Andréa Graciano, Sérgio Nesteriuck, Gilbertto Prado *

Considerações sobre "pattern"

*

Andréa Graciano é graduada em Engenharia Civil pela Unicamp e em Design Gráfico pela Universidade Anhembi Morumbi, pós graduada com MBA em Tecnologia da Informação na USP/Fia e com DESS em Gestão de Projetos pela Universidade de Grenoble (França) e, atualmente, é mestranda em design na Universidade Anhembi Morumbi.
<andrea.graciano@me.com>

Sérgio Nesteriuck é doutor em Comunicação e Semiótica pela PUC-SP (2007). Professor do Curso de Design de Games, Animação e do PPG Design da Universidade Anhembi Morumbi.
<nesteriuk@hotmail.com>

Gilbertto Prado é artista multimídia, estudou Artes e Engenharia na Unicamp e obteve o doutorado em Artes na Universidade Paris I – Panthéon Sorbonne em 1994. É professor titular aposentado da ECA/USP e atualmente Professor do PPG Design da Universidade Anhembi Morumbi.
<gttoprado@gmail.com>

Resumo O objetivo deste artigo é apresentar conceitos associados ao termo pattern e sua aplicação em diferentes áreas do conhecimento. Por meio da investigação bibliográfica, surgem duas principais abordagens para o assunto: o pattern como motivo visual utilizado amplamente no design de superfície e como solução encapsulada, proposta originalmente para a arquitetura mas que, posteriormente, se torna referência no desenvolvimento de sistemas no campo da ciência da computação. Este trabalho apresenta, também, exemplos de utilização prática do pattern na tipografia e no design de jogos em função de seu caráter modular e reutilizável.

Palavras chave Artes Visuais, *Pattern*, *Design* de superfície, *Design patterns*, Tipografia, Padronagem, *Design* de jogos.

Considerations about "Pattern"

Abstract *This paper's objective is to presente concepts of pattern and its application in different areas of knowledge. Carrying out a bibliographic research, two main approaches of this subject emerge: pattern as a visual module widely used in surface design, and pattern as a closed solution, which was originally proposed for architecture but turned into a model for system development in the field of computer science. Examples of actual uses of pattern in typography are also presented. Drawing attention to its modular and reusable feature.*

Keywords *Visual arts, Pattern, Surface design, Design patterns, Typography, Smealess, Game design.*

Introdução

A pesquisa que dá origem a este artigo surgiu de uma dificuldade encontrada durante a elaboração de um trabalho de mestrado, cujos objetos de estudo são imagens semelhantes às apresentadas na figura abaixo.

Figura 1 (da esquerda para a direita): Piso Squarish (Fonte:www.pophamdesign.com); Tecido Jessica Jones (Fonte:www.printandpattern.blogspot.com); e Adesivo de Parede Minerva Oppa (Fonte: www.oppa.com.br/adesivo-de-parede-minerva-azul).

O piso, o tecido e o papel de parede do exemplo têm em comum, além de sua aplicação no design de superfície, algumas características identificáveis na sua estrutura. Todas as composições visuais apresentadas são formadas por módulos geométricos, dispostos sobre um grid, seguindo um critério de repetição ou posicionados aleatoriamente. No caso específico do tecido e do papel de parede, teve-se também a preocupação com a articulação entre os desenhos de maneira a cobrir toda a superfície sem apresentar descontinuidades ou emendas.

Ao tentar se referir a esse tipo em particular de estampa utilizando o termo "pattern", sem apresentar aos leitores/interlocutores uma referência visual ou uma explicação mais detalhada, percebeu-se de imediato um problema na comunicação em virtude das várias interpretações a este termo associadas, conforme será visto adiante neste texto.

Em função disso, nasce a necessidade de entender, por meio de uma investigação bibliográfica, o que é pattern e qual seria a palavra mais adequada para representar as composições visuais mostradas acima.

O objetivo geral deste artigo é discutir sobre os conceitos relacionados ao termo pattern. Neste sentido, a primeira parte deste texto aborda a questão do "padrão", que corresponde à tradução da palavra "pattern" para o português. Na sequência, discute-se sobre o pattern como módulo visual e suas relações com o design de superfície. Em "Pattern como solução encapsulada" é apresentada a abordagem de Alexander (2012) e sua linguagem de padrões, passando pelos conceitos de modularização, caixa-preta, programação orientada a objetos, chegando à solução proposta por Gamma

et al. (2007): os padrões de projeto ou, no original em inglês, design patterns. Por fim, o caráter modular e reutilizável do pattern é exemplificado por uma aplicação prática em design gráfico, a tipografia e no design de jogos.

Padrão = default + standard + pattern

Como tradução de "pattern", do inglês para o português, utiliza-se comumente a palavra "padrão". Porém, por falta de opções melhores, "padrão" também recebe a tradução de duas outras palavras "default" e "standard", que embora guardem uma afinidade com "pattern", têm significados distintos.

Essa observação se torna importante na medida em que boa parte dos textos sobre o assunto são produzidos originalmente em inglês e na tradução para o português acabam perdendo, muitas vezes, parte essencial de seu sentido original, eventualmente, levando os leitores à confusões ou múltiplas interpretações. Por isso, no decorrer deste artigo, o termo "pattern" aparecerá sempre na sua forma original, em inglês. As definições para default, standard e pattern utilizadas são as seguintes:

Default

Segundo Akita (2011), default representa uma opção pré-selecionada, adotada por um programa de computador, ou outro mecanismo, quando nenhuma outra alternativa é especificada. Por exemplo, o default de um editor de texto é trazer as novas páginas em branco.

Standard

Honrby (1995, p.1161) defini a palavra standard como um nível requerido ou concordado de qualidade ou realização; um princípio moral e de comportamento; ou ainda, uma ideia ou coisa usada como medida, norma ou modelo, um parâmetro de comparação.

O standard pode ser entendido também como sendo "uma forma de linguagem que é amplamente reconhecida como sendo correta" (AKITA, 2011). No universo da música, por exemplo, utiliza-se a expressão "standard do Jazz", que segundo Santos (2012, p. 4), é qualquer composição que, tendo adquirido certa popularidade, torna-se parte do repertório que um jazzista deve conhecer. Já em hotelaria, um apartamento standard define padrões básicos de qualidade e conforto para a acomodação em determinado hotel. Portanto, standard pode ser entendido como uma norma, algo que todos concordam como base de comparação ou requerimentos mínimos a serem seguidos.

Entretanto, uma abordagem diferente é proposta por Vassão (2008), relacionando standard e pattern. Para o autor, o standard funciona como uma regra de utilização para o pattern, uma norma que delimita seus usos. "O standard opera como uma apropriação e transpõe o pattern como um stencil, carimbando-o de local em local" (VASSÃO, 2008, p. 267).

Pattern

Para Akita (2011), pattern é um desenho decorativo repetitivo, um arranjo, uma sequência regular de objetos comparáveis ou ainda, uma forma regular e inteligível. Em inglês, os desenhos, modelos e esquemas usados como guia para bordados, crochê, tricô e outras atividades artesanais também são denominados patterns. Neste caso, estes esquemas, apesar de serem chamados de "pattern" trazem, também, o sentido de standard, uma vez que apresentam as regras (instruções) para a (re)produção do trabalho.

Embora existam várias abordagens para o termo pattern, uma característica presente em todas as suas definições é a repetição. Ele é construído para ser repetido ou aplicado várias vezes, seguindo ou não a uma regra, o standard. Nesse sentido, o pattern pode ser entendido como um módulo reutilizável.

Vassão (2008) apresenta duas abordagens possíveis para o assunto: o pattern como módulo visual e o pattern como solução encapsulada. No primeiro contexto têm-se uma maior relação com a visualidade, com o resultado gráfico, aplicado amplamente em projetos de design de superfície. Já o pattern como solução encapsulada, refere-se a um conceito mais abstrato, utilizado com frequência em programação de computadores.

Pattern como módulo visual

Todos, o tecido, o papel de parede e até mesmo o revestimento do piso, apresentados na figura 1, são exemplos de projetos de um campo específico, recente e multidisciplinar do design denominado design de superfície.

O design de superfície consiste numa "atividade criativa e técnica que se ocupa com a criação e desenvolvimento de qualidades estéticas, funcionais e estruturais, projetadas especificamente para constituição e/ou tratamentos de superfícies" (RUTHSCHILLING, 2013, p. 23).

Seu principal campo de aplicação e com maior diversidade de técnicas é a área têxtil, embora atue também em papelaria (papel de parede, papel de embrulho, embalagens, etc.), cerâmica (louças e revestimentos para paredes e pisos), materiais sintéticos (plásticos laminados, adesivos, etc.) e até mesmo em superfícies e ambientes virtuais (utilizados no desenvolvimento de jogos e animações).

O design de superfície utiliza a abordagem do pattern, no sentido de módulo visual, na elaboração de seus projetos. Os módulos são recorrentes na composição, ou seja, aparecem muitas vezes, apresentando variações de tamanho, posição e até pequenas modificações formais.

A construção do módulo (pattern) e de seu sistema de repetição (standard) são considerados por Ruthschilling (2013, p. 63) como sendo os recursos de sucesso de um projeto de design de superfície. Devido às restrições do processo produtivo, é inviável estampar grandes superfícies de uma única vez. A impressão de um papel de parede ou de um tecido, por exemplo, é feita por partes e as dimensões dessas partes (módulos) são definidas

em função do equipamento utilizado na produção. Portanto, a continuidade e a contiguidade da composição visual é decorrência da habilidade do designer para projetar estes módulo e suas articulações.

Segundo Ruthschilling (2013, p. 63), com os avanços tecnológicos, o módulo e o sistema de repetição deixam de ser condições necessárias para o desenvolvimento de projetos em meios eletrônicos, mas continuam sendo conhecimentos fundamentais da área.

Módulo

Ruthschilling (2013, p. 64) define módulo como sendo a unidade da padronagem, isto é, a menor área que inclui todos elementos visuais que constituem o desenho. A composição visual criada a partir dos módulos se dá, segundo a autora, em dois níveis: dentro do módulo, em que estão organizados os elementos ou motivos e na articulação entre os módulos, de acordo com a estrutura preestabelecida de repetição.

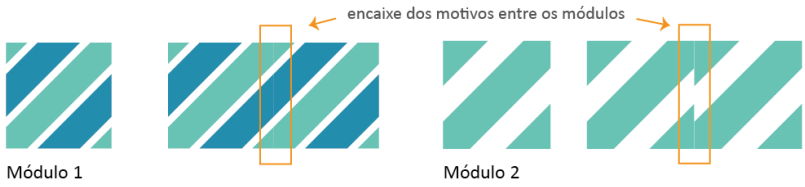

Figura 2 Exemplos de módulos e detalhe do encaixe entre dois módulos repetidos consecutivamente. O módulo 1 foi baseado em Ruthschilling (2013, p. 64) e o módulo 2 baseado no piso da linha Rep Stripe (www.pophamdesign.com).
Fonte Autores, 2015.

Na figura 2, ambos os módulos possuem o mesmo motivo geométrico: as listras, entretanto, quando esses módulos são repetidos lado a lado, na junção entre eles observa-se que para o Módulo 1 houve a preocupação em criar encaixes perfeitos, ou seja, não se percebe nenhuma emenda entre eles.

Já no Módulo 2, o encaixe não é preciso, o que não representa, necessariamente, um erro de projeto visto que o designer, dominando os elementos de composição e suas relações operacionais, pode deliberadamente optar por construir módulos que não se encaixem nas vizinhanças, mas que mantenham a fluência e o ritmo visual.

> O encaixe dos motivos entre módulos é o estudo feito prevendo os pontos de encontros das formas entre um módulo e outro de maneira que, quando justapostos de maneira predeterminada pelo sistema de repetição definido ou escolhido pelo designer, forma o desenho. (RUTHSCHILLING, 2013, p. 64)

Para o estudo do encaixe e do efeito resultante, Schwartz (2008, p. 63) propõe a utilização de um conjunto mínimo de patterns denominados "Unidade Compositiva". No caso de módulos quadrados ou retangulares, o conjunto mínimo é formado por quatro módulos (partes) adjacentes. Entretanto, Ruthschilling (2003, p. 65) recomenda a utilização de nove módulos, para evidenciar o módulo central e suas relações visuais com todos os vizinhos ao seu entorno.

Sistemas de repetição

O sistema de repetição consiste na "maneira pela qual um módulo vai se repetir a intervalos constantes" (RUTHSCHILLING, 2013, p. 67). Em outras palavras, são as regras (standard) que irão nortear a utilização do módulo (pattern), sobre uma estrutura (grid) responsável por sua organização no espaço. No caso do piso e do tecido, apresentados na figura 1, o grid utilizado é um quadriculado ou grade básica, sem deslocamentos das células, portanto, pode ser considerado como um sistema alinhado.

Em um sistema de repetição alinhado, de acordo com Ruthschilling (2013, p. 68) pode-se identificar as seguintes operações:

- translação: o módulo desloca-se sobre um eixo;
- rotação: o deslocamento do módulo é radial ao redor de um ponto; e
- reflexão: espelhamento em relação a um ou ambos os eixos.

Figura 3 Exemplos de sistemas de repetição alinhados e multimódulos.
Fonte RUTHSCHILLING, 2013, p. 69.

Os módulos, podem ser combinados gerando novos módulos mais complexos denominados de multimódulos, ou seja: "o sistema se constitui a partir de um outro sistema menor do que ele, formando diferentes desenhos e aumentando as possibilidades combinatórias." (RUTHSCHILLING, 2013, p.69).

Embora haja um consenso entre autores da área, como Munari (1998), Wong (2010) e Ruthschilling (2013), de que o grid quadriculado seja a forma de repetição dos módulos mais básica e utilizada, existem outras propostas baseadas em grades não-alinhadas que exploram o deslocamento das células, como mostrado na figura 4, ou ainda, "baseadas em sistemas progressivos, onde as células têm seu tamanho alterado progressivamente" (RUTHSCHILLING, 2013, p. 68).

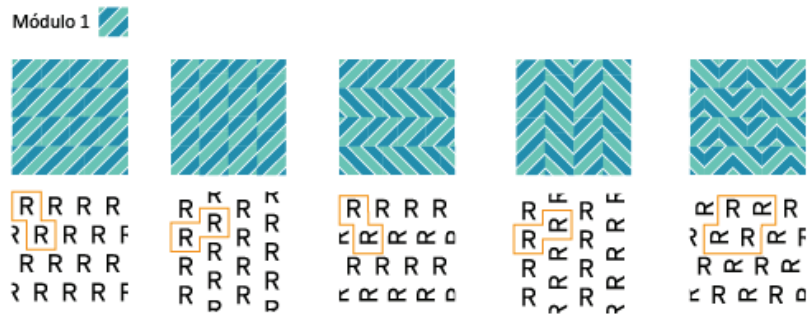

Figura 4 Sistemas de repetição não-alinhados
Fonte RUTHSCHILLING, 2013, p. 70.

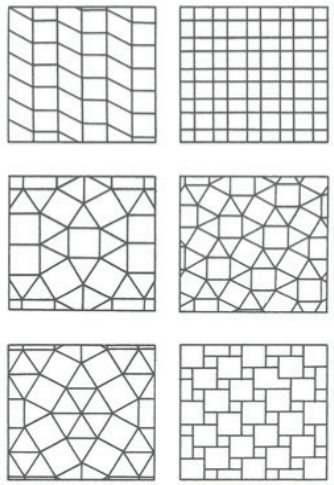

Figura 5 Exemplos de estruturas de grids não-alinhados.
Fonte WONG, 2010, p. 64.

Para sistemas de repetição não-alinhados, Wong (2010, p. 66) sugere variações na estrutura básica (como mudanças de proporção, divisões adicionais e deslizamentos), estruturas poligonais, repetição múltipla, em gradação e radiação.

Segundo Schwartz (2008, p. 66), existem ainda outras possibilidades para a repetição do módulo: a equivalência de áreas; os fractais; a pavimentação do plano ou tesselações; e, as malhas. Entretanto, a abordagem destes conceitos encontra-se fora do escopo deste artigo.

Mikado_Xplosion: pattern como motivo visual

Quando o processo produtivo não é um fator limitante da criação, pode-se gerar composições, a partir de patterns, sem a necessidade de um sistema de repetição pré-definido.

No limite entre o design de superfície e a arte, encontra-se o Mikado_Xplosion, trabalho do artista digital Pascal Dombis, para a edição do ano de 2008 da Emoção Art.ficial, bienal internacional de arte e tecnologia, realizada em São Paulo. Nessa instalação, de site específico de 22 x 15m, Dombis adesivou com sua obra a fachada do edifício do Itaú Cultural na Avenida Paulista.

Por meio de algoritmos, o computador repete o pattern - no caso as linhas - um milhão de vezes, variando aleatoriamente suas posições e cores e, é desse processo que emerge a obra, imprevisível.

O nome "Mikado" faz referência à uma brincadeira de criança, uma versão do jogo pega-varetas. Dombis diz que da mesma forma que para iniciar o jogo deve-se soltar as varetas e deixá-las cair ao acaso, é exatamente isso que ele recria digitalmente, como se tivesse jogado milhares de linhas na fachada do edifício.

Figura 6 Mikado_Xplosion, Pascal Dombis, 2008
Fonte dombis.com

Pattern como solução encapsulada

Em 1977, o arquiteto Christopher Alexander propôs a utilização, em projetos de arquitetura e urbanismo, de um conjunto de 253 patterns que receberam o nome de "Pattern Language" e deram origem a um livro, lançado em português sob o título "Uma Linguagem de Padrões". Alexander acreditava que fazendo uso da linguagem de padrões era possível projetar desde simples residências até cidades inteiras.

Os patterns de Alexander, diferentemente do que foi apresentado anteriormente, consistiam em abstrações, soluções genéricas de problemas recorrentes durante o projeto em contextos particulares.

> Cada padrão descreve um problema que ocorre repetidas vezes em nosso meio ambiente e então descreve o ponto central da solução do problema, de modo que você possa usar a mesma solução milhares de vezes, mas sem jamais ter de repeti-la da mesma maneira. (ALEXANDER, 2012, p. xiv)

Seus patterns não são puramente visuais mas continuam trazendo como características básicas a modularidade e a reutilização. Alexander (2012, p. xiv) esclarece que, por uma questão de conveniência e clareza, todos os patterns são apresentados no livro em um mesmo formato contendo: o nome, uma fotografia, o contexto, que explica como o pattern ajuda a completar padrões maiores, o problema, a solução - que sempre aparece na forma de uma instrução - e, por fim, um diagrama, que funciona como uma representação gráfica do próprio pattern.

A abordagem dos patterns de Alexander (2012) foi, segundo Vassão (2008, p. 189), uma das referências fundamentais para o desenvolvimento da ideia de reutilização de componentes computacionais.

Em projetos que se utilizam da linguagem de padrões, seja na arquitetura, no urbanismo, na computação, na indústria ou em qualquer outro sistema complexo característico das sociedades contemporâneas, o sucesso do projeto é decorrência da "capacidade de um designer ou projetista em identificar e reconhecer uma entidade que pode ser convertida em um módulo funcional". (VASSÃO, 2008, p. 189)

Caixa-preta, módulo e objeto

O conceito de caixa-preta, no contexto da cibernética, é apresentado por Vassão (2008, p. 130) como sendo um módulo funciona cujo conteúdo é intencionalmente ignorado, mas seus contatos com o mundo exterior são cuidadosamente estudados.

A afirmação de Vassão pode ser melhor entendida por meio de um exemplo simplificado. Supondo que um web designer, ao construir um site de uma loja virtual, encontre o seguinte problema: quando o cliente procurar por um produto, o sistema precisa mostrar a foto, a descrição, o preço e sua disponibilidade em estoque. Então, a partir do nome do produto, o

designer desenvolve um algoritmo[1] que consulta o banco de dados da loja e traz as quatro informações (foto, descrição, preço e disponibilidade) do produto na tela.

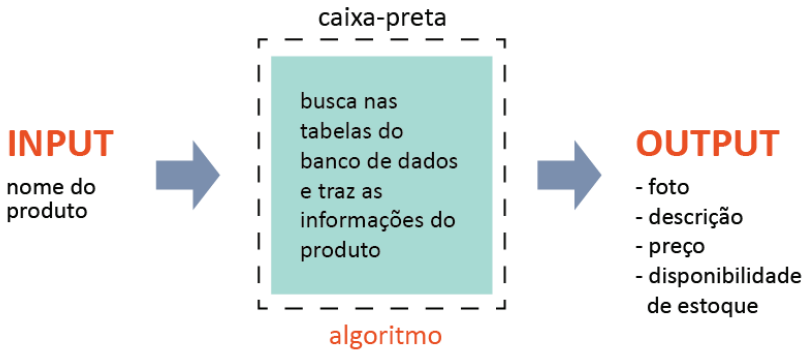

Figura 7 Exemplo de módulo funcional
Fonte Autores, 2015.

O mesmo algoritmo será utilizado diversas vezes no site enquanto os clientes procuram pelos produtos. Antes de se efetivar uma compra, é preciso verificar o estoque, logo, o código será usado novamente. O designer pode, ainda, aplicar essa mesma solução em outros sites semelhantes que ele desenvolver.

Portanto, se esse algoritmo for bem planejado, será escrito (programado) uma única vez e reutilizado diversas vezes. Trata-se, então, de um módulo, um pattern, um componente reutilizável, também conhecido nas ciências da computação como "objeto".

Depois de pronto, o desenvolvedor não precisa mais se preocupar com o código interno do objeto, seu conteúdo pode ser intencionalmente ignorado, desde que se saiba para que o módulo serve e quais são seus parâmetros de entrada (input) e sua saída (output).

Diz-se então que o código foi encapsulado. Isto é, o "pattern" opera segundo o conceito de caixa-preta, em que os parâmetros de entrada e saída funcionam como conectores entre o módulo e o restante do sistema.

> O módulo funcional, resumidamente, é uma representação de seus inputs e outputs, ao ponto de seu conteúdo poder ser ignorado, contando com as funções desempenhadas de acordo com as especificações dadas. (VASSÃO, 2008, p. 130)

No exemplo dado, o nome do produto seria o parâmetro de entrada (input) do módulo, enquanto a fotografia, a descrição, o preço e a disponibilidade de estoque são os parâmetros de saída (output). Alterando-se o input, altera-se o output. Ou seja, para cada produto consultado, o mesmo código trará a fotografia, a descrição, o preço e a disponibilidade de estoque correspondentes a este produto.

O fato do módulo trazer como resposta essas quatro informações (outputs), não significa que todas elas precisem, necessariamente, ser utilizadas.

1 A programação dos computadores se dá através de códigos denominados algoritmos, que são conjuntos de instruções suficientemente detalhadas e precisas para serem executadas pelo computador

Por exemplo, no contexto específico da confirmação da venda, só interessará como resultado a informação referente à disponibilidade do estoque.

Esse exemplo simples, evidencia como a programação orientada a objetos, que é uma abordagem computacional baseada na modularização, pode reduzir o esforço de programação e reprogramação, devido a possibilidade da reutilização dos módulos. Outro "[...] aspecto importante é que os objetos permitem uma infinidade de combinações entre si". (VASSÃO, 2008, p. 133)

Design Patterns

Na opinião de Grady Booch, "uma das maneiras de medir a qualidade de um sistema orientado a objetos é avaliar se os desenvolvedores tomaram bastante cuidado com as colaborações comuns entre os objetos". (BOOCH, 2007 apud GAMMA et al., 2007). O autor acredita que o emprego dos padrões no projeto leva a uma arquitetura menor, mais simples e mais compreensível.

Em 1995, um grupo de programadores denominado Gang of Four (GoF), formado por Erich Gamma, Richard Helm, Ralph Johnson e John Vlissides propõe uma coleção de patterns para problemas e soluções recorrentes[2]. Na opinião de Vassão (2008, p. 135), os autores dão continuidade às propostas lançadas por Alexander.

Os autores perceberam na medida em que se acumulam experiências em projetos usando objetos, observa-se a repetição de determinadas colaborações entre eles que independem da linguagem de programação ou da tecnologia utilizadas. Ao criar e catalogar essas soluções de projeto (design patterns) que consideravam representativas, estas se tornam referência no universo da computação.

Gamma et al. (2007, p. 328) traça um comparativo entre seus "design patterns" e a "pattern language" proposta por Alexander. Ambas são baseadas na observação de sistemas existentes e na busca por padrões neles recorrentes. Embora diferentes, se utilizam de gabaritos para descrever os padrões.

O trabalho de Gamma et al. (2007) e de Alexander (2012) "[...] dependem de linguagem natural e de muitos exemplos para a descrição dos padrões, em detrimento do uso de linguagens formais e ambos os trabalhos fornecem motivação (rationales) a cada padrão". (GAMMA et al., 2007, p. 328).

Entretanto, os seus trabalhos se diferem, segundo o autor, nos seguintes pontos: Alexander se baseia na experiência de milhares de anos de edificações para criar seus patterns enquanto Gamma olha para um passado bem mais recente; Alexander dá uma ordenação segundo a qual seus patterns devem ser usados, Gamma não; Os padrões de Alexander enfatizam o problema enquanto os design patterns descrevem as soluções com mais detalhes; e, por fim, Alexander afirma que o uso de seus padrões gera edificações completas, mas os padrões de Gamma não geram programas completos.

2 Em seu livro lançado em português sob o título de "Padrões de Projeto: soluções reutilizáveis de software orientado a objetos" apresentam os 23 design patterns por eles criados e agrupados nas categorias de: padrões de criação, padrões estruturais e padrões comportamentais.

Patterns no design de jogos

Tomando o game SimCity (Eletronic Arts, 1989) como exemplo, a partir do ponto de vista do jogador, sem se prender a complexidade inserida no desenvolvimento do jogo, pode-se perceber a possibilidade de utilização dos patterns no design de games. SimCity, é uma série de jogos de computador de simulação e construção de cidades, criada por Will Wright[3], e tem como objetivo básicos criar e gerir uma cidade.

Traçando um paralelo com o que foi visto até o momento sobre pattern, tem-se o terreno, que no início do jogo se apresenta vazio – sem construções, apenas a topografia - que pode ser visto como um grid a ser continuamente preenchido por elementos arquitetônicos (casas, prédios, etc) e urbanísticos (pontes, ruas, praças, etc.), ou seja, patterns que o usuário escolhe livremente, dentre as opções oferecidas pelo jogo.

Um mesmo elemento pode ser repetido diversas vezes ao longo da cidade, mantendo sua forma mas, eventualmente, alterando sua orientação, como destacado na figura 9. É da repetição e combinação desses módulos que surgem cidades únicas.

Entretanto, os módulos aplicados no jogo não são apenas visuais. Um prédio, por exemplo, não se resume a sua aparência arquitetônica, atrelados a ele estão outros atributos não visíveis que são essenciais para o jogo, tais como o custo da construção, os recursos que o prédio consome, sua vida útil, etc, operando os patterns em outro nível de abstração. O conjunto desses atributos associados ao módulo visual funciona como solução encapsulada, que possibilita a trama do jogo acontecer, caso contrário, ele ficaria apenas na visualidade.

Figura 8 Exemplo de cidade criada no SimCity 4
Fonte www.shacknews.com/article/73062/simcity-preview

3 Will Wright, designer de games americano, é conhecido também por seus outros sucessos: a série de jogos The Sims (2000) e Spore (2008).

Pattern e tipografia

A tipografia[4], de forma semelhante ao exemplo anterior, pode ser considerada como um conjunto de patterns que ao mesmo tempo são visuais e soluções encapsuladas que criam um código de representação para a fala.

A partir da combinação de apenas 26 letras (módulos) do alfabeto latino seguindo as regras impostas pelo idioma (standard) são formados todos os vocábulos da língua portuguesa. Essas palavras, que podem ser consideradas como multimódulos, quando combinadas novamente segundo outras regras específicas, geram os textos, a comunicação escrita.

Um caracter, seja ele uma letra, um número ou um sinal, pode ser "desenhado" em vários estilos desde que mantenham algumas características fundamentais que permitam seu reconhecimento. A figura 9 mostra a letra "a" de várias fontes tipográficas que apesar de apresentarem variações em seu desenho, continuam mantendo sua identificação como sendo um "a".

No caso, trata-se de opções de fontes digitais, mas o mesmo aconteceria com a tipografia mecânica, com a máquina de escrever, com a caligrafia artística ou com a escrita manual. Remetendo a ideia de Alexander (2012) de que uma solução pode ser utilizada vária vezes sem repetir, necessariamente, sua forma (desenho).

4 Entende-se por fonte tipográfica "o conjunto de caracteres que incluem letras, algarismos, pontuações e sinais que fazem parte de um determinado estilo." (CLAIR, 2009, p. 362)

Figura 9 Letras "a" nas tipografias Myriad Pro, Rufina, Renaissance, Quadranta e Impact
Fonte Autores, 2015.

Uma fonte tipográfica digital, entretanto, pode ser pensada apenas como um conjunto de módulos visuais, desconsiderando-se a função de solução encapsulada dos caracteres. Nessa abordagem, esses patterns, podem ser utilizados para criar composições visuais.

O artista coreano Sung-Jin Kin utiliza os ornamentos da fonte Bodoni, para criar a imagem apresentada na figura 11. Ele parte de glifos (desenhos) simples para compor elementos complexos, cuja combinação faz surgir a obra.

Já a designer Marina Chaccur criou, em 2005, a Plants of Mine:

> (...) uma fonte de dingbats produzida a partir do desenho de plantas suculentas, e baseada no estudo de ornamentos tipográficos. As fontes podem ser utilizadas como ilustrações, bordas e para criar padrões (CHACCUR, 2005).

A fonte Plants of Mine, cujos glifos são formados pelos desenhos das pétalas da suculenta, foi aplicada para gerar digitalmente uma textura com o auxílio de um programa de computador (figura 10). Nessa composição, a designer utiliza dois módulos distintos: as folhas em preto e as brancas, que se repetem em uma estrutura de gradação e radiação, como sugerido por Wong (2010, p. 90).

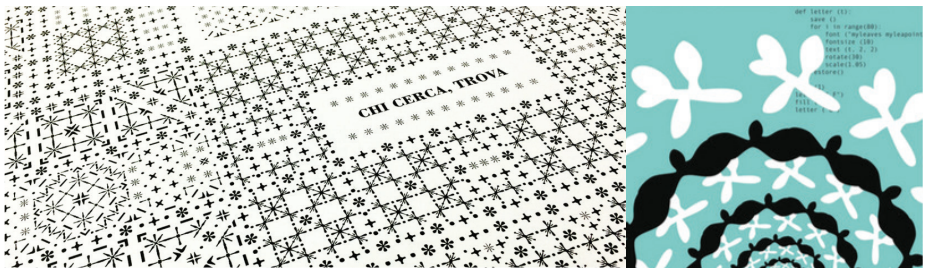

Figura 10 A esquerda: Composição formada por ornamentos da fonte Bodoni, Sung-Jin Kin (Fonte: www.minograph.com). A direita: Python Texture (CHACCUR, 2007)

A tipografia criada por Chaccur não possui letras, algarismos, pontuações ou sinais, apenas desenhos de pétalas. Como a própria autora afirmou, trata-se de uma fonte de dingbats.

> As fontes dingbats não são exatamente uma categoria tipográfica, mas considerando que elas assumem posições no teclado, e são produzidas no formato de fonte e comercializadas por Typefoundries, podem ser consideradas como tal. (ROCHA, 2012, p. 86)

Dingbats são, portanto, "marcas decorativas, sinais de impressão ou símbolos fornecidos da mesma maneira que uma fonte tipográfica" (CLAIR, 2009, p. 360). Enquanto dingbats são ícones ou imagens figurativas, as fontes ornamentais são compostas por desenhos utilizados na decoração do texto.

> Ornamentos têm sido utilizados para compor padrões gráficos (patterns), molduras (borders) e também podem ser usados isoladamente como vinhetas. Cumprem a função de estruturar e valorizar o texto (ROCHA, 2012, p. 100).

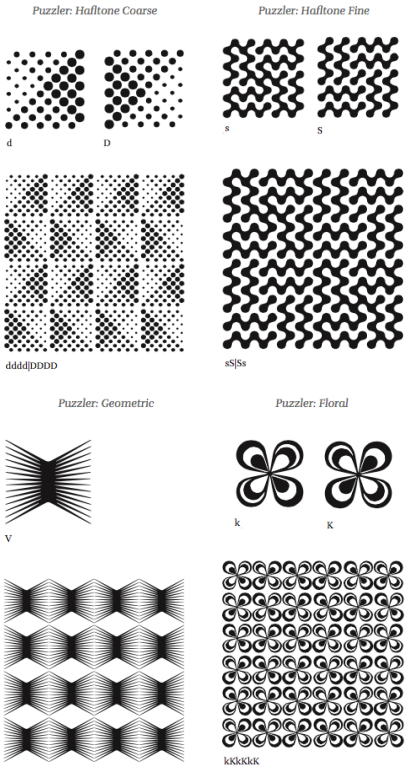

Figura 11 Família tipográfica *Puzzler*
Fonte LICKO, 2005.

Algumas fontes ornamentais têm seus glifos (módulos) especialmente planejados para formarem composições visuais e não um texto escrito, como é o caso de algumas das tipografias criadas por Zuzana Licko: a Whirligig (1994), a Hypnopaedia (1997) e a Puzzler (2005). Reas (2010, p.60), por exemplo, afirma que criar composições visuais a partir da Whirligig, por ela ser "empacotada" como uma fonte tipográfica, é tão simples quanto digitar.

O mesmo pensamento se aplica às demais fontes ornamentais. Como os módulos estão associados às letras do teclado, ao se digitar, os desenhos vão se repetindo lado a lado, gerando composições. A figura ao lado apresenta a Puzzler de Zuzana Licko, uma família tipográfica composta por quatro fontes, que totalizam 140 elementos criados com o propósito de serem repetidos gerando composições visuais.

Sobre a Puzzler, Licko comenta que "[...] cada elemento é acessado através de uma letra do teclado, tornando fácil a repetição dos desenhos" (LICKO, 2005, p. 2). O comentário de Licko (2005), assim como o de Reas (2010), destacam a facilidade do uso e reuso dos módulos uma vez que foram constituídos como uma fonte tipográfica digital que pode ser utilizada por diversos programas, desde o Word até o Illustrator ou Photoshop, basta que ela esteja instalada no computador.

Portanto, a tipografia funciona como um receptáculo para os módulos, um lugar onde eles ficam reunidos e prontos para serem reutilizados, quantas vezes forem necessárias ou desejadas, conservando seu desenho original.

A cada repetição cria-se uma cópia deste original que pode ser modificada por meio de seus parâmetros de cor, tamanho, posição, etc. Além disso, os patterns podem se intercambiar, permitindo uma infinidade de combinações potenciais entre si.

Considerações finais

Da investigação realizada surgem duas principais abordagens para o pattern que foram discutidas no decorrer deste artigo: o pattern como módulo visual e como solução encapsulada.

A primeira delas, intrinsicamente ligada a visualidade, é aplicada amplamente pelo design de superfície, no qual os módulos geralmente devem respeitar um sistema de repetição (standard) pré-definido. Porém, em alguns casos, as regras de repetição podem dar lugar à aleatoriedade, fazendo com que surjam composições imprevistas ou inesperadas.

Já o pattern como solução encapsulada é aplicado nas ciências da computação e se apresenta como uma ferramenta promissora para ser aplicada em projetos complexos, aproveitando-se do seu caráter modular e reutilizável.

Este artigo apresenta apenas alguns exemplos de aplicações do pattern, restritos as áreas do design de superfície, de games e gráfico, além de comentar sobre arquitetura, arte e computação. Mas, devido a sua aplicabilidade em diversos contextos, novas investigações podem ser desenvolvidas tendo como foco outras áreas do conhecimento.

A partir da discussão resultante desta pesquisa, conclui-se que a utilização do termo "pattern" sem uma devida contextualização, pode realmente gerar problemas de interpretação, em função dos motivos apresentados no decorrer deste texto. O que nos leva a dificuldade que deu origem a esta investigação: como podem ser nomeadas as composições visuais apresentadas na figura 1, já que não poderiam ser chamadas de pattern? O termo "padronagem" pode ser uma opção.

Segundo Gubert (2011, p. 71), padronagem é uma composição visual que possui como característica fundamental a clara recorrência ou repetição de formas e demais elementos gráficos, projetada para ser aplicada sobre superfícies. Lupton e Phillips (2008), respondendo à questão inicial desta pesquisa, descrevem o conceito de padronagem da seguinte forma:

> Ela pode ser produzida manualmente, por máquinas ou códigos, mas é sempre resultado de uma repetição. Um exército de pontos pode ser regulado por um grid geométrico rígido ou agrupado ao acaso ao longo de uma superfície seguindo marcas irregulares feitas a mão [...] os padrões seguem alguns princípios repetitivos, sejam eles editados por um grid mecânico, um algoritmo digital ou pelo ritmo físico da ferramenta de um artesão que trabalhe sobre a superfície. (LUPTON E PHILLIPS, 2008. P. 187)

Referências

AKITA, F. **A língua portuguesa é péssima**: standard vs pattern. Editora Abril, 2011. Artigo. Disponível em: http://info.abril.com.br/noticias/rede/gestao20/software/a-lingua-portuguesa-brasileira-e-pessima-standard-vs-pattern/. Acessado em out/2015

ALEXANDER, C. et al. **Uma linguagem de padrões**: a pattern language. Porto Alegre: Bookman, 2012.

BOOCH, G. Apresentação. In: GAMMA et al. **Padrões de Projeto**: soluções reutilizáveis de software orientados a objetos; Porto Alegre: Bookman, 2007.

CHACCUR, Marina. **Site da designer**. 2007. Disponível em www.marinachaccur.com.br . Acessado em: set/2014.

CLAIR, Kate. **Manual de Tipografia**: a história, as técnicas e a arte. Porto Alegre: Bookman, 2009.

GAMMA, E. et al. **Padrões de Projeto:** soluções reutilizáveis de software orientados a objetos; Porto Alegre: Bookman, 2007.

GUBERT, M.L. **Design de Interiores**: a padronagem como elemento compositivo no ambiente contemporâneo; 2011. Dissertação (Mestrado em Design) - Faculdade de Arquitetura, Universidade Federal do Rio Grande do Sul, Porto Alegre.

HONRBY, A. Oxford **Advanced Learner's Dicitionary of Current English**; Oxford: Oxford University Press, 1995.

LICKO, Zuzana . **Catalogo Puzzler**, 2005. Disponível em www.emigre. com, Acessado em nov/2015.

LUPTON, Ellen; PHILLIPS, Jennifer Cole. **Novos Fundamentos do Design**. São Paulo: Cosac Naify, 2008

MUNARI, Bruno. **Das Coisas Nascem Coisas**. São Paulo. Martins Fontes, 1998.

REAS, Casey et al. **Form + Code**: in design, art and architeture; New York: Princeton Arquitetural Press, 2010.

ROCHA, Claudio. **Novo Projeto Tipográfico**; São Paulo: Editora Rosari, 2012.

RUTHSCHILLING, Evelise. **Design de Superfície**; Porto Alegre: Editora UFRGS, 2013.

SANTOS, Fábio. **As funções da harmonia e da melodia na bossa nova e no jazz**; 2012. Artigo – Instituto de Artes, UNICAMP, Campinas.

SCHWARTZ, Ada Raquel D. **Design de Superfície**: por uma visão projetual geométrica e tridimensional; 2008. Dissertação (mestrado) – Faculdade de Arquitetura, Artes e Comunicação, UNESP, Bauru.

VASSÃO, Caio Adorno. **Arquitetura Livre**: complexidade, metadesign e ciência nômade, Tese (doutorado) – Faculdade de Arquitetura e Urbanismo, Universidade de São Paulo, São Paulo, 2008

WONG, Wucius. **Princípios de forma e desenho**; São Paulo: Editora WMF Martis Fontes, 2010.

Recebido: 22 de Agosto de 2016

Aprovado: 10 de Outubro de 2016