve Corporation
Reuso	Comercial
<u>AVALIAÇÃO</u>	
Recursos Disponíveis	★ ★ ★ (3 de 3)
Abstração	★ ★ ☆ (2 de 3)

Quadro 2 - Avaliação da Plataforma "Source"

Conforme podemos observar no Quadro 2, a plataforma “Source” disponibiliza um grande porte de aplicações disponíveis. O website do seu fabricante chega a afirmar que seu componente fixo (engine) “é reconhecido com o mais flexível, compreensível e poderoso ambiente de desenvolvimento de games disponível” (VALVE, 2007). Lembramos que o “Source Engine” é um encapsulamento de vários engines.

No que tange a abstração da plataforma em questão, foi pontuado com *score* 2 de 3 pois, apesar de prover um reuso impressionante de uma plataforma que engloba os recursos computacionais mais modernos da atualidade, necessita-se de um conhecimento técnico mínimo para manuseio dos componentes variáveis. Em outras palavras, para produção de um “MOD” profissional, é necessário possuir conhecimentos técnicos de um profissional de Game Design. E estes conhecimentos seriam do designer que reutiliza a plataforma, e não estão “embutidos” na mesma, disponíveis ao reuso.

4.1.3 Blogger

Introduzindo os estudos de caso no campo do Web Design, foi realizado inicialmente a observação de uma plataforma. Haja visto que, conforme mencionado, as plataformas na Web assumem estruturas demasiadamente diversificadas, a presente pesquisa realizou a verificação de uma plataforma específica, com o objetivo de identificar quais são os seus componentes e demais características. A saber, componentes fixos, componentes variáveis, capacidade de abstração e reuso. Escolheu-se, portanto, o *blog* do jornalista Michelson Borges, analisando como a partir da plataforma *Blogger* (<http://www.blogger.com>) ele modificou os componentes na geração do seu site (Veja Figura 31).



Figura 31 - Blog de Michelson Borges: Plataforma "Blogger"

Para uma maior precisão na investigação dos componentes, foi acessado ainda o *Blogger* e criado um *blog* experimental. O website oferece um assistente que, em 3 passos, modifica a plataforma e cria um *blog* personalizado (Veja Figura 32). Sendo possível fazer

mais modificações posteriormente. Ao longo destes passos coleta-se informações relativas à identificação do usuário, título do *blog*, endereço online, dentre outros. Em determinado momento do segundo passo, o *Blogger* disponibiliza alguns *templates* para que o usuário decida qual o modelo visual base do produto final. Porém, ao contrário dos *templates* comerciais, estes são mais flexíveis quanto à personalização, permitindo a criação de sites mais originais.

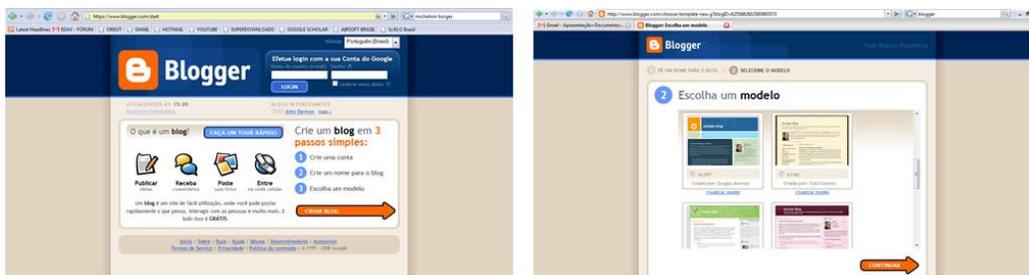


Figura 32 - Assistente de Modificação da Plataforma "Blogger"

Em posse destas informações, retomou-se a análise do *blog* de Michelson Borges identificando quais componentes ele modificou. Desse modo foram coletadas, como resultados deste estudo de caso, as características da plataforma Blogger descritas na seção a seguir.

4.1.3.1 Resultados e Breves considerações

A observação da plataforma *Blogger* nos resultou nos componentes e características abaixo reportados.

- **Componentes fixos:** CSS (*"Cascading Style Sheet"*) com várias tipografias, cores e configurações; aplicações de vídeo; aplicações de tratamento de imagem; sistema de busca interna; arquiteturas de informação e interface disponíveis; estruturas do banco de dados; outros inacessíveis. Não foi possível destrinchar todos os componentes fixos, pois a abstração envolvida é alta (Ponto positivo para a plataforma). De modo que algumas aplicações, como o próprio banco de dados, roda do lado do servidor, dentro da plataforma, não sendo possível o acesso.
- **Componentes variáveis:** imagens, textos, vídeos, plano de fundo, configuração da página escolhida, título, endereço eletrônico, informações do usuário (dono do *blog*).
- **Abstração:** quem edita os componentes variáveis abstrai-se da complexidade do desenvolvimento das modernas aplicações dispostas nos componentes fixos. Ao editar a tipografia do título do *blog*, por exemplo, o autor está abstraindo-se dos recursos CSS que estão provendo aquela formatação.

- **Reuso:** milhares de blogs com base na plataforma *Blogger* estão em circulação na Web (Figura 33).

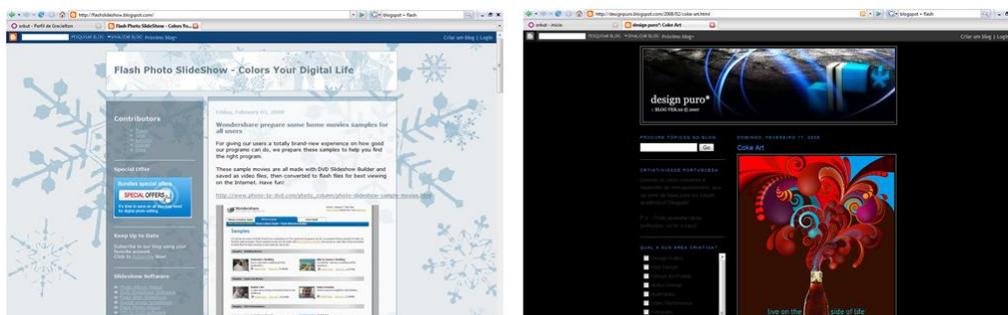


Figura 33 - Websites reusando a Plataforma "Blogger"

Pode-se considerar que as plataformas apresentam suas limitações de forma muito clara. Conforme outrora reportou o próprio Goering (2002), elas não representam um modelo mágico no qual é possível gerar uma infinidade de projetos totalmente disjuntos. Assim sendo, convém de ressaltar certa limitação da plataforma *Blogger*, no aspecto de flexibilidade. No caso das plataformas de games, às vezes, não é possível distinguir que um game e outro estão usando os mesmos componentes fixos, devido à diversidade dos gráficos e “gameplay” de ambos. Entretanto, tratando-se de websites gerados pelo *Blogger*, os projetos podem ser julgados um tanto “engessados”, pois as modificações se parecem por demais. Sendo necessário balancear, portanto, este paradigma de projeto com outras alternativas disponíveis.

Em contrapartida, talvez pelos benefícios da elevada abstração, esta plataforma é um sucesso na Web, chamando tanto a atenção que foi comprada pela *Google Inc*. Os usuários aparentam optar por uma plataforma não tomando por prioridade sua flexibilidade (que seria provida por uma grande quantidade de componentes fixos), mas pela facilidade em construir seu *blog*. Em outras palavras, o critério principal que aparentemente leva os usuários da plataforma *Blogger* a optar pela mesma seria a abstração provida. Por exemplo, quanto o jornalista Michelson Borges gastaria, em tempo e dinheiro, com web designers experientes para chegar a um projeto onde fosse possível publicar seus estudos? Levando em consideração a facilidade de publicar essas informações e inserir fotos anexadas com rapidez. Além disso, seria necessário investir em estudos de arquitetura da informação para que a massa informacional fosse exposta aos visitantes com a melhor usabilidade possível. Lembrando que a aparência dos sites gerados pela plataforma *Blogger* é de um produto feito por profissionais.

O conhecimento sobre tudo isto foi transferido e “embutido” no projeto. Michelson Borges publicou suas informações como se ele próprio possuísse os conhecimentos de design, acima descritos, para fazê-lo. Não está sendo afirmado que ele não os possuía, porém não se fez necessário investir tempo no projeto do website, mas apenas na escolha de qual plataforma (de *blogs*) o atenderia melhor. Disponibilizando, desse modo, mais tempo e energia para serem empregados nos seus estudos pessoais e publicações.

Diante destas considerações, encontra-se disposto o resultado da avaliação da plataforma Blogger no Quadro 3 abaixo.

<u>DADOS DA PLATAFORMA</u>	
Plataforma	Blogger
Fabricante/Proprietário	Pyra Labs/Google
Reuso	Gratuito/Público
<u>AVALIAÇÃO</u>	
Recursos Disponíveis	★☆☆ (1,5 de 3)
Abstração	★★★ (3 de 3)

Quadro 3 - Avaliação da Plataforma "Blogger"

Devido à baixa flexibilidade (resultado de uma possível restrição de componentes fixos que possibilitariam uma maior diversidade de reutilizações) foi pontuado com *score* 1,5 os “*Recursos Disponíveis*” (Vide Quadro 3). Em contrapartida, talvez justamente por esta relativa simplicidade dos componentes fixos, o nível de abstração seja tão alto. Não é necessário absolutamente nenhum conhecimento de Web Design para criação do seu *blog* personalizado. O maior nível de abstração de todas as plataformas avaliadas pela presente pesquisa foi encontrado na plataforma *Blogger*. Em outras palavras, o conhecimento transferido por estar “embutido no produto” é elevado tal de modo que qualquer amador pode construir um website profissional reusando as soluções providas por esta plataforma.

4.1.4 Wordpress

Na busca por plataformas de Web Design, na categoria *blog*, que provenham uma maior flexibilidade chegou-se ao *Wordpress*. Os resultados do uso desta plataforma são bastante diversificados, variando de websites pessoais ao website oficial do Ministério da Cultura (Ver Figura 34).



Figura 34 - Website do Ministério da Cultura: Plataforma “WordPress”

O objetivo geral na realização deste procedimento é a verificação da disponibilidade do *WordPress* em prover reuso de soluções e transferência de conhecimento, considerando os recursos disponíveis para reuso e a capacidade de abstração envolvida, que por sua vez provê “conhecimento embutido”. Em outras palavras, demonstrar que o *WordPress* é uma plataforma. Este estudo de caso dividiu-se em dois momentos. Primeiramente foi modificado a plataforma *WordPress* e gerado um *blog* experimental com o intuito de demonstrar o manuseio dos componentes variáveis. Num segundo momento, foi observado como uma turma de 15 alunos, do 4º período da graduação em Web Design/Sistemas para Internet da Faculdade Marista do Recife, utilizava a plataforma para realização de um exercício em sala de aula. Este exercício consistia em reusar o *WordPress* para construção de um website à livre escolha, porém sendo sugerido um *blog* pessoal. Tudo isto verificando-se o tempo decorrido para construção destes websites bem como os recursos mais reutilizados.

Deste modo, o objetivo específico deste estudo de caso é detectar quais recursos de projeto foram mais reutilizados do *WordPress* e a redução estimada de tempo de projeto em comparação ao uso destes mesmos recursos caso fossem desenvolvidos “do zero”. Simultaneamente faz-se possível observar a flexibilidade da plataforma em estudo, durante a geração de múltiplos websites. Com isto seria possível demonstrar de forma mais completa os benefícios do projeto baseado em plataforma.

O procedimento iniciou-se na observação desta plataforma através da criação de um website experimental. Ao acessar o portal *Wordpress Brasil* (<http://br.wordpress.org>) existem instruções para “Instalação em 5 minutos”. Esta instalação consiste em baixar na íntegra a plataforma para o seu PC, alterar alguns componentes variáveis, e enviá-la de volta para o servidor online. Entretanto, se fez necessário criar manualmente um banco de dados para que a plataforma *Wordpress* realizasse sua auto-instalação neste banco. Após estes passos a modificação da plataforma no modelo básico está criada, tomando-me um total de 15 minutos.

O que percebi de imediato é que seria necessário investir um certo tempo aprendendo sobre o *Wordpress* (sobre seus componentes variáveis). E isto serve para qualquer usuário que tentar criar seu site reutilizando esta plataforma. Ou seja, o programa administrador fornecido pelo *Wordpress* (uma espécie de SDK) não abrange todos os componentes variáveis. Fez-se necessário acessar manualmente alguns arquivos específicos (como no caso da plataforma “*Source*”), principalmente o “*index.php*” (arquivo que é a página principal) e o “*style.css*” (arquivo que edita a tipografia) para realizar maiores modificações.

Acessando o “SDK” que fica na pasta “*/wp-admin*”, os arquivos modificáveis que ficam na pasta “*/wp-content/themes*” e editando seus arquivos (com conhecimento técnico de Web Design) foi-se gerada a modificação abaixo (Figura 35).



Figura 35 - Website experimental gerado pelo reuso da Plataforma "WordPress"

Para reverificação destas informações, foi aplicado este mesmo experimento utilizando os alunos de Web Design conforme supracitado. Os resultados da aplicação deste estudo de caso estão dispostos na seção seguinte.

4.1.4.1 Resultados e Breves Considerações

O produto final gerado pela plataforma *WordPress* apresenta relativa sofisticação, incluindo sistema de busca interna. O mesmo foi produzido num intervalo de 72 horas, entretanto pelo menos metade deste tempo foi dedicado em aprender a manusear os componentes variáveis. Sendo necessário conhecimentos de Web Design por parte do usuário para realização do projeto, reduzindo muito a abstração envolvida apesar de existir bastante “conhecimento embutido” nos componentes fixos. Por exemplo, existe a necessidade de saber não só saber o que é um banco de dados, mas ter habilidade para criar um. Este conhecimento não é de posse de qualquer amador. Apesar de tudo, conforme ilustrado anteriormente, com alguma dedicação, profissionais do setor de Web Design podem ter substancial redução de tempo e riscos de projeto especializando-se no reuso da plataforma em questão.

Quanto aos websites experimentais gerados pelos alunos, estes foram concebidos num intervalo de menos de 48 horas, oscilando de 1 a 7 horas de trabalho contínuo (Alguns

projetos eram mais complexos e bem compostos que outros). O conhecimento sobre como instalar e reusar a plataforma *Wordpress* disseminou-se rapidamente entre eles. Os resultados gerados incluíam, em sua maioria, recursos como calendário, *links*, transmissão de vídeo, e busca interna. Além, é claro, da publicação de notícias, característica central dos websites categoria *blog* (Ver Figura 36). Recursos considerados avançados no que tange o conhecimento técnico e que se fossem desenvolvidos “da estaca zero”, certamente a construção de um mesmo website, com os mesmos recursos, precisaria de pelo menos 10 vezes mais tempo. Isto considerando ainda que os profissionais que o desenvolveriam teriam conhecimento técnico sobre cada um destes aplicativos que envolvem conhecimentos disjuntos.

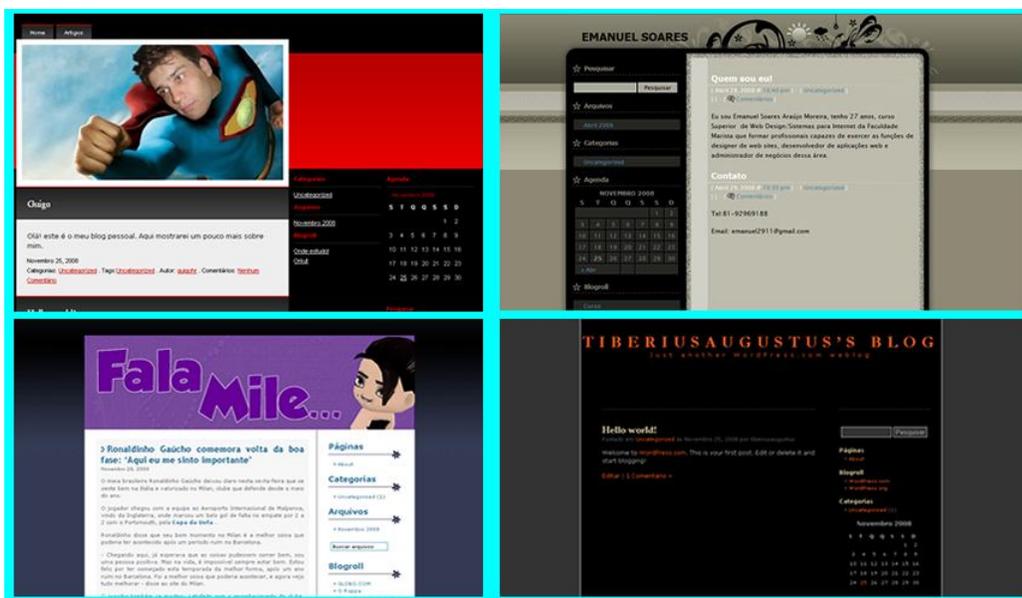


Figura 36 - Websites concebidos pelos alunos reusando a Plataforma "Wordpress"

A abstração envolvida pela plataforma *Wordpress* é evidente, porém a necessidade de aprender a manusear a mesma reduz a quantidade de conhecimento “embutido no produto” e passa a ser do usuário que, sendo um amador, encontrará dificuldades de fazer um website mais complexo. Afinal, raramente estes usuários pertencem a uma turma de web designers, localizados num mesmo espaço físico, disseminando o conhecimento entre eles.

Sendo assim, a flexibilidade do *Wordpress* é elevada, porém a abstração foi julgada deficiente. O que vem a contrastar-se com a elevada abstração e baixa flexibilidade do *Blogger*. Ao que parece, estas características, que estão respectivamente relacionadas à facilidade de manuseio dos componentes variáveis e quantidade de componentes fixos, são

inversamente proporcionais. Assim sendo considerou-se a avaliação da plataforma *Wordpress* no Quadro 4.

DADOS DA PLATAFORMA	
Plataforma	Wordpress
Fabricante/Proprietário	Matt Mullenweg Ryan Boren Donncha O Caoimh
Reuso	Gratuito/Público
AVALIAÇÃO	
Recursos Disponíveis	★ ★ ★ (3 de 3)
Abstração	★ ★ ☆ (2 de 3)

Quadro 4 - Avaliação da Plataforma "Wordpress"

Encerrando os experimentos com plataformas, com o intuito de facilitar a visualização dos resultados das mesmas, encontra-se disponibilizado abaixo o Quadro 5 com o comparativo das avaliações das plataformas.

Plataforma	Source
Recursos Disponíveis	★ ★ ★ (3 de 3)
Abstração	★ ★ ☆ (2 de 3)
Plataforma	Blogger
Recursos Disponíveis	★ ☆ ☆ (1,5 de 3)
Abstração	★ ★ ★ (3 de 3)
Plataforma	Wordpress
Recursos Disponíveis	★ ★ ★ (3 de 3)
Abstração	★ ☆ ☆ (1 de 3)

Quadro 5 - Comparativo das avaliações das Plataformas

4.2 Padrões

Conforme justificado no início deste capítulo, no presente projeto de pesquisa foram concentrados os esforços sobre padrões demonstrando sua aplicação no design de

artefatos digitais através do Web Design. Para informações ricas, e reconhecidas, sobre a atuação dos padrões no Game Design recomenda-se a leitura de Björk e Holopainen (2005).

4.2.1 Padrão Wizard

O presente estudo de caso parte do pressuposto que os padrões visam solucionar problemas de projeto corriqueiros. Foi identificado, portanto um problema amplamente encontrado nos websites, atingindo inclusive os grandes portais como “*Yahoo!*”: a confusão na solicitação de informação nos formulários de cadastro. O presente procedimento objetiva reusar uma solução descrita no padrão *Wizard* para solucionar esse problema, propondo o redesign de formulários de cadastro de email do “*Yahoo!*”. Na apuração de informações que ilustrem a redução do problema proposto através da aplicação do padrão, foram coletados dados de 40 entrevistas realizadas com a opinião de usuários.

Desse modo, solucionando um problema através do reuso de um padrão, pretende-se demonstrar de modo mais claro como os padrões oferecem reuso de soluções e transferência de conhecimento, em conformidade com o ilustrado na Figura 4.

Introduzindo o campo de atuação específica, convém esclarecer que os formulários são usados na Web os mais diversos fins. Entretanto, seu propósito base é coletar uma série de dados fornecidos pelo usuário (através da digitação em campos nomeados) para então envio dos mesmos. Esta operação pode servir para enviar uma mensagem de contato, cadastrar-se num serviço, acessar sua conta de banco ou mesmo registrar um novo email.

Descrevendo melhor o problema em mãos, foi-se verificado que os formulários em geral não informam os campos que são de preenchimento obrigatório. Quando o usuário submete o formulário ao envio, usualmente recebe uma mensagem de erro por não ter preenchido algum campo que era obrigatório, mas que não fora previamente avisado. Agravando este problema, alguns formulários apresentam-se demasiadamente longos, solicitando muitas informações de caráter pessoal, as quais o usuário não faz idéia, às vezes, do por que estarem solicitando estas informações. Esta desconfiança faz com que ele deixe alguns campos em branco e, então, recebe mais mensagens de erro durante a operação de envio dos dados, pois alguns dos campos que ele deixara em branco, por desconfiança, eram obrigatórios (mas isso não fora previamente avisado). Como propor uma solução para este problema?

O presente procedimento consiste, inicialmente, em identificar numa Linguagem de Padrões (repertório com vários padrões) uma solução para este problema. Foi encontrado no “*Designing Interfaces*”, de Jennifer Tidwell (2005), o padrão *Wizard*. Em poucas palavras, o que este padrão propõe é a instalação de um quadrinho ao lado de cada campo do formulário com uma breve explanação do mesmo (nisto descreveria se o campo é obrigatório e a que se destina aquele dado).

Reaproveitando toda a experiência de design descrita no padrão *Wizard*, neste experimento foi reprojeto o formulário do “*Yahoo!*” (Especificamente o de cadastro de email). Um total de 40 alunos foram entrevistados, no curso superior de Web Design da Faculdade Marista do Recife, sendo-lhes apresentados os formulários antes e depois do redesign pelo uso do padrão, constando abaixo dos formulários os seguintes questionamentos:

- **Pergunta 1** – Você consegue identificar os campos obrigatórios?
- **Pergunta 2** – Você tem plena consciência sobre o destino, e para qual propósito os dados informados serão utilizados?

Na seção seguinte encontra-se disponível os resultados obtidos pela execução deste procedimento bem como discussões relativas aos mesmos.

4.2.1.1 Resultados e Breves Considerações

Conforme descrito, o procedimento foi iniciado apresentando os formulários originais do “*Yahoo!*”, coletando as informações relativas à opinião dos usuários, direcionando-lhes pelas duas perguntas chave outrora citadas. Na Figura 37 é possível apreciar o formulário original.

Yahoo! - Ajuda

YAHOO! BRASIL

Olá!
 Você irá configurar sua conta do Yahoo! em três etapas simples! Basta responder a algumas perguntas simples, selecionar uma ID e senha e está tudo pronto.

Já possui uma ID ou endereço de e-mail?

[Esqueceu sua senha ou ID Yahoo!?](#)

Eu prefiro conteúdo do

1. Fale-nos um pouco de você...

Nome
 Sexo
 Data de nascimento Dia - Selecionar mês - Ano
 País
 CEP

2. Selecione uma ID e senha

ID Yahoo! e e-mail @
 Senha
 Redigite sua senha

3. Caso você esqueça sua ID ou senha...

E-mail alternativo
 Pergunta de segurança
 Sua resposta

Apenas mais alguns detalhes...

Preferências de marketing Quero receber, ocasionalmente, informações sobre promoções e ofertas especiais do Yahoo!.

Digite o código exibido



Você concorda? Li e concordo com os [Termos do Serviço do Yahoo!](#) e a [Política de Privacidade do Yahoo!](#), e em receber importantes comunicados do Yahoo! eletronicamente.
 Li e aceito os [Termos do Serviço do Mail](#).
 Para sua conveniência, estes documentos serão enviados via e-mail para a sua conta do Yahoo! Mail.

Figura 37 - Formulário original do "Yahoo!"

Na Figura 38 disponibilizei o formulário modificado, com a ilustração do quadrinho proposto pelo padrão *Wizard* ao lado dos campos.

YAHOO! BRASIL Yahoo! - Ajuda

Olá!
 Você irá configurar sua conta do Yahoo! em três etapas simples! Basta responder a algumas perguntas simples, selecionar uma ID e senha e está tudo pronto.

Já possui uma ID ou endereço de e-mail?
 [Esqueceu sua senha ou ID Yahoo!?](#)

Eu prefiro conteúdo do Yahoo! Brasil

1. Fale-nos um pouco de você...

Nome
 Sexo - Selecionar um(a) -
 Data de nascimento - Selecionar mês -
 País Brasil
 CEP

Essa 1ª etapa não é obrigatória e servirá para mantermos conhecimento do usuário, segmentando-o de acordo com suas características

2. Selecione uma ID e senha

ID Yahoo! e e-mail @ yahoo.com.br
 Senha Força da senha
 Redigite sua senha

Essa 2ª etapa é obrigatória e servirá para que sua conta de e-mail seja criada

3. Caso você esqueça sua ID ou senha...

E-mail alternativo
 Pergunta de segurança - Selecionar um(a) -
 Sua resposta

Essa 3ª é obrigatório e servirá para enviarmos a confirmação da criação da sua conta, e caso perca a senha solicitaremos sua pergunta e resposta de segurança

Apenas mais alguns detalhes...

Preferências de marketing Quero receber, ocasionalmente, informações sobre promoções e ofertas especiais do Yahoo!.

Digite o código exibido

Essa 4ª e última etapa é obrigatória e servirá para codificarmos seus dados impedindo possíveis tentativas de fraude

Você concorda? Li e concordo com os [Termos do Serviço do Yahoo!](#) e a [Política de Privacidade do Yahoo!](#), e em receber importantes comunicados do Yahoo! eletronicamente.
 Li e aceito os [Termos do Serviço do Mail](#).
 Para sua conveniência, estes documentos serão enviados via e-mail para a sua conta do Yahoo! Mail.

Figura 38 - Redesign do formulário do "Yahoo!"

Nas Figuras 39 e 40 abaixo estão disponibilizadas, percentualmente, as respostas dos usuários para as perguntas feitas mediante a apresentação dos formulários em seqüência. Primeiro os originais, depois os modificados (Em cada apresentação as respostas eram coletadas).

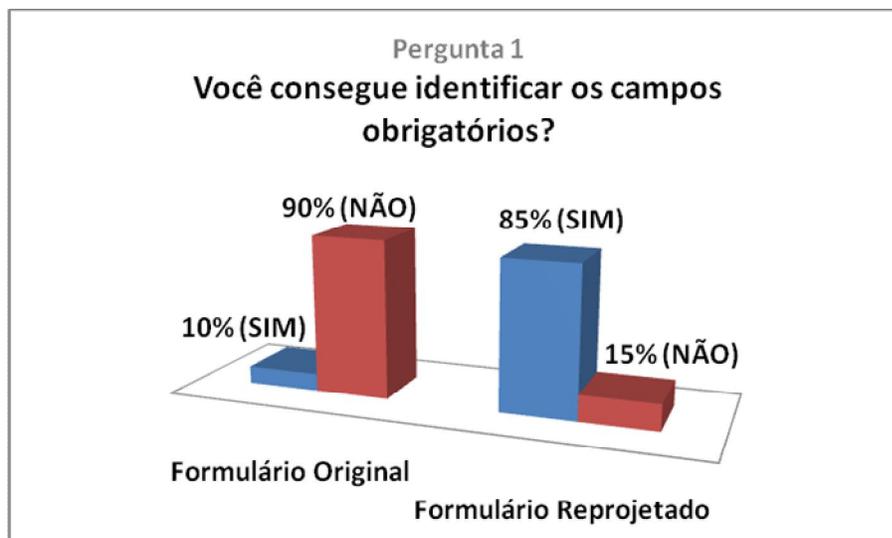


Figura 39 - Percentual de Respostas para Pergunta 1 - Formulário do “Yahoo!”

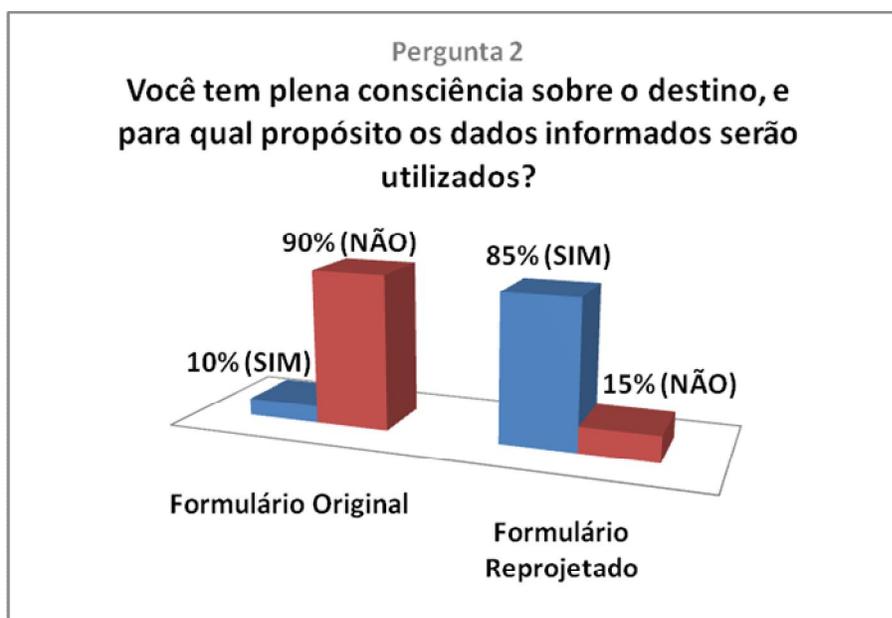


Figura 40 - Percentual de Respostas para Pergunta 2 - Formulário do “Yahoo!”

Os números apresentados como resultados da investigação mostram-se substancialmente favoráveis aos benefícios da aplicação do padrão *Wizard* na solução que tange a redução da confusão nos formulários de grandes portais da Web. Existe um expressivo contraste nas respostas dos usuários diante do formulário original e do redesign do mesmo. Enquanto que apenas 10% dos entrevistados conseguiam identificar quais eram os campos obrigatórios nos formulários originais, foi-se obtido um resultado de 90% no formulário reprojeto através do reaproveitamento da experiência de projeto do padrão *Wizard* (Vide Figura 39). Quanto à desconfiança relativa ao destino das informações

concedidas no formulário, pode-se observar que houve uma redução igualmente significativa. De 90% dos entrevistados que apresentavam esta desconfiança no formulário original, pode-se observar uma redução para o valor de 15% no formulário reprojetoado (Vide Figura 40).

Diante destes resultados, somos induzidos a crer que haveria uma relativa redução da confusão nos formulários dos grandes portais analisados pela efetiva redução dos problemas discutidos ao longo deste estudo de caso. Tudo isto vem reafirmar a importância do paradigma de projeto baseado em padrões no Web Design e, por consequência, no design de artefatos digitais.

A experiência de projeto descrita através do padrão *Wizard* num contexto específico foi reutilizada no contexto atual, que lida com formulários de grandes portais. O conhecimento disponível no padrão (sob a descrição de uma experiência de projeto aplicada noutro contexto) foi transferido, permitindo que, em posse deste conhecimento, fosse possível o reuso da solução no contexto atual.

Espera-se, com este estudo de caso, seja possível ilustrar como os padrões oferecem reuso de soluções e transferência de conhecimento dentro da atividade projetual. Além de contribuir com o setor oferecendo uma proposta de redesign de portais massivamente acessados na Web.

4.2.2 Padrão Breadcrumbs (Migalhas de Pão)

Para este estudo de caso foi identificado, novamente, um problema corriqueiro na Web e buscada uma solução nos padrões. As falhas de navegabilidade são motivo do fracasso de muitos portais da Web (KRUG, 2001). Abordando o problema em específico, para este experimento foi escolhido website da UFPE, sendo proposta uma solução para falhas de navegabilidade do mesmo através da reutilização da experiência de projeto descrita no padrão *Breadcrumbs*. Diante de um procedimento direcionado, foi-se verificado que 50 usuários – em sua maioria antes sem consciência de sua localização dentro do referido website – passaram responder corretamente sua posição dentro do portal da UFPE. Isto após o redesign do referido portal realizado neste experimento, com o auxílio do padrão *Breadcrumbs*.

Delineando o problema solucionado (neste estudo de caso) pelo reuso do referido padrão, encontraremos na Web extensas bases informacionais, acessadas através de

algoritmos de navegabilidade compostos por *hyperlinks*. Através dos mesmos, os usuários podem navegar por dentro de um website ou de um website para outro. Neste caso será considerada a primeira circunstância. É basicamente isto que milhões de usuários fazem simultaneamente em todo o mundo, navegar dentro de websites.

As interfaces de navegação precisam atender a três questionamentos fundamentais, por parte dos usuários: “Onde estou?”, “Onde estive?” e “Onde posso ir?”. As respostas a estas perguntas são de primária importância para o sucesso, ou fracasso, de um website. As pessoas não usarão um website no qual não conseguem navegar.

Foi-se identificado, portanto, que no website da UFPE (assim como em outros portais) é comum o usuário perder-se durante a navegação. Considerando ser este um problema de projetual corriqueiro na Web, foi-se verificado no “*Yahoo! Design Patterns Library*”, dentro da categoria “navegação”, alguma experiência de projeto válida para o contexto atual. Detectou-se que o padrão *Breadcrumbs* era usado quando “o usuário não consegue navegar facilmente por dentro as hierarquias de páginas”. Este padrão propõe que seja criada uma espécie de rastro durante a navegação, visível num local estrategicamente disposto no *layout* da página (por razões cognitivas) permitindo que o usuário veja exatamente por onde ele andou e exatamente onde ele está, dentro da hierarquia de páginas.

O procedimento consistiu em fazer um redesign de uma página específica do website da UFPE, considerando toda a experiência de projeto descrita no padrão *Breadcrumbs* (Ver APÊNDICE A). A página escolhida foi a do “Gabinete do Reitor”, apresentada na Figura 41. Em seguida 50 alunos dentre diversos cursos da Faculdade Marista do Recife (Web Design, direito, publicidade, administração e marketing) foram entrevistados. O intuito era de averiguar se os mesmos sabiam exatamente em que página eles estavam, apresentando-lhes primeiramente o website da UFPE original (Figura 41) e, logo em seguida, website reprojeto pelo reuso do padrão *Breadcrumbs* (Figura 42). Cada apresentação consistia na página do website impressa em papel efetuando, logo abaixo, questionamentos perguntando se o usuário sabia onde estava e, logo abaixo, onde estava. Seguindo as orientações contidas no padrão, podemos observar as “migalhas de pão” (rastro de navegação) no canto superior esquerdo (Figura 42).



Figura 41 - Página do website original da UFPE



Figura 42 - Página reprojeta da UFPE

Os resultados obtidos com a realização deste procedimento, bem como as discussões pertinentes, encontram-se na seção a seguir.

4.2.2.1 Resultados e Breves Considerações

Conforme podemos observar nas Figuras 43 e 44, diante da página original 80% dos usuários afirmavam saber em que página estavam, entretanto somente 7,5% responderam corretamente sua posição. Em outras palavras, 92,5% dos usuários estavam perdidos.

Reutilizando a experiência de projeto do padrão *Breadcrumbs*, foi-se gerado o redesign da mesma página do portal da UFPE. Neste caso 88% dos usuários afirmavam saber exatamente em que página estavam, mas foi possível elevar para 75% o número de respostas corretas quanto a sua localização no website (Figura 45).

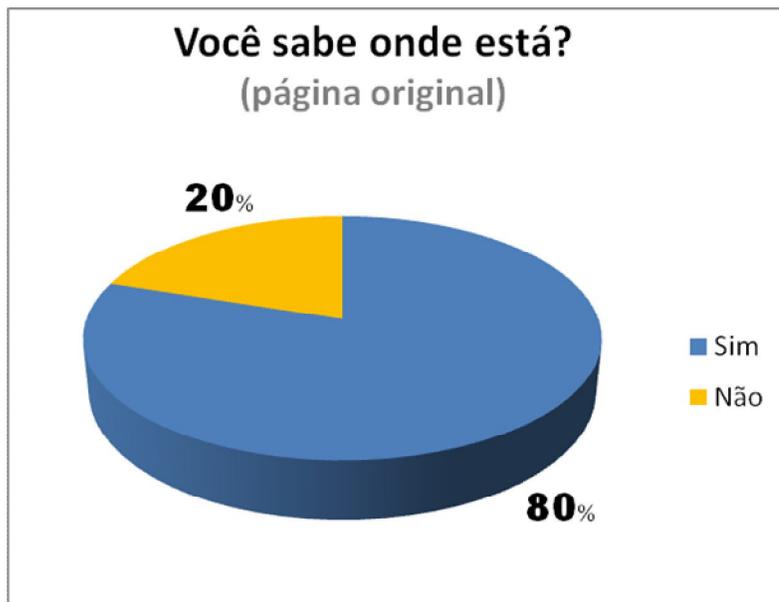


Figura 43 - Resultados mediante apresentação da página original da UFPE



Figura 44 - Resultados mediante apresentação da página da UFPE reprojeta

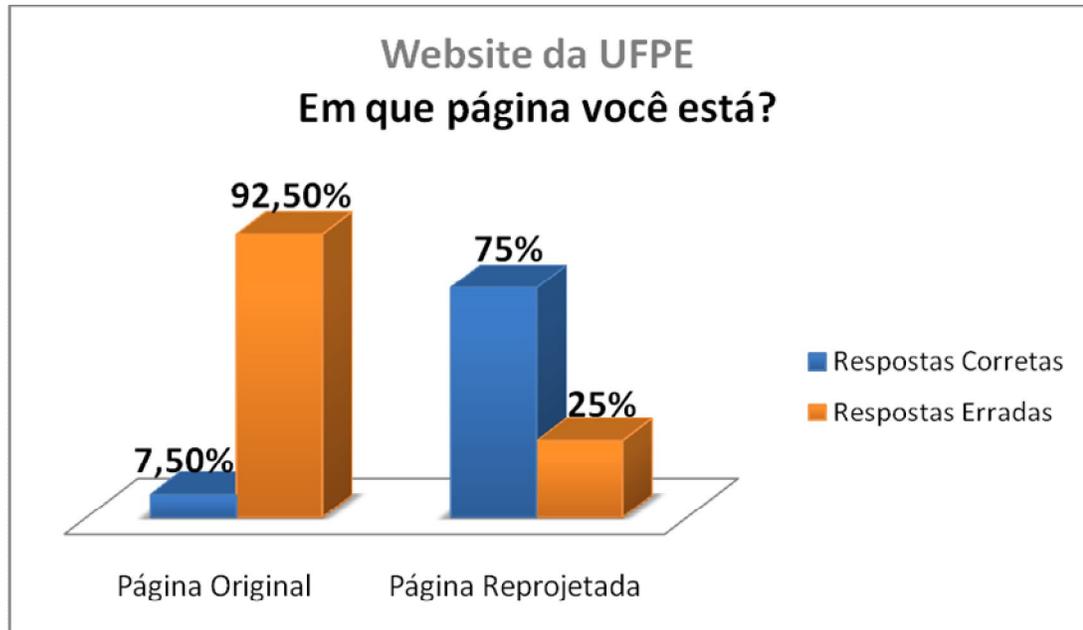


Figura 45 - Resultados do redesign do website da UFPE

Estes números nos levantam alguns questionamentos. Por exemplo, por que não obtive 100% de respostas corretas quanto a posição do usuário na página reprojeta, uma vez que as “migalhas de pão” mostravam-lhes sua exata posição? Uma das hipóteses que podemos levantar é que as “migalhas de pão” não estavam nítidas o suficiente; ou indevidamente posicionadas.

De qualquer modo, observamos, mais uma vez, expressivos resultados na solução de um problema corriqueiro através do paradigma de projeto baseado em padrões. O conhecimento da experiência de design descrita no padrão *Breadcrumbs* foi absorvido e, após isto, a solução foi reaplicada no presente contexto.

Desta maneira espera-se ter ilustrado mais uma vez, com sucesso, como os padrões oferecem reuso de soluções e transferência de conhecimento. Além de disponibilizar uma proposta de solução para um problema de navegabilidade ao website da UFPE.

4.3 Considerações Gerais

4.3.1 Discussões sobre Padrões

Segundo o que foi explicitamente apresentado em nossos estudos de caso, o esquema proposto na Figura 4, sobre como os padrões oferecem reuso de soluções e transferência

de conhecimento, demonstra-se plausível. Tanto do estudo de caso do padrão *Wizard* quanto no do *Breadcrumbs*, aprendeu-se com a experiência de projeto aplicada em outro contexto, reutilizando a solução para o meu contexto em mãos. Assim sendo, é possível afirmar que a transferência de conhecimento se dá pelo aprendizado, através de um processo comunicativo, e o reuso de soluções pelo manuseio deste conhecimento adquirido, usando-o no contexto atual.

Quanto à afirmação de Adikson (2005), que destaca a dificuldade em achar uma solução para o problema em mãos pesquisando (lendo) bancos de padrões (Linguagens de Padrões), pelo menos nos presentes estudos de caso isto não ocorreu. Ambas as soluções foram localizadas em questão de minutos. Talvez por tratarem-se de problemas extremamente corriqueiros, ao ponto de serem detectados em portais de elevada audiência na Web.

Convém levantar ainda uma discussão tão intrínseca aos padrões quanto a sua própria composição. Este paradigma de projeto corriqueiramente levanta questionamentos quanto a uma suposta comodidade formal. Ou seja, soluções prontas que limitariam a pesquisa por novas soluções. Entretanto, a proposta primária deste paradigma projetual, quando concebido por Alexander (1979) era de manter a sua assinatura, suas características e sua qualidade (inerente à sua experiência) em seus projetos, mesmo que fossem desenvolvidos por terceiros. Trata-se não só de um mecanismo poderoso para transferência de conhecimento mas também para manter qualidade de produtos gerados dentro de uma mesma indústria que amplia seu quadro de projetistas ou perde projetistas experientes. Sendo assim, se os padrões levam a uma comodidade formal por parte dos designers, não cabe a esta pesquisa por um ponto final neste questionamento, mas esclarecer que apenas que não era esta a proposta dos padrões em seu advento nem continua sendo ainda hoje.

De qualquer forma, uma coisa é certa, o design baseado em padrões é um recurso poderoso para reduzir tempo e riscos de projeto. Sendo esta a necessidade que o projeto em mãos demanda, os padrões irão suprir satisfatoriamente, conforme ilustrado através dos estudos de caso e ao longo dos exemplos citados no estado da arte.

4.3.2 Discussões sobre Plataformas

De acordo com Goering(2002) e Sangiovanni-Vincentelli(2002), o paradigma de projeto baseado em plataforma está em consolidação e, portanto, sendo conceituado. Ao identificar, conceituar, expor e experimentar características comuns em todas as conceituações de

plataforma encontradas, espera-se ter contribuído com o setor no que tange o esclarecimento do que vem a ser uma plataforma e os aspectos que a englobam. A saber, componentes fixos, componentes variáveis, capacidade de abstração e reuso.

Possuindo apenas 1/3 das publicações do estado da arte desta pesquisa, contrabalanceando a grande quantidade de publicações sobre padrões, as plataformas apresentam-se timidamente, embora com uma proposta extremamente atraente de acelerar o desenvolvimento dos projetos e reduzir riscos. Talvez porque os padrões sejam, em si, publicações enquanto que as plataformas representam projetos prontos (no caso presente, softwares).

O design de software utiliza-se do conceito das plataformas principalmente na Programação Orientada a Objetos, entretanto o termo “plataforma” demonstra-se pouco conhecida no setor. Instiga-se a possibilidade disto se dar por duas prováveis justificativas, abaixo pontualmente destacadas.

- Desentendimentos sobre o conceito de plataformas (GOERING, 2002). Sendo muitas vezes associadas a programas em uso, sistemas operacionais ou mesmo tipo de hardware utilizado. Por exemplo, é comum referir-se a um dado programa rodando numa dada “plataforma”, relacionando-a ao hardware que suporta este programa e não associando-a com um paradigma de projeto no qual foi concebido este hardware;
- O conceito de plataforma já é intrínseco na Programação Orientada a Objetos. Fazendo com que as pessoas prefiram referir-se a termos já usados na área do que adotar uma terminologia que está sendo forjada e discutida. Ainda mais sem haver necessidade motivadora para isso. Todavia, a terminologia plataforma estende-se, com a mesma conceituação básica de componentes fixos, componentes variáveis e abstração, em outras aplicações fora do contexto de software.

4.3.3 Proposta de Integração entre Padrões e Plataformas

Antes de iniciar estas considerações, se faz necessário esclarecer que tratam-se de paradigmas projetuais completamente distintos. É possível a aplicação de ambos ao mesmo projeto, ou apenas um deles e, de qualquer forma, não interferirão em seus campos de atuação. Porém, uma vez que ambos propõem-se a solucionar os mesmos problemas

projetuais, uma proposta de aplicação integrada seria interessante, pela possibilidade de ampliar seu poder em cumprir seus objetivos.

Sendo assim, uma metodologia de projeto baseada em padrões e plataformas atuando de forma síncrona poderia, perfeitamente, possuir uma Linguagem de Padrões onde cada padrão, além de seus componentes tradicionais como nome, problema, contexto, padrões relacionados, ostentar ainda um componente chamado “plataformas relacionadas”. Uma conexão dos padrões propondo plataformas que possuam assuntos relacionados com o problema em questão. Isto poderia ainda auxiliar na busca por plataformas, uma vez que não existem “catálogos” das mesmas, como encontraremos nos padrões.

5 CONCLUSÕES

Diante das informações levantadas e discutidas ao longo deste documento é possível concluir que os padrões e plataformas contribuem efetivamente ao design de artefatos digitais provendo reuso de soluções e transferência de conhecimento. Como fruto dos estudos de caso, temos a clara ilustração do funcionamento destes paradigmas de projeto: as plataformas colaboram neste sentido devido a sua efetiva capacidade de abstração, enquanto que os padrões através de um procedimento didático baseado na retórica grega.

Foi confirmado, através dos estudos de caso, os demais benefícios agregados pelo uso destes paradigmas projetuais no design de artefatos digitais. Dentre eles, destaca-se a drástica redução de tempo de projeto, ilustrada tanto nos projetos dos “MOD”s derivados da plataforma “*Source*”, quanto nos redesigns dos portais do “*Yahoo!*” e da UFPE. É factível que a redução de custos de projeto pelo emprego destes paradigmas projetuais se dá não somente pela redução de tempo, mas também dos esforços envolvidos (quantidade de profissionais). Chega-se a esta conclusão devido ao contraste de esforço necessário para construção de uma plataforma como a “*Source*” (que custou cerca de \$40 milhões de dólares), com os “MOD”s apresentados nos estudos de caso, desenvolvidos por equipes de 1 a 2 pessoas e em poucos dias. Acrescentando-se a esta demonstração, é possível observar os redesigns dos portais do “*Yahoo!*” e da UFPE, que demandariam arquitetos da informação, designers gráficos e demais profissionais do setor de Web Design para reprojeter a interface, o que foi conseguido apenas reaproveitando os padrões disponíveis.

Foi demonstrado que os paradigmas de projeto baseados em padrões e plataformas agregam ainda outros benefícios, incluindo aumento da qualidade do produto final (mesmo sendo produzido por um designer novato) e redução de riscos do projeto (pois é fruto da derivação de um projeto empírico).

No que tange as contribuições desta pesquisa, é cabível dar destaque ao modelo de plataforma de games proposta. A taxonomia realizada no Game Design descreve de modo claro e objetivo os resultados do emprego deste modelo de plataforma, composta por engine como componente fixo, além de objetos, cenário, Interface Gráfica e sons como componentes variáveis. Vindo a contribuir, com esta taxonomia e modelo de plataforma, para a comunidade de Game Design em congressos que englobam o setor.

Em suma, o presente projeto levantou a hipótese que os padrões e plataformas contribuiriam ostensivamente no reuso de soluções e transferência de conhecimento dentro do design de artefatos digitais. Abordou-se este campo de estudo através do Game Design e Web Design, nos quais foram gerados games e websites concebidos sobre estes paradigmas. Com os resultados destes estudos de caso averiguou-se que a hipótese levantada é verdadeira. Por fim, espera-se ter conseguido contribuir com a comunidade científica através da geração de taxonomias, artigos e softwares englobando o uso de padrões e plataformas, permitindo assim ilustrar detalhadamente seu funcionamento e benefícios.

5.1 Limitações e Desdobramentos

É cabível propor ainda o desenvolvimento de uma metodologia de design que englobasse os padrões e plataformas dentre fases de projeto ou ainda que integrasse estes paradigmas. O que não foi possível desenvolver no presente projeto de pesquisa devido as limitações temporais do mesmo.

Seria interessante ainda uma pesquisa de longo prazo verificando o desempenho de equipes de design que atuam com e sem os paradigmas projetuais em questão. Permitindo assim a geração de dados mais depurados a respeito das contribuições dos padrões e plataformas no que tange tempo de desenvolvimento e qualidade do projeto final.

Outro desdobramento proposto seria a criação de um sistema baseado em tecnologias da Web 2.0 capaz de buscar padrões em múltiplos repositórios. Reduzindo assim o tempo de busca, principal limitação deste paradigma de projeto.

No que tange o design baseado em plataformas aplicado ao Game Design, muito ainda tem por ser feito. A utilização de outras plataformas, além da “Source” pode nos revelar novos benefícios.

Atualmente muitos grupos de Game Design possuem a limitação de não possuir engenheiros de software ou cientistas da computação em seu efetivo. Restringindo-se à análise de games e não à produção dos mesmos. Um desdobramento da presente pesquisa observando como a extensiva abstração provida pelas plataformas de games poderiam contribuir para estes grupos de Game Design poderiam render informações que alavancariam a produção de games através do reuso de plataformas.

REFERÊNCIAS

ADKISSON, H. *How Useful are User Interface Patterns? Our Experience*. Disponível em: <http://www.blinkinteractive.com/ourexperience/essays/2007/11/interface_patterns.php>. Acesso em: 20 ago 2008.

ALEXANDER, C. *Notes on the Synthesis of Form*. Cambridge: Harvard University Press, 1964.

ALEXANDER, C. *The Timeless Way of Building*. Nova Iorque: Oxford University Press, 1979.

ALEXANDER, C. *A Pattern Language*. Nova Iorque: Oxford University Press, 1977.

APPLETON, B. *Introducing Patterns! Patterns and Software: Essential Concepts and Terminology*. Disponível em: <<http://www.cmcrossroads.com/bradapp/docs/patterns-intro.html>>. Acessado em: 12 set 1997.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR14724*: informação e documentação – trabalhos acadêmicos - apresentação. Rio de Janeiro, 2001.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR6023*: informação e documentação - referências - elaboração. Rio de Janeiro, 2000.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR12225*: títulos de lombada.. Rio de Janeiro, 1992.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR6027*: sumário. Rio de Janeiro, 1989.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR6024*: numeração progressiva das seções de um documento. Rio de Janeiro, 1989.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR6028*: resumos. Rio de Janeiro, 1990.

BECK, K. & CUNNINGHAM, W. *Using Pattern Languages for Object-Oriented Programs*, 1987. Technical Report n° CR-87-43. Disponível em: <<http://c2.com/doc/oopsla87.html>>. Acessado em: 30 out. 2008.

BERKELEY. *How Much Information 2003?* University of California. Disponível em: <<http://www.sims.berkeley.edu/research/projects/how-much-info-2003/>>. Acessado em: 20 ago 2005.

BETHKE, E. *Game Development and Production*. Plano: Wordware Publishing, 2003.

- BJÖRK, S. & HOLOPAINEN, J. *Patterns in Game Design*. Rockland: Charles River Media, 2005.
- BOLCHINI, D. *Web Design Patterns: improving quality and performance in Web Application design*. Dissertação (Mestrado). Universit della Svizzera Italiana – USI, Communication Sciences, 2000.
- BOULDIN, D. Platform-based System-on-Chip Design. In: Proceedings of 2003 NASA Symposium on VLSI Design, Cour d’Alene, 28 a 29 maio, 2003. *11th NASA Symposium on VLSI Design*. Cour d’Alene: ID, 2003, pp. 1-4.
- BÜRDEK, B. *História teoria e prática do design de produtos*. São Paulo: Blücher, 2006.
- BUSCHMANN, F., MEUNIER, R. *A System of Patterns*. Nova Iorque: ACM Press, Addison-Wesley Publishing Co., 1995.
- CARDOSO, M., SILVA, L., GALLINA, M., KISTMANN, V. O Design Automotivo e a Modularização. In: P&D Design 2006 | 7º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design, 2006, Curitiba. *Anais do 7º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design*. Curitiba: AEnD-BR, 2006.
- CHEN, R., SGROI, M., LAVAGNO, L., MARTIN, G., SANGIOVANNI-VINCENTELLI, A., RABAEY, J. UML and platform-based design. UML for real: design of embedded real-time systems. Norwell: Kluwer Academic Publishers, 2003.
- CLAYPOOL, K. E CLAYPOOL, M. Software engineering design: teaching software engineering through game design. In: ITiCSE '05., Caprica, jun. 2005. *Proceedings of 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*. Nova Iorque: ACM Press, 2005.
- COPLIEN, J. *Advanced C++ Programming Styles and Idioms*. Reading: Addison-Wesley, 1992.
- COPLIEN, J. & SCHMIDT, D. *Pattern Languages of Program Design*. Reading: Addison-Wesley, 1995.
- DE MARINIS, M., FANUCCI, L., GIAMBASTIANI, A., RENIERI, A., ROCCHI, A., ROSADINI, C., SICILIA, C., SICILIA, D. Sensor Platform Design for Automotive Applications. In: DSD 2003, 1 a 6 de set. 2003. *Euromicro Symposium on Digital Systems Design*. Washington: IEEE Computer Society, 2003.
- EBERLY, D. *Game Physics*. San Francisco: Morgan Kaufmann, 2003.
- EBERLY, D. *3D Game Engine Design*. San Francisco: Morgan Kauffman, 2001.
- ERICKSON, T. Interaction Pattern Languages: A Lingua Franca for Interaction Design? In: *UPA 98 Conference*. Washington: 1998.
- FIGUEIRÔA, D., VILAR NETO, E., NUNES, I., CORREIA, A., ALVES, H., CAMPOS, F., NEVES, A. Técnicas Matemáticas aplicadas ao Processo de Design. In: P&D Design 2008 |

8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design, 2008, São Paulo. *Anais do 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design*. São Paulo: SENAC, 2008.

FIGUEIRÔA, D., CALAZANS, D., CAMPOS, F., CAMPELLO, S., NEVES, A. Conceituando Web Design através de Categorização. In: P&D Design 2008 | 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design, 2008, São Paulo. *Anais do 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design*. São Paulo: SENAC, 2008.

FIGUEIRÔA, D., CAMPOS, F., NEVES, A. O Paradigma do Projeto baseado em Plataforma aplicado ao Web Design. In: P&D Design 2008 | 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design, 2008, São Paulo. *Anais do 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design*. São Paulo: SENAC, 2008.

FIGUEIRÔA, D., CAMPOS, F., CHAVES, W., NEVES, A. Game Design Baseado em Plataforma. In: P&D Design 2008 | 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design, 2008, São Paulo. *Anais do 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design*. São Paulo: SENAC, 2008.

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.

FLÜSSER, V. O Mundo Codificado. São Paulo: Cosacnaify, 2007.

FREEMAN, E., FREEMAN, E., SIERRA, K., BATES, B. Head First Design Patterns. Sebastopol: O'Reilly Media, Inc., 2004

FRIZELL, S. *A Pattern-Based Design Methodology for Web-based Instruction*. Thesis Research. Auburn: Auburn University, 2001.

FULLERTON, T., SWAIN, C., HOFFMAN, S. *Game Design Workshop: Designing, Prototyping, and Playtesting Games*. Lawrence: CMP Books, 2004

GAMMA, E.; HELM, R.; JOHNSON, R.; VLISSIDES, J. Padrões de Projeto. Porto Alegre: Editora Bookman, 2000.

GOERING, R. Platform-based Design: A Choice, Not a Panacea. Washington: *EETimes*, 2002.

GRAAF, B., LORMANS, M., TOETENEL, H. Embedded software engineering: the state of the practice. *IEEE Software*. Los Alamitos: IEEE Computer Society Press, 2003.

GRAHAM, I. A Pattern Language for Web Usability. Londres: Addison Wesley, 2003.

GRANLUND, A., LAFRENIÉRE, D., CARR, D. A Pattern-supported Approach to the User Interface Design Process. *Proceedings of HCI International 2001*. New Orleans, 5 a 10 ago.

2001. *9th International Conference on Human-Computer Interaction*. New Orleans: Lawrence Erlbaum Associates, 2001. vol. 1, p. 282-286.
- GRIFFITHS, R., PEMBERTON, L. Patterns in Human-Computer Interaction Design. In: IHM-HCI 2001 Conference, Lille, 10 a 14 set. 2001. *Proceedings of IHM-HCI 2001 Conference*. Lille: France, 2001.
- ISAKOWITZ, T.; STOHR E.; BALASUBRAMANIAN P. RMM: A Methodology for Structured Hypermedia Design. *Communications of ACM*, Ago. 1995, v. 38, n. 8.
- KABASHI, A. *Game Software Processes*. Dissertação (Mestrado). Ronneby: Software Engineering, School of Engineering, Blekinge Institute of Technology, 2006.
- KRUG, S. Não me faça pensar. São Paulo: Editora Market Books, 2001.
- MACDONALD, N. What is web design?. Hove: Rotovision, 2003.
- MELO, A. *Padrões de Projeto (Design Patterns) em PHP da Teoria à Prática*. PHP Magazine, Jan. 2007. Ed. 01, 7.
- MELO, A., NEVES, A. Fábrica de Websites. In: P&D Design 2006 | 7º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design, 2006, Curitiba. *Anais do 7º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design*. Curitiba: AEnD-BR, 2006.
- NIEBORG, D. *Am I Mod or Not? – An analysis of First Person Shooter modification culture*. Creative Gamers Seminar - Exploring Participatory Culture in Gaming. University of Tampere. Disponível em: <<http://gamespace.nl/research>>. Acessado em: 20 nov 2005.
- NIELSEN, J. & TAHIR, M. Homepage: Usabilidade. Rio de Janeiro: Campus, 2002.
- NSAME, P., LINZER, H., KAUTZMAN, M., LAVERE, J., KLING, G., MCGRODDY, K. E GUERRIERO, T. IBM Platform-Based Design Methodology Enablement. *Micronews*, Essex Junction, Nov. 2001. Essex Junction: IBM Microelectronics, 2001, vol. 7, n. 4, p. 14.
- PONT, M. Patterns for Time-Triggered Embedded Systems: Building Reliable Applications with the 8051 Family of Microcontrollers. London.: Addison-Wesley, 2001.
- ROUSE III, R. Game Design Theory and Practice. Plano: Wordware Publishing, 2001.
- SALEN, K. E ZIMMERMAN, E. Rules of Play – Game Design Fundamentals. Cambridge: MIT Press, 2004.
- SANGIOVANNI-VINCENTELLI, A. *Defining Platform-based Design*. EEDesign. Disponível em: <<http://www.eetimes.com/story/OEG20020204S0062>>. Acessado em: 10 abr 2007.
- SHAW, M. Patterns for Software Architectures. *Coplien & Schmidt, Pattern Languages of Program Design*. Reading: Addison-Wesley, 1995. p. 453-462.

TALARICO NETO, A. *Linguagem de Padrões para Apoiar o Projeto de Material Instrucional para EAD*. Dissertação (Mestrado). São Carlos: Departamento de Computação. UFSCAR, 2005.

TIDWELL, J. *Designing Interfaces: Patterns for Effective Interaction Design*. Cambridge: O'Reilly Media, Inc., 2005

WOLF, W. What is a platform? *Platform-Based Design, Part1*. Disponível em: <<http://www.cs.princeton.edu/courses/archive/spr02/cs598u/wolf1.ppt>>. Acessado em: 25 dez 2006.

WEHRMEISTER, M., BECKER, L. B. ; PEREIRA, C. Metodologia de Projeto Orientada a Objetos Baseada em Plataformas para Sistemas Tempo-Real Embarcados. In: VII Workshop de Tempo Real, Fortaleza, 13 maio 2005. *Anais do VII Workshop de Tempo Real*. Porto Alegre: SBC, 2005, v. 1. p. 1-8.

VALVE. Steam Player Number Statistics. *Steam Network Status*. Disponível em: <http://www.steampowered.com/status/game_stats.html>. Acessado em: 20 nov 2007.

VLISSIDES, J. Patterns: The Top Ten Misconceptions. *Object Magazine*. Disponível em: <<http://www.research.ibm.com/designpatterns/pubs/top10misc.html>>. Acessado em: 21 dez 2007

ANEXO A - PRODUÇÃO ACADÊMICA E TÉCNICA

O findar desta pesquisa trouxe diversos resultados que oscilam de softwares experimentais gerados a artigos científicos. Apresenta-se neste anexo os principais resultados gerados pelo presente trabalho, esperando que os mesmos venham a servir de contribuições, principalmente ao design de artefatos digitais.

A presente produção científica, composta por diversos artigos descrevendo partes desta pesquisa, resultou num total de 5 documentos publicados e diversos outros artigos ainda a serem submetidos. Os desdobramentos são diversos, sendo inclusive muitas destas produções continuação das outras, o que proveu linearidade ao presente trabalho.

Abaixo disponibilizado, de modo sintetizado, os resultados obtidos (em publicações) por consequência deste trabalho.

Artigos Completos Publicados

FIGUEIRÔA, D., VILAR NETO, E., NUNES, I., CORREIA, A., ALVES, H., CAMPOS, F., NEVES, A. Técnicas Matemáticas aplicadas ao Processo de Design. In: P&D Design 2008 | 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design, 2008, São Paulo. **Anais do 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design**. São Paulo: SENAC, 2008.

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.

Artigos Resumidos Publicados

FIGUEIRÔA, D., CALAZANS, D., CAMPOS, F., CAMPELLO, S., NEVES, A. Conceituando Web Design através de Categorização. In: P&D Design 2008 | 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design, 2008, São Paulo. **Anais do 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design**. São Paulo: SENAC, 2008.

FIGUEIRÔA, D., CAMPOS, F., NEVES, A. O Paradigma do Projeto baseado em Plataforma aplicado ao Web Design. In: P&D Design 2008 | 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design, 2008, São Paulo. **Anais do 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design**. São Paulo: SENAC, 2008.

FIGUEIRÔA, D., CAMPOS, F., CHAVES, W., NEVES, A. Game Design Baseado em Plataforma. In: P&D Design 2008 | 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design, 2008, São Paulo. **Anais do 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design**. São Paulo: SENAC, 2008.

Softwares

- **“AutoProj”**: software concebido sob tecnologia PHP que auxilia o designer em decisões projetuais, modelando as massas de crença e extraindo a incerteza envolvida através de um mecanismo matemático denominado *Lateo*. Disponível online: <http://www.dino.eti.br/fabiocampos>.
- **“SBGames MOD”**: protótipo de game para PC, estilo FPS, que é o resultado de modelações experimentais na plataforma “*Source*”.
- **“P&D MOD”**: protótipo de game para PC que é um desdobramento do “*SBGames MOD*”, objetivando demonstrar a modificação de um novo componente variável encontrado na plataforma “*Source*”: os sons. Disponível online: <http://www.dino.eti.br/pedmod>.

APÊNDICE A – Padrão “Breadcrumbs”

Problem Summary

The user needs to be able to navigate up (towards the root page) and have an understanding of where she is in relation to the rest of the site.

EXAMPLE:

[Travel](#) > [Guides](#) > [North America](#) > [United States](#) > [New York](#) > [New York City](#) > Things to do

Things to do in New York City (547)

REFINE RESULTS BY

City
[Boston](#) (153)

SORT BY: [Popularity](#) | [Name](#) | [Distance](#)

Play ▶

Breadcrumb showing the Things To Do page for Boston, MA in Yahoo! Travel's travel guide

Use When

The page displayed is within a hierarchy of pages and is not the topmost page.

The user cannot easily navigate through the hierarchy via other local navigation methods. For example, if the page is fairly deep in a hierarchy, the breadcrumb maybe the simplest way to provide navigation.

The page may be arrived at from an external source (e.g., a search results page) and the user will need a sense of context.

Solution

Display a horizontal list of labels starting with the topmost page and continuing down the site's hierarchy to the current page.

Labels

Where possible, labels should match the title of their corresponding page.

Use the rules of title capitalization for labels in the breadcrumb.

Separate each label with a greater-than sign (>).

Include the title of the current page as the last label in the breadcrumb.

Do not use the label "Home" for the topmost page. Instead use the specific name for the topmost page (e.g. "Travel" or "Weather" rather than "Home".)

Hyperlinks

Hyperlink all labels in the breadcrumb except the last one (which corresponds to the title of the current page.)

Never hyperlink the greater-than sign (>) or the spaces that separate the labels.

The hyperlink should be styled the same regardless of whether it has been visited or not.

Other Considerations

Never display a breadcrumb on the site's topmost page.

The breadcrumb sometimes corresponds to the user's recent browsing history, but is not equivalent to it.

Dynamically include the titles of user-generated content pages as appropriate.

Rationale

Breadcrumbs provide context relative to the rest of the site.

Breadcrumbs provide a way to navigate up the site hierarchy.

The term 'Breadcrumb' can be misleading. It implies that this is the trail that got us here and will take us back the way we came. In reality, our Breadcrumbs pattern is more like Homeward Path as described in the Diemen Patterns Repository. However, we chose the name 'Breadcrumbs' since it is the most common name for this idiom.

Accessibility

Each breadcrumb label should match the corresponding page title.

Allow the breadcrumb and each link to be navigated to with the Tab key.

When an individual breadcrumb label has keyboard focus, the Enter key will navigate to the linked page.

Santos, Dino Lincoln Figueirôa
Design de artefatos digitais baseados em
padrões e plataformas / Dino Lincoln Figueirôa
Santos. – Recife: O Autor, 2009.
94 folhas. : il., fig., quadros.

Dissertação (mestrado) – Universidade Federal
de Pernambuco. CAC. Design, 2009.

Inclui bibliografia, anexo e apêndice.

1. Projeto de sistemas. 2. Desenho (Projetos). 3.
Sites da Web - Projetos. 4. Jogos por computador. I.
Título.

74 CDU (2.ed.)

740 CDD (22.ed.)

UFPE

CAC2009-75