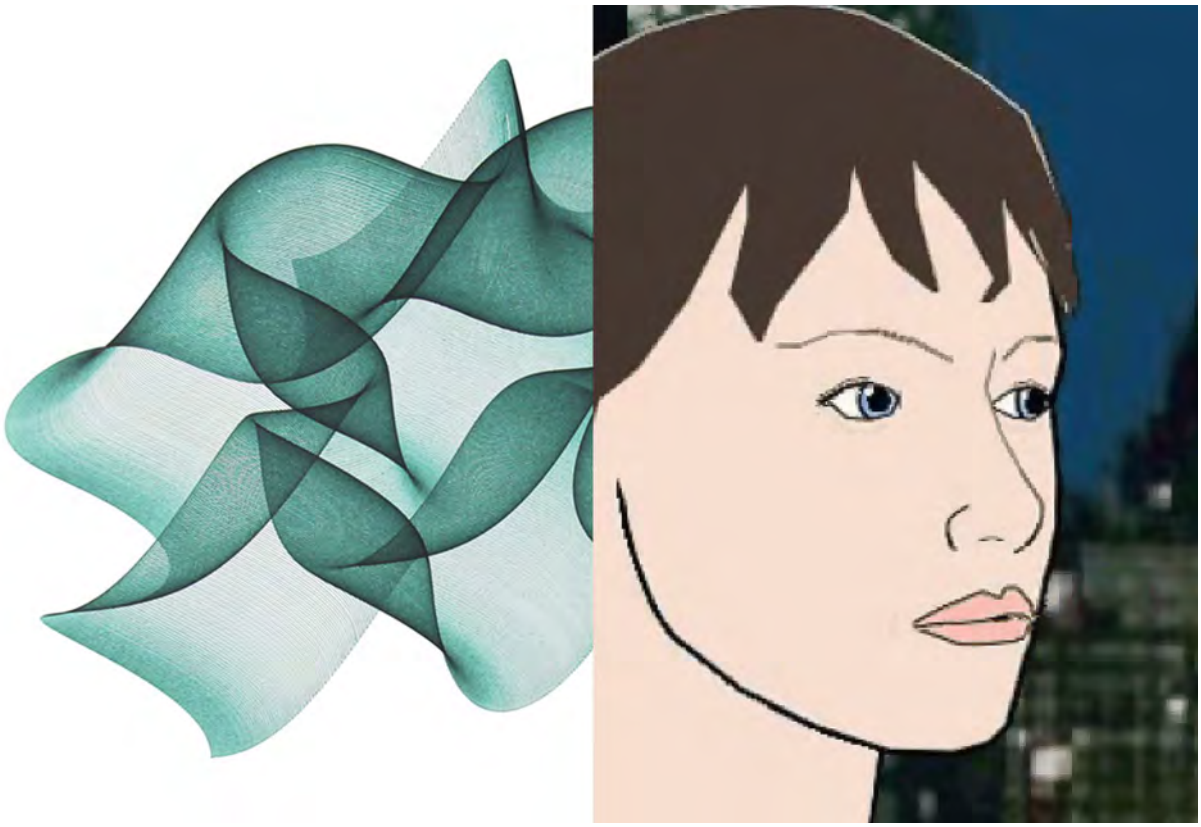
Daniel Peixoto Ferreira

Expressividade Procedural

narrativas e jogos digitais



Escola de Comunicações e Artes

Universidade de São Paulo

ECA/ USP

São Paulo | Março de 2016

Daniel Peixoto Ferreira

Expressividade Procedural

narrativas e jogos digitais

Tese apresentada como exigência parcial para a obtenção do título de Doutor em Artes Visuais.

Orientador:

Prof. Dr. Gilberto dos Santos Prado

Programa:

Pós-Graduação em Artes Visuais

Área de Concentração:

Poéticas Visuais

Linha de Pesquisa:

Multimeios

Escola de Comunicações e Artes
Universidade de São Paulo
ECA/ USP

São Paulo | Março de 2016

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

Catalogação na publicação

Serviço de Biblioteca e Documentação

Escola de Comunicações e Artes da Universidade de São Paulo

(Dados fornecidos pelo autor)

Ferreira, Daniel Peixoto

Expressividade Procedural: narrativas e jogos digitais / Daniel Peixoto Ferreira. – São Paulo: D. P. Ferreira, 2016.

277 p.: il.

Tese (Doutorado) - Programa de Pós-Graduação em Artes Visuais - Escola de Comunicações e Artes / Universidade de São Paulo.

Orientador: Gilberto dos Santos Prado

Bibliografia

1. Expressividade Procedural 2. Narrativas Digitais 3. Jogos Digitais 4. Criação Digital I. Prado, Gilberto dos Santos II. Título.

CDD 21.ed. - 700

Apoio:
CAPES - Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico

Ficha catalográfica: gerada a partir do sistema online da Biblioteca da ECA.

Capa: composição das obras:
Algorithmic Poetry for a Three Story Wall (Roman Verostko, 2011)
e Façade (Michael Mateas e Andrew Stern, 2005).

Nome: Daniel Peixoto Ferreira

Título: Expressividade Procedural: narrativas e jogos digitais

Tese apresentada como exigência parcial para a obtenção do título de Doutor em Artes Visuais, sob orientação do Prof. Dr. Gilberto dos Santos Prado. Programa: Pós-Graduação em Artes Visuais. Área de Concentração: Poéticas Visuais. Linha de Pesquisa: Multimeios. Escola de Comunicações e Artes. Universidade de São Paulo.

Aprovado em

São Paulo, _____ de _____ de _____

Banca Examinadora

_____

_____

_____

_____

_____

Ao apoio incondicional da minha família.

À companhia e carinho dos meus amigos.

À dedicação, interesse e energia dos parceiros criativos.

Agradecimentos

Ao meu orientador Gilbertto Prado, o apoio e confiança.

Ao coorientador da minha pesquisa no exterior na University College London, Andrew Burn, bem como a Alison Gazzard, a calorosa acolhida.

A Silvia Laurentiz e Milton Sogabe as valiosas contribuições na qualificação.

Aos professores, colegas, membros do Grupo Poéticas Digitais (ECA/USP), amigos e familiares, a discussão, estímulo e inspiração.

Ao Programa de Pós-Graduação em Artes Visuais da ECA/USP,
a Monica Tavares, o Programa Ciência Sem Fronteiras,
a University College London e o London Knowledge Lab,
a infraestrutura e apoio.

À CAPES e o CNPq o apoio financeiro.

Resumo

FERREIRA, Daniel Peixoto. Expressividade Procedural: narrativas e jogos digitais. Tese (Doutorado) - Escola de Comunicações e Artes, Universidade de São Paulo, 2016.

Esta tese trata a respeito de uma abordagem criativa específica em mídias digitais: o emprego **expressivo** da **autoria procedural** (isto é, da criação ou manipulação de algoritmos). Esta abordagem, denominada aqui como **expressividade procedural**, difere fundamentalmente do emprego do computador unicamente para emular formas e linguagens tradicionais. Nossa investigação inclui estudos práticos realizados no contexto do projeto Playing Beowulf (University College London), no qual trabalhei como parte da equipe de pesquisa e desenvolvimento do MissionMaker, um aplicativo educacional para criação de **narrativas e jogos digitais**. Esperamos com esta tese contribuir na compreensão sobre o uso expressivo da autoria procedural em mídias digitais, e no seu papel fundamental na realização da **expressividade própria** deste meio.

Palavras-chave: expressividade procedural, narrativas digitais, jogos digitais, criação digital

Abstract

FERREIRA, Daniel Peixoto. Procedural Expressiveness: digital narratives and games. Thesis (doctoral) - School of Communication and Arts, University of São Paulo. 2016.

This thesis is about a specific creative approach in digital media: the **expressive** use of **procedural authorship** (i.e., the creation or manipulation of algorithms). This approach, which we call **procedural expressiveness**, distinguishes itself in a fundamental manner from the use of computers solely to emulate traditional forms and languages. Our investigation includes practical studies conducted in the context of the Playing Beowulf project (University College London), where I worked as part of the research and development team of MissionMaker, an educational software for creating **digital narratives** and **games**. With this thesis we hope to contribute in understanding the expressive use of procedural authorship in digital media, and its fundamental role in achieving the **unique expressiveness** of this medium.

Keywords: procedural expressiveness, digital narratives, digital games, digital art

Sumário

1 Introdução **17**

2 Expressividade Procedural **23**

2.1 Apresentação 25

2.2 Referências Teóricas 35

2.2.1 Os Algoristas (Jean Pierre Hebert, Roman Verostko e outros) 37

2.2.2 Inteligência Artificial Expressiva (Michael Mateas e outros) 43

2.2.3 Processamento Expressivo (Noah Wardrip-Fruin) 57

2.2.4 Unidades Operacionais (Ian Bogost) 62

2.3 Considerações e Esclarecimentos 73

2.4 Metodologia de Análise (uma proposta) 79

3 Playing Beowulf (Estudos Práticos) **87**

3.1 Sobre o Projeto 91

3.1.1 Beowulf 92

3.1.2 MissionMaker 97

3.1.3 Oficina Piloto 101

3.2 Análise dos Jogos 103

3.2.1 O Servo Fiel (Andrew Smith) 107

3.2.2 A Caveira de Æschere (Alice Hawkins e Soraya Shabaan) 115

3.2.3 Voltando Atrás (Harriet Piercy) 119

3.2.4 Vingando Æschere (Katie Smith) 125

3.2.5 Através do Pântano (Laura Scott) 127

3.2.6 A Recompensa (Niamh Hickey) 133

3.2.7 Comer, Beber, Dormir (Tracey Matthews) 137

3.3 Análise do Aplicativo: MissionMaker 149

3.3.1 Análise Multimodal (Metodologia) 151

3.3.2 Recursos Modais Procedurais 155

3.3.3 Compondo Modos 185

3.3.4 Comparando Modos (Recursos Compartilhados) 189

3.4 Algumas Considerações 203

3.5 Futuros Desenvolvimentos 207

3.5.1 MissionMaker2 207

3.5.2 Próximas Iterações 217

4 Pelo Uso Expressivo do Meio Digital **223**
4.1 Abertura Local e Linearidade Global 227
4.2 Escolha de Escopo 235
4.3 Coerência na Mecânica de Jogo 241

5 Considerações Finais **247**

Referências **259**

Glossário **273**

Lista de Figuras **275**

1 Introdução

Esta pesquisa tem como **objeto** a expressividade procedural, que trata-se do emprego expressivo ou artístico das mídias digitais a partir da sua programabilidade e autonomia, isto é, quando o artista ou criativo exerce autoria procedural, criando ou manipulando algoritmos (em oposição ao emprego do meio digital unicamente para emular formas e práticas tradicionais).

O **objetivo principal** desta tese é de contribuir para a compreensão dos mecanismos e estratégias procedurais de criação em mídias digitais, nas quais reside o potencial expressivo próprio deste meio, que o distingue das demais formas e linguagens tradicionais. Informamos e embasamos este percurso central de pesquisa a partir da investigação e mapeamento desta abordagem criativa e expressiva, através de análises e elaborações teóricas, realizadas sobre o trabalho de artistas, criativos e estudiosos na área. Também realizamos estudos práticos no recorte específico das narrativas e jogos digitais, com base em um aplicativo para criação de jogos voltado para fins educacionais.

Nossa investigação se dá no encontro entre a **expressividade** do artista ou criativo e os mecanismos da **lógica computacional**, sem enfatizar um aspecto sobre o outro. Convém notar que, como veremos, apesar do viés educacional do projeto no qual baseamos os estudos práticos, nosso foco não é no seu aspecto pedagógico, mas sim no processo criativo.

Além disso, a autoria procedural nos interessa principalmente a partir da **perspectiva do artista ou criativo**, em oposição ao momento da execução do algoritmo ou a recepção por parte do público. Estes momentos são considerados unicamente como meio de acesso ao processo criativo e o projeto de criação imaginado pelo autor (que pode incluir um modelo de recepção). O papel do receptor nos interessa principalmente a partir do grau de agência planejado e implementado pelo criador - isto é, os mecanismos e procedimentos que viabilizam interagir e eventualmente interferir no sistema. O mesmo se aplica à questão da interatividade de uma maneira mais ampla, que, apesar de ser um elemento fundamental em determinados tipos de produção digital (como é o caso das narrativas e jogos), trata-se de um aspecto funcionalmente secundário à proceduralidade. Como veremos, no contexto mais amplo da expressividade procedural, a interatividade nem sempre está presente, como no caso de produções generativas textuais, visuais ou sonoras.

Isto é, nesta pesquisa consideramos a recepção e a interatividade, entretanto fazemos isso principalmente a partir da sua modelagem no contexto do sistema ou simulação, da perspectiva do artista ou criativo.[1]

[1] Considerar a interatividade como um elemento secundário à proceduralidade do ponto de vista da lógica computacional não significa que este seja um aspecto menos relevante. Autores como Ian Bogost e Lev Manovich tratam sobre a interatividade para discutir tópicos como ética e política nas mídias digitais. O pesquisador Miguel Sicart examina jogos digitais especificamente a partir da interatividade (SICART, 2011). A interatividade também é central no trabalho de muitos artistas, como é o caso dos coletivos Grupo Poéticas Digitais (ECA/USP) e SCIArts, por exemplo.

Percurso da Tese

Este estudo dá seguimento a uma pesquisa anterior, realizada em nível de mestrado, na qual tratamos sobre os elementos e questões mais fundamentais envolvidos na expressividade procedural, examinando a natureza da mente criativa e da máquina digital, e refletindo sobre o relacionamento entre arte e técnica no processo criativo (FERREIRA, 2011).[2]

No percurso da presente tese resgataremos alguns dos tópicos e conceitos daquele estudo anterior, atualizando e recontextualizando-os de acordo com os novos desenvolvimentos e recorte desta pesquisa. Naquela pesquisa anterior, uma das reflexões principais trata sobre a importância das estratégias procedurais de criação na consolidação de uma expressividade própria do meio digital. Nesta pesquisa atual daremos seguimento, elaboraremos e aprofundaremos sobre esta reflexão.

Neste **capítulo 1**, introduzimos o objeto, os objetivos principal e secundários, bem como o foco deste estudo. Também é apresentado este percurso da tese, e a abordagem adotada na pesquisa.

No **capítulo 2** (p. 23), apresentaremos a **expressividade procedural**, definindo e descrevendo os fundamentos desta abordagem, incluindo um breve histórico no contexto da arte em mídias digitais. Também expandimos a investigação a partir de quatro **referências teóricas principais** (p. 35). Em seguida, realizaremos algumas **considerações e esclarecimentos** (p. 73) sobre o objeto desta pesquisa. Finalmente, apresentaremos nossa **metodologia de análise** (p. 79), baseada em uma proposta elaborada na nossa pesquisa anterior, desenvolvida especialmente para o estudo das manifestações criativas e expressivas em meio digital.

No **capítulo 3** (p. 87) apresentaremos os estudos práticos feitos durante o período sanduíche desta pesquisa, como parte do projeto Playing Beowulf, no University College London (Reino Unido), realizados com base em um aplicativo para criação de jogos voltado para o ensino da programação.

Primeiro, trataremos **sobre o projeto** (p. 91), introduzindo o poema Beowulf, o aplicativo MissionMaker e a oficina piloto que serviu de base para os estudos práticos. Em seguida, apresentaremos a **análise dos jogos** (p. 103), incluindo a investigação do processo criativo dos participantes durante a oficina, bem como entrevistas conduzidas com eles. Depois, realizaremos a **análise do aplicativo** (p. 149), examinando e identificando as estratégias procedurais de criação presentes na ferramenta, e comparando o processo criativo com o de outras formas e linguagens tradicionais. Realizaremos **algumas considerações** (p. 203) sobre os estudos práticos, e finalmente apresentaremos os **futuros**

[2] Naquela pesquisa anterior, o objeto central de análise era denominado como "poéticas procedurais". O conceito de "poética" refere-se a um determinado modo de criar, proposta ou postura artística, que pode estar associada a uma obra, artista, prática ou ferramenta. Nesta pesquisa atual predomina o conceito mais amplo da expressividade, embora as práticas investigadas também se apliquem às manifestações artísticas e poéticas.

desenvolvimentos (p. 207) previstos para o projeto, incluindo a nova versão do aplicativo, bem como suas próximas iterações, que refletem o trabalho de pesquisa e desenvolvimento realizado neste período.

No **capítulo 4** (p. 223) retomaremos os temas e reflexões principais da pesquisa, aprofundando a análise sobre a questão da consistência e coerência nas estratégias procedurais de criação, e no seu papel na consolidação de uma expressividade própria das mídias digitais. Trataremos sobre os temas da **abertura local e linearidade global** (p. 227), **escolha de escopo** (p. 235) e **coerência na mecânica de jogo** (p. 241).

Finalmente, o **capítulo 5** (p. 247) consiste nas nossas considerações finais, resgatando os temas centrais da pesquisa, explorando alguns dos questionamentos que resultaram das investigações e apresentando as principais contribuições que esperamos ter feito com este estudo.[3]

Abordagem de Pesquisa

Esta pesquisa se deu a partir de uma abordagem holística, metódica e explorativa. **Holística**, por conta do seu caráter interdisciplinar, investigando o processo criativo em meio digital a partir dos seus aspectos técnicos, teóricos e expressivos de maneira integrada, sem a ênfase de um sobre os demais. **Metódica**, pela postura objetiva e sistemática na categorização e identificação das estruturas e mecanismos envolvidos nestas práticas. **Explorativa**, porque, embora o objeto e o objetivo desta pesquisa estivessem relativamente bem definidos no início do processo, os percursos específicos, descobertas e reflexões surgiram e se definiram de maneira gradual e dinâmica ao longo dos estudos teóricos, e principalmente dos práticos.

As formas de investigação empregadas incluíram a leitura, fichamento, análise e reflexão sobre **referências teóricas** (livros, artigos e outras publicações) e **práticas** (produções artísticas, jogos, experimentos, aplicativos e ferramentas), **diálogos** com pesquisadores, artistas, criativos e desenvolvedores, **produção própria** (criativa, técnica e experimental) e, finalmente, os **estudos práticos** realizados durante o período sanduíche desta pesquisa.[4]

[3] No final da tese são apresentadas também as referências da pesquisa (p. 259), bem como um glossário dos principais termos empregados (p. 273) e uma lista de figuras (p. 275).

[4] Minha produção criativa contribuiu como uma forma de investigação ativa dos temas da pesquisa. Participei com produções generativas no evento Procedural Generation Jam e no FILE (Festival Internacional de Linguagem Eletrônica), ambos em 2015, sendo duas obras em cada. Colaborei com a tradução das legendas de Hello World! Processing (ALAEJOS, 2013), um documentário sobre programação criativa, junto com Marcelo Padovani Macieira e Artur Cabral Reis (Processing SP). Também colaborei com o coletivo artístico Grupo Poéticas Digitais (ECA/USP). Durante todo o período deste estudo publiquei periodicamente no blog desta pesquisa textos, bem como a minha produção artística, criativa e experimental, relacionados aos temas investigados - URL: `www.7luas.com.br/pesquisa`

2 Expressividade Procedural

No contexto desta pesquisa a **expressividade procedural** refere-se às manifestações artísticas ou expressivas em mídias digitais baseadas fundamentalmente no aspecto procedural deste meio - isto é, na sua programabilidade e autonomia (independentemente do suporte de exibição do produto final). Nesta situação, o artista ou criativo (ou um colaborador sob sua orientação) realiza um ato de metacriação, idealizando, produzindo ou manipulando algoritmos (ou instruções) a serem executadas pelo computador - isto é, a partir da autoria procedural.

Este emprego das mídias digitais difere fundamentalmente das práticas nas quais o computador é empregado unicamente para emular formas expressivas tradicionais, de natureza estática ou linear. Por exemplo, é o caso do autor que utiliza um processador de texto para produzir Literatura, o desenhista que emprega um aplicativo gráfico para criar a partir de estratégias expressivas tradicionais das Artes Visuais ou um músico que grava e edita uma composição tradicional em um software de edição de áudio. Embora a princípio estes sejam usos legítimos das mídias digitais, eles não são o foco da nossa pesquisa.[5]

2.1 Apresentação

O conceito de **expressividade** empregado nesta pesquisa se refere principalmente à definição de Janet Murray sobre a forma expressiva, em oposição à aditiva.

Segundo Murray, o uso expressivo de um meio se dá a partir do emprego dos mecanismos que lhe são próprios e únicos - no caso do meio digital, este seria o aspecto procedural, como veremos adiante. Este uso se opõe à forma aditiva, que se refere ao uso de um meio ou linguagem a partir das estratégias de outras formas expressivas (MURRAY, 2003).

Por exemplo, o Cinema inicialmente era uma forma aditiva, baseada na combinação de estratégias expressivas de outros meios, principalmente Fotografia e Teatro (como a montagem dos cenários, posicionamento de atores na cena, e conceitos de composição e enquadramento). Depois de um período de exploração formal, a linguagem do Cinema eventualmente amadureceu e se consolidou como uma forma expressiva própria.

[5]Empregamos o termo "estático" como sinônimo de fixo ou predefinido - qualidades inerentes às formas expressivas tradicionais. Um filme, por exemplo, apesar de ser capaz de representar movimento, é um conteúdo estático pois é idêntico a cada nova exibição (nesta pesquisa não tratamos sobre a questão da subjetividade de interpretação do receptor). Utilizamos a grafia com a primeira letra em maiúsculo quando nos referimos ao nome das formas expressivas tradicionais (por exemplo, "Música"), diferenciando de outros usos (por exemplo, ao tratar de uma "música" específica, ou de um conteúdo de natureza "musical", expressivo ou não).

Embora esta pesquisa se dê no contexto das artes, o conceito de expressividade empregado aqui não se limita às práticas artísticas ou poéticas - se refere também à noção mais ampla de comunicar uma determinada ideia ou significado. Também convém notar que consideramos a expressividade unicamente em referência ao potencial oferecido por um determinado meio ou linguagem, e no seu emprego por um artista ou criativo - em outras palavras, a expressividade reside na mente criativa, e não na materialidade do meio empregado ou da obra produzida (no caso da autoria procedural, nos algoritmos ou na máquina na qual eles residem).[6]

Além disso, apesar de parte da nossa investigação se dar no recorte das narrativas e jogos digitais, isso é feito principalmente de forma a acessar as estratégias procedurais de criação de uma maneira mais ampla e fundamental, independentemente das suas aplicações específicas.

O aspecto que distingue as mídias digitais dos demais meios e linguagens, a partir do qual a sua expressividade própria pode se realizar, seria o seu aspecto **procedural**. Segundo Murray, este é "o elemento mais importante que [este meio] adiciona ao nosso repertório de poderes de representação" (MURRAY, 2003:274).[7]

Outros autores compartilham desta opinião, como o cientista da computação Alan Kay, que há décadas tem se dedicado à questão da literacia procedural (KAY, 2015), o artista Harold Cohen, que afirma que "para o artista, a essência do computador é a sua autonomia" (COHEN, 1986:194) e o pesquisador Michael Mateas, que atua na área da inteligência artificial e narrativa interativa. Mateas explica que a característica principal do computador como meio de representação está justamente na sua capacidade de "computar", isto é, de realizar cálculos e executar procedimentos - todos os demais aspectos das mídias digitais seriam derivados deste (MATEAS; STERN, 2005; MATEAS, 2003).[8]

[6] A "carga expressiva" de uma determinada obra, por exemplo, refere-se a um potencial, associado à intenção do seu criador. Mesmo quando nos referimos ao "artista", nos interessa principalmente o elemento da intenção expressiva.

[7] Janet Murray descreve os quatro aspectos que definem o meio digital como: (1) **participativo**, que se refere à possibilidade de interação com o usuário, (2) **espacial**, referente ao seu caráter "navegável", (3) **enciclopédico**, pela capacidade praticamente ilimitada de armazenagem e acesso à informação e (4) **procedural**, que é a capacidade de executar instruções de maneira autônoma. Citação: tradução nossa - texto original: "[t]he most important element the new medium adds to our repertoire of representational powers" (MURRAY, 2003:274).

[8] Segundo Alan Kay, há três tipos de literacia: a de leitura, criação e gênero - a literacia procedural plena viria da combinação destes três tipos. Kay utiliza o termo "alfabetização computacional", ao invés de "literacia procedural", embora o sentido seja o mesmo (KAY, 2015). Para uma argumentação mais aprofundada sobre a primazia do aspecto procedural nas mídias digitais, refira à nossa pesquisa anterior (FERREIRA, 2011). Citação: tradução nossa - texto original: "For the artist, the essence of the computer is its autonomy." (COHEN, 1986:194)

Alan Kay descreve o computador como sendo uma **metamídia**, por conta da sua capacidade de emular (ou simular) outros meios e linguagens (KAY, 1984) - isso inclui tanto formas tradicionais, quanto linguagens completamente originais, que nem ao menos poderiam existir fora do meio digital.

Entretanto, é importante notar que, embora toda atividade e operação realizada em um computador dependa da proceduralidade do meio digital, o conteúdo sendo acessado, criado ou manipulado pode ser tanto estático quanto procedural.

Quando um escritor utiliza um processador de texto, embora esta atividade esteja sendo realizada em um computador, e a execução do aplicativo dependa do aspecto procedural deste meio, o conteúdo sendo criado é estático (o texto) e as estratégias expressivas sendo empregadas pelo autor são essencialmente de natureza tradicional (da Literatura) - isto é, neste caso, o computador é empregado como uma forma aditiva.

Nesta pesquisa, nos interessa unicamente o uso do computador como forma expressiva, que se dá a partir do emprego das estratégias procedurais de criação, ou autorial procedural - isto é, quando o artista ou criativo cria ou manipula algoritmos (seja diretamente, através da produção de código, ou de maneira indireta, utilizando ferramentas ou interfaces visuais).

Além disso, embora isso seja evidente, reiteramos que a expressividade procedural não se refere ao emprego do computador para fins práticos - isto é, sem uma intenção artística ou expressiva -, seja lidando com conteúdo estático (texto, imagens, planilhas e assim por diante) ou de natureza procedural (como aplicativos utilitários, ferramentas e bibliotecas especializadas).

Recorte

A tabela a seguir (fig. 1) ilustra o recorte descrito nas páginas anteriores, localizando o nosso objeto de estudo entre as diferentes atividades de criação em mídias digitais. As fileiras representam a **intenção** de quem cria - para fins práticos ou para fins expressivos -, e as colunas representam a **natureza do conteúdo** sendo criado ou manipulado - estática ou procedural.[9]

O primeiro quadrante da tabela (topo esquerdo) representa o emprego do meio digital para a criação de conteúdo de natureza **estática** e para fins **práticos** - por exemplo, escrever e-mails e textos, criar e editar imagens ou áudio, produzir planilhas e assim por diante.[10]

		NATUREZA do CONTEÚDO	
		ESTÁTICA	PROCEDURAL
INTENÇÃO	PRÁTICA	texto, imagem, vídeo, áudio, números, planilhas etc.	aplicativos utilitários, entretenimento (jogos), scripts, macros etc.
	EXPRESSIVA	Literatura, Música, Cinema, Artes Visuais etc.	**Expressividade Procedural**

Figura 1 – Tabela: Atividades criativas em mídias digitais

[9]Na tese ocasionalmente nos referimos ao conceito da criatividade de uma maneira mais ampla, ou tratamos de práticas ou produções criativas que não são artísticas ou expressivas. Isso é feito nos casos em que estas se referem a estratégias procedurais fundamentais, compartilhadas com o emprego do computador para fins expressivos.

[10]Este quadrante **não** contém produções expressivas. Por exemplo, uma performance artística baseada no envio manual de e-mails não estaria aqui - ela estaria na segunda fileira (mais especificamente, no terceiro quadrante - esquerda inferior - uma vez que sua forma expressiva predominante seria de natureza essencialmente tradicional, a Arte Postal).

O segundo quadrante (topo direito) representa a criação de conteúdo **procedural** com uma intenção **prática** - como o desenvolvimento de aplicativos utilitários e outras ferramentas e componentes relacionados, e mesmo a criação de scripts ou macros simples (por exemplo, a criação de regras em um filtro de e-mail pode ser considerado como uma forma bastante simples de programação).[11]

O terceiro quadrante (inferior esquerdo) - criação de conteúdo **estático** com intenção **expressiva** - refere-se ao emprego do meio digital como uma forma aditiva, isto é, a partir de estratégias expressivas de outras formas tradicionais. Isso inclui escrever uma poesia, gravar e editar uma música ou criar uma ilustração - considerando estas atividades sendo realizadas com fins expressivos. Este é o uso mais comum do computador entre os artistas e criativos utilizando este meio, quantitativamente falando.[12]

Finalmente, o quarto e último quadrante (inferior direito) refere-se ao nosso objeto de estudo: a criação **procedural** com fins **expressivos** - isto é, a expressividade procedural.

As quatro categorias definidas pelos quadrantes nesta tabela não são rígidas, principalmente considerando as manifestações e aplicações práticas em cada caso. Assim, um mesmo artista ou criativo pode atuar a partir de diferentes abordagens, e lidando com tipos de conteúdo distintos, inclusive simultaneamente.

Por exemplo, a rigor os jogos digitais estão localizados no segundo quadrante (superior direito), por tratarem-se de produções predominantemente práticas (com fins de entretenimento). Porém, no caso dos gêneros e títulos que apresentam aspectos e elementos expressivos procedurais, estes se localizariam em uma posição intermediária entre os dois quadrantes da direita.

Estes conceitos e exemplos serão retomados e aprofundados mais adiante na tese, à medida em que elaboramos o nosso objeto de investigação.

[11] Mesmo no caso de um aplicativo que pode ser empregado para fins artísticos (por exemplo, um editor de vídeo), a princípio a intenção de quem o programou não é expressiva, mas sim prática. Note que este quadrante **não** inclui produções procedurais predominantemente expressivas, como é o caso da Arte de Software - por exemplo o Auto-Illustrator (Adrian Ward, 2002) é uma paródia dos aplicativos gráficos, que subverte as suas ferramentas e funções de maneiras inesperadas e bem-humoradas (este tipo de produção pertence ao quadrante inferior direito - expressividade procedural). De um maneira simplificada, scripts e macros são tipos de algoritmos especializados.

[12] Este quadrante da tabela se refere unicamente às práticas tradicionais destas linguagens, empregando unicamente estratégias expressivas tradicionais. Assim, não faz parte dele a Arte Generativa, por exemplo, que consiste em sistemas que geram produções visuais, sonoras ou mesmo verbais e narrativas - embora estas práticas possam produzir resultados esteticamente similares a aqueles de manifestações artísticas tradicionais similares, o processo criativo é fundamentalmente distinto.

Manifestações e Aplicações

Embora as estratégias procedurais de criação sejam sempre realizadas através das mídias digitais, a sua **materialização** não precisa necessariamente acontecer neste suporte. Ela também pode se dar em outros meios, geralmente associados a formas expressivas tradicionais, como o impresso, a escultura e a performance.

No próprio suporte digital, a materialização pode se dar de diferentes formas: através de um aplicativo sendo executado no computador ou de maneira estática, exibido visualmente na tela ou através de ondas sonoras (através das caixas de som), e assim por diante.

A autoria procedural também pode acontecer em linguagens de diferentes **tipos** ou **naturezas**, como visual, verbal, narrativa, musical e cinética.

Além disso, esta abordagem pode acontecer em diferentes **etapas** do processo criativo, como mostra o diagrama abaixo (fig. 2). A obra final pode ser gerada por um algoritmo (metacriação); o ato da execução do algoritmo pode ser apresentado como exposição; o processo como um todo pode ser uma performance; e assim por diante.[13]

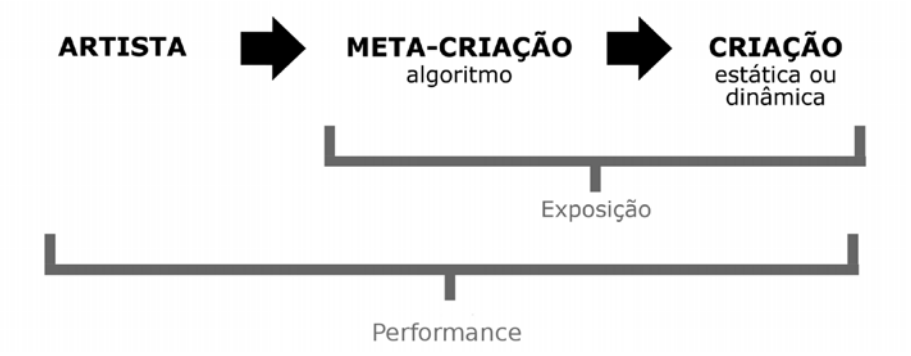


Figura 2 – Diagrama: Autoria procedural: etapas do processo criativo

[13]Na autoria procedural, o produto direto do ato criativo é o algoritmo - este, por sua vez, pode gerar ou executar a obra final. Nesta tese, nos referimos a este produto de diferentes formas, através de termos como "código", "função", "sistema" e "simulação", de acordo com o contexto ou perspectiva, ou de forma a evidenciar algum aspecto particular. Mais adiante trataremos novamente sobre as diferentes etapas no processo criativo nas quais a autoria procedural pode estar presente (p. 84).

Embora nesta tese a maior parte dos exemplos e estudos práticos se encontrem na área das narrativas e jogos digitais, as estratégias procedurais de criação podem ser aplicadas em uma variedade de outras áreas, suportes e linguagens.

No **Cinema**, um exemplo do emprego desta abordagem é o Automavision, um sistema procedural para composição de cenas, empregado no filme The Boss of It All (Lars Von Trier, 2006). Outro exemplo é CLOUDS (James George e Jonathan Minard, 2014), um documentário digital, no qual a edição acontece de maneira dinâmica de acordo com os tópicos e preferências do receptor. Na **Música**, podemos citar o Verbasizer (Ty Robert, 1995), um sistema desenvolvido para auxiliar o músico David Bowie na criação de temáticas e letras (APTED, 2011), e o aplicativo Biophilia (Björk, 2011), no qual músicas do álbum homônimo são apresentadas na forma de sistemas interativos. Na **Dança**, a companhia britânica Random Dance utiliza um sistema chamado Agente de Linguagem Coreográfica (em Inglês, "Choreographic Language Agent"; OpenEnded, 2010) para auxiliar na criação de coreografias. No **jornalismo**, podemos citar o jogo Cutthroat Capitalism (WIRED, 2009) e o projeto experimental Game-o-Matic (Expressive Intelligence Studio/UC Santa Cruz), cujo objetivo é servir como uma ferramenta para democratizar a expressão e a comunicação de ideias a partir de jogos e outros tipos de sistemas e produções digitais (WARDRIP-FRUIN, 2014).[14]

Como veremos, ao longo desta tese serão feitas conexões com outras práticas e aplicações da autoria procedural, como estas citadas acima, uma vez que todas compartilham de grande parte das mesmas estratégias e mecanismos criativos e expressivos. Isto é, os mesmos tipos de algoritmos utilizados para descrever o comportamento de um personagem em uma narrativa interativa também podem ser aplicados na criação de uma coreografia de dança ou mesmo para a geração de uma escultura.

Além disso, há inúmeras intersecções entre os diferentes tipos de produções baseadas em estratégias procedurais de criação. Por exemplo, da mesma forma que uma obra de arte pode apresentar mecanismos de interação similares a aqueles encontrados em jogos, os jogos por sua vez geralmente possuem em alguma medida elementos e aspectos artísticos ou expressivos.[15]

[14] Para mais exemplos relacionados à expressividade procedural de uma maneira mais ampla, refira à nossa pesquisa anterior (FERREIRA, 2011). Há uma imagem de Cutthroat Capitalism mais adiante na tese (fig. 13; p. 72).

[15] No contexto desta pesquisa, jogos digitais não são considerados necessariamente manifestações artísticas, uma vez que seu propósito principal é o entretenimento e o desafio, embora possam apresentar aspectos e elementos expressivos (o mesmo vale para certos tipos de narrativas digitais, nas quais predomina a jogabilidade). Os chamados Art Games (ou "jogos artísticos"), por outro lado, tratam-se de produções predominantemente artísticas ou expressivas.

Breve História da Autoria Procedural

As práticas artísticas e expressivas sempre estiveram de alguma maneira associadas à noção de formalização e objetividade. A abordagem sistemática à criação já era evidente nas primeiras obras de Arte Permutacional, no **século 15**, que envolviam a definição de um universo de elementos e as regras de combinação entre eles, que por sua vez serviam de base para a produção das permutações. Mais tarde, com a invenção das máquinas digitais, de certa forma surgia também o meio ideal para este tipo de abordagem, uma vez que o computador permitia automatizar a execução de instruções com precisão (MOLES, 1990).

Também podem ser identificados paralelos entre as estratégias procedurais de criação e as práticas artísticas conceituais do final dos **anos 50**, mais especificamente a Arte Declarativa, na qual a "ideia se torna uma máquina que faz a arte." (LEWITT, 1996:822)[16]

Nos **anos 60**, artistas como Frieder Nake, Vera Molnar e Manfred Mohr passam a empregar o computador no seu processo criativo, como uma continuidade natural do seu método de criação tradicional sistemático, quase algorítmico (elaboraremos sobre isso mais adiante). Em 1968 acontece a exposição pioneira de "arte cibernética" (ou arte tecnológica) Cybernetic Serendipity, em Londres, com curadoria de Jasia Reichardt.

No Brasil, o artista Waldemar Cordeiro produziu, juntamente com o físico Giorgio Moscati, as primeiras obras de Arte Computacional no país, no final dos anos 60. A partir desta década, começam a surgir mais projetos buscando tornar os mecanismos procedurais do meio numérico mais acessíveis para o usuário, como o sistema Sketchpad (Ivan Sutherland, 1963), a linguagem de programação Logo (Wally Feurzeig e Seymour Papert, 1967) e o dispositivo experimental Dynabook (Alan Kay, 1977).

Nos **anos 70**, mais artistas e criativos começam a ter acesso a esta tecnologia, com a chegada dos computadores pessoais. No final desta década surgem as *demos,* uma nova manifestação criativa em meio digital associada aos hackers e à pirataria de software. Em 1984 Michel Bret desenvolve uma linguagem de programação de alto nível chamada Anyflo, voltada para a construção de ferramentas de criação em meio digital.

A popularidade da programação de computadores cresce no final dos **anos 90**, com o surgimento de projetos como os ambientes de autoria Pure Data (Miller Puckette, 1996) e o Flash (Macromedia, 1996), linguagens como a Processing (Ben Fry e Casey Reas, 2001) e aplicativos voltados especificamente para a autoria de jogos, como o Adventure Game Studio (Chris Jones, 1997) e o Construct

[16]Citação: tradução nossa - texto original: "The idea becomes a machine that makes the art."

(Scirra, 2007). De certo forma, dava-se início a uma verdadeira revolução no emprego criativo e expressivo da tecnologia, e que segue até hoje.[17]

Nos últimos anos, tem havido um crescimento particularmente significativo na produção de jogos independentes, e de designers interessados em explorar as suas mecânicas e sistemas de maneira expressiva. É o caso de jogos como Dys4ia (Anna Anthropy, 2012), que trata sobre questões de gênero, Papers, Please (Lucas Pope, 2013), abordando o tema da imigração, The Beginner's Guide (Davey Wreden, 2015), um "metajogo" sobre um designer de jogos (e o seu processo criativo) e Sunset (Tale of Tales, 2015), que examina relações de poder e outros temas.

Muitos destes jogos independentes, alguns dos quais serão analisados ao longo desta tese, apresentam um nível de qualidade comparável a aqueles de grandes produtoras. Porém, há um volume ainda maior de jogos sendo produzidos por designers com menos experiência, e mesmo por amadores. Isso se deve em grande parte à crescente democratização de ferramentas de autoria como GameMaker (YoYo Games, 1999-), capaz de criar jogos em uma variedade de gêneros e estilos, Unity (2005-), particularmente robusto para projetos 3D, Twine (Chris Klimas, 2009-), para produções baseadas principalmente em texto e hyperlinks, e inúmeras outras.

No ano passado, uma produtora brasileira lançou o Adventurezator (Pigasus, 2015), um aplicativo para criação de adventures similares aos jogos clássicos no gênero, porém incluindo ferramentas que permitem a autoria de regras e comportamentos complexos para os personagens e outros elementos da simulação.[18]

Outra área - ou tecnologia - que tem ganhado cada vez mais evidência é a geração procedural de conteúdo (ou "procedural generation", também abreviada como "procgen"), que consiste no emprego radical da proceduralidade em detrimento da criação estática (mesma abordagem adotada pelos criadores de demos nos anos 70). Esta tecnologia tem sido empregada por artistas (principalmente para a criação de músicas e produções visuais), bem como por outros criativos, em áreas como na arquitetura, tipografia e moda, bem como em jogos.

[17] Convém mencionar que já existiam aplicativos e linguagens especificamente voltadas para a criação de jogos antes disso, um exemplo notável sendo Garry Kitchen's GameMaker (Activision, 1985). Entretanto, ainda levariam alguns anos até que a democratização da criação de jogos se tornasse verdadeiramente significativa e abrangente.

[18] Também tem havido um crescimento significativo nas produções para web e plataformas móveis, em parte por conta da adoção cada vez maior do HTML5, desde o final de 2014 (o Flash tem perdido popularidade, principalmente por questões de performance, estabilidade e segurança). O Adventurezator, além de servir como uma ferramenta de autoria, também vem acompanhado de um jogo completo.

Um dos jogos que exploram esta abordagem de maneira mais evidente é a aventura Dwarf Fortress (Bay 12 Games, 2006-), na qual o "mundo" virtual é gerado proceduralmente, incluindo as cidades e cada um dos seus habitantes, bem como a história descrevendo os eventos do passado de todos os personagens.

Outro título que deve empregar a geração procedural de conteúdo de maneira radical é No Man's Sky (Hello Games), atualmente em desenvolvimento. Trata-se de um jogo de exploração espacial no qual, segundo seus criadores, galáxias inteiras serão gerados unicamente a partir de algoritmos, em tempo real, incluindo cada um dos planetas, sua fauna e flora.[19]

Convém notar que, no caso específico da geração procedural de conteúdo, há uma ênfase maior nos aspectos técnicos e práticos da programação, em oposição ao seu emprego expressivo ou artístico. De qualquer forma, estas práticas refletem de maneira evidente o potencial da autoria procedural, servindo de inspiração e referência para outros artistas e criativos na área.

[19] Dwarf Fortress é um jogo relativamente obscuro, em parte devido à complexidade do seu sistema e a simplicidade dos seus gráficos. No Man's Sky, cujo lançamento está previsto para Junho deste ano de 2016, deve apresentar uma jogabilidade mais tradicional e acessível (o jogo é citado de novo mais adiante, incluindo uma imagem; p. 176).

2.2 Referências Teóricas

A seguir, aprofundaremos sobre o tema da expressividade procedural a partir da investigação de quatro referências teóricas específicas.

Examinaremos (1) o grupo dos Algoristas, artistas pioneiros que fizeram a transição das práticas tradicionais para a Arte Computacional (p. 37); (2) a aplicação do conceito de Inteligência Artificial Expressiva na criação do "drama interativo" Façade (Michael Mateas e Andrew Stern, 2005), incluindo uma comparação entre esta abordagem e os engines de física (p. 43); (3) o conceito de processamento expressivo a partir dos estudos de Noah Wardrip-Fruin na área da literatura digital (p. 57); e (4) a teoria de unidades operacionais, do pesquisador e designer de jogos Ian Bogost (p. 62).

Como veremos, cada uma destas referências nos permitirá examinar diferentes aspectos e elementos da autoria e da expressividade procedural. Além disso, ao realizar uma leitura crítica destas referências, contrapondo a elas as nossas próprias elaborações teóricas, é possível identificar perspectivas distintas e únicas sobre o nosso objeto de estudo.

2.2.1 Os Algoristas (Jean Pierre Hebert, Roman Verostko e outros)

Examinaremos a seguir o emprego expressivo da programação de computadores em uma das suas manifestações mais próximas (ou similares) às chamadas artes tradicionais. Ao investigar o processo criativo de artistas que fizeram a transição dos meios tradicionais para o uso expressivo do código, e refletir sobre as suas motivações e intenções, nos perguntamos: a expressividade procedural é uma nova manifestação artística, ou unicamente uma abordagem alternativa a práticas já existentes?

O termo Algorista foi proposto por Jean Pierre Hebert em 1995 para definir o artista que cria a partir de algoritmos, e cujo trabalho inclui código de autoria própria (HÉRBERT, 2015). Herbert descreveu o manifesto do grupo na forma do seguinte pseudocódigo:[20]

The Algorist Manifesto

```
if (creation && object of art && algorithm && one's own algorithm) {
        include * an algorist *
} elseif (!creation || !object of art || !algorithm || !one's own algorithm) {
        exclude * not an algorist *
}
```

A interpretação da primeira parte deste pseudocódigo em linguagem verbal seria algo como: "Se uma pessoa cria um objeto de arte utilizando algoritmos, incluindo código de sua autoria, ela pode ser considerada um algorista."[21]

Entre os artistas pioneiros desta abordagem estão Herbert Frank, Manfred Mohr, Frieder Nake, Georg Nees, Vera Molnar e Edward Zajec. Outros nomes importantes no movimento são Mark Wilson, Roman Verostko, Charles Csuri e Harold Cohen.

Cohen começou o seu percurso acadêmico e artístico nas belas artes, atuando principalmente como pintor. No final da década de 60, teve o seu primeiro contato com a computação, posteriormente se aprofundando na área da Inteligência Artificial, no início dos anos 70. Em 1973, Cohen iniciou o desenvolvimento de Aaron, um sistema computacional para geração de imagens originais, que continua ativo até hoje.

[20] Pseudocódigo obtido no site de Roman Verostko (sem data; sem numeração de página).
URL: `www.verostko.com/algorist.html`

[21] A estrutura deste um pseudocódigo é incomum, pois a segunda parte, a partir do "elseif", a princípio não seria necessária. A rigor ela apenas repete o que já foi expressado na primeira parte.

Para Roman Verostko, a rigor a arte dos algoristas - ou "Arte Algorítmica" - não se restringe aos computadores. Ela abrange desde os padrões encontrados nas "cestas trançadas" pré-históricas até a arte geométrica e conceitual do século XX.

> Arte conceitual pode ser vista como uma forma de arte algorítmica, porém somos sábios em manter nosso foco histórico na natureza e contexto da arte "conceitual". Considerar relacionamentos é valioso. Porém também precisamos identificar qualidades e interesses específicos apropriados para a época analisada.
>
> A meu ver a arte algorítmica pode ser encontrada ao longo da história da "criação artística", incluindo a arte pré-histórica. Contudo, em cada instancia ela será tingida pelo seu contexto histórico. (informação pessoal)[22]

Isto é, segundo o autor, o fato da Arte Algorítmica se aplicar tanto aos processos computacionais quanto à Arte Conceitual, e mesmo ao artesanato na pré-história, não tira a legitimidade da classificação.

> Para o uso do termo "algorista" aplicado a artistas, precisamos olhar para o contexto histórico, [considerando] o emprego [de fato] e o emprego pretendido. O termo genérico "algorista" se refere a qualquer pessoa que trabalha com algoritmos para qualquer propósito. Seu uso era presente séculos antes do meu nascimento. Então a nossa adaptação no final do século XX para um tipo específico de artista não muda a história que nos precede.
>
> Adaptamos o termo "algorista" para identificar artistas do final do século XX que criaram sua arte utilizando computadores para implementar seus algoritmos. Isso não muda o significado fundamental mais amplo do termo. Historiadores e curadores irão adaptar ou rejeitar o uso com o passar do tempo. Já existe literatura suficiente para sugerir que trata-se de uma adaptação útil. (Idem)[23]

[22] Correspondência eletrônica pessoal com o autor em Novembro de 2013 (citada com autorização; sem numeração de página). Citação: tradução nossa - texto original: "Conceptual art can be viewed as a form of algorithmic art yet we are wise to retain our historic focus on the nature and context of 'conceptual' art. Expounding on relationships is valuable. Yet we also need to identify specific qualities and interests proper to the time under review. From my view algorithmic art can be found throughout the history of 'art-making' including prehistoric art. Yet, in every instance it will be colored with its historic context."

[23] Citação: tradução nossa - texto original: "For the use of the term 'algorist' applied to artists, we need to look at historical context, usage, and intended usage. The general term 'algorist' refers to anyone who works with algorithms for any purpose whatsoever. Usage was present centuries before I was born. So our adaptation in the late 20th Century for a specific kind of artist doesn't change the history that preceded us. We adapted the term 'algorist' to identify artists of the late 20th century who created their art by using computers to implement their algorithms. This does not change the fundamental broad meaning of the term. Historians and curators will either adapt or reject the usage as time goes on. There is already enough literature to suggest that it is a useful adaptation."

Assim, o termo "algorista", tal qual ele é utilizado por estes artistas, não tem como intenção redefinir o significado histórico deste conceito, mas sim de servir como uma maneira de distinguir entre as duas formas pelas quais é possível criar utilizando o computador: entre os artistas que escrevem eles próprios os algoritmos, e aqueles que utilizam algoritmos ou programas já existentes, como por exemplo um desenhista que cria através de um aplicativo gráfico como o Photoshop (Adobe).

> Como historiador, eu queria um termo para identificar tanto a mim mesmo como este tipo de artista, quanto todos os outros que seguiam tal prática. O eventual acordo em adaptar o termo "algorista" não foi com a intenção, a meu ver, de reescrever o seu uso na história da arte antes do advento do computador.
>
> A intenção foi de distinguir entre "artistas computacionais" que escreviam seu próprio código original, e aqueles que utilizavam programas comerciais, que estavam se tornando lugar comum nos anos 90. [...]
>
> Eu sempre vi minha arte como arte algorítmica. Arte algorítmica é um termo apropriado para a arte que remete a monumentos como o Stonehenge e a fabricação de cestos na pré-história, tapeçaria etc. (Idem)[24]

Do Automatismo à Expressão

A maioria dos artistas pioneiros no uso de algoritmos no processo criativo atuavam em linguagens artísticas tradicionais antes de adotar o computador como ferramenta de trabalho. É o caso de Vera Molnar, nascida em 1924, cuja carreira artística começou com a produção de desenhos e pinturas, aos doze anos de idade.

Com o tempo, a artista passou a se interessar por formas geométricas, adotando uma abordagem mais sistemática às suas produções. Eventualmente, passou a utilizar o computador no seu trabalho, em 1968.[25]

[24] Citação: tradução nossa - texto original: "As a historian I wanted a term to identify both myself as such an artist and all the others who followed such a practice. The eventual agreement to adapt the term 'algorist' was not meant, in my mind, to rewrite historical usage in the history of art before the advent of the computer. It was intended to distinguish between 'computer artists' who wrote their own original code and those who used commercial programs that were becoming common place in the 1990's. [...] I have always viewed my art as algorithmic art. Algorithmic art is an appropriate term for art that reaches back to monuments like Stonehenge, and prehistoric basket weaving, rug making etc."

[25] Informações obtidas no perfil da artista no Digital Art Museum (DAM).
URL: `www.dam.org/dox/2456.6Pkud.H.1.De.php`

Figura 3 – Autorretrato (Vera Molnar, 1939)

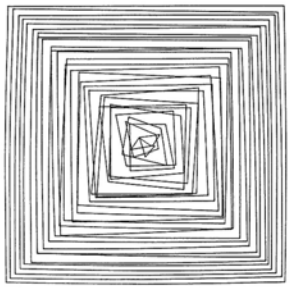


Figura 4 – Variação na série Hypertransformations (1954)

Figura 5 – Structure de Quadrilatères (1988)

As figuras acima representam obras de diferentes momentos da carreira de Molnar – um autorretrato de 1939 (fig. 3), uma produção computacional de 1954 (fig. 4) e outra de 1988 (fig. 5).[26]

A artista declara que "[s]em o auxílio de um computador, não seria possível materializar de maneira tão fiel uma imagem que anteriormente existia apenas na mente do artista." (MOLNAR, 1998) Por um lado, esta afirmação parece sugerir o uso do computador unicamente como uma maneira de automatizar ou otimizar um processo que já existia previamente na mente do artista, de natureza tradicional e estática.[27]

Manfred Mohr, ao escrever sobre o seu próprio trabalho, transmite uma ideia similar: "Através da máquina [...] eu me torno uma super-versão de mim mesmo. [Com o computador eu consigo alcançar] lugares que eu não conseguiria entrar por estar bloqueando o meu próprio caminho." (FOWLER, 2016)[28]

Isto é, de um lado, temos o computador unicamente como uma "super-versão" de um artista, realizando ideias que o ser humano é limitado demais para realizar. Do outro, temos as mídias digitais como uma forma expressiva própria, distinta daquelas das demais linguagens tradicionais (mesmo das mais sistemáticas).

[26]Imagens obtidas no site da artista. URL: `www.veramolnar.com`

[27]No trabalho de Molnar e de outros artistas com uma abordagem similar, as imagens geradas pelos algoritmos ainda passam por uma etapa de análise e seleção, realizada pessoalmente pela própria artista, antes da finalização da obra propriamente dita. Isto é, o processo envolve, além do elemento computacional, a sensibilidade da artista no momento da seleção. Citação: tradução nossa - texto original: "Without the aid of a computer, it would not [be] possible to materialize quite so faithfully an image that previously existed only in the artist's mind." (documento digital, sem numeração de página)

[28]Citação: tradução nossa - texto original: "Through the machine [...] I become the super-version of myself. [With the computer I can reach] places that I couldn't go into because I'm standing in my way." (documento digital, sem numeração de página; na tradução em Português foi introduzida a palavra "próprio" para evidenciar que Mohr referia-se a si mesmo na citação original)

Entretanto, a diferença entre estas duas situações não é tão clara. Além disso, pode-se argumentar que ela é arbitrária pois, a partir de um certo ponto, a diferença unicamente **quantitativa** (na capacidade de processar com rapidez um grande volume de informação e cálculos complexos) se torna uma diferença **qualitativa**.

Também devemos considerar o caráter subjetivo na análise da carga expressiva ou poética de uma determinada obra ou processo. De um lado, temos o emprego de algoritmos unicamente com um propósito prático, e do outro, como uma maneira de explorar o meio digital como uma forma expressiva própria. Determinar em que ponto nesta escala se localiza uma determinada obra, ou a abordagem de um artista ou criativo, por exemplo, não é trivial.

De qualquer maneira, como observa Grant David Taylor, curador da exposição The American Algorists: Linear Sublime (2013), entre as produções dos algoristas, e no discurso dos seus artistas, há uma forte ênfase na ideia do emprego do computador como uma maneira de automatizar o processo criativo (TAYLOR, 2013).

Segundo Molnar: "Esta máquina, por mais impressionante que seja, nada mais é do que uma mera ferramenta na mão do pintor. [...] O computador ajuda, mas ele não 'faz', nem 'projeta' ou 'inventa' nada." (MOLNAR, 1990)[29]

Sob esta perspectiva, a citação a seguir pode ser interpretada de duas maneiras distintas, de acordo com o que se considera como sendo o objeto da realização mencionada por Molnar: "Pode soar paradoxal, mas a máquina, que é vista como fria e desumana, pode ajudar a realizar o que há de mais subjetivo, inatingível e profundo em um ser humano." (Idem, 1998)[30]

No contexto desta pesquisa, esta fala é interpretada como sendo em referência a uma intenção expressiva na mente do artista. Outra interpretação possível é considerar que o objeto referenciado na citação é uma obra em particular, ou uma proposta poética específica. Neste caso, a máquina estaria servindo um propósito mais prático, de realizar determinados gestos que permitiriam a materialização da obra.

[29] Citação: tradução nossa - texto original: "This machine, as impressive as it may be, is after all merely a tool in the hand of the painter. [...] The computer helps, but it does not 'do', does not 'design' or 'invent' anything." (documento digital, sem numeração de página)

[30] Citação: tradução nossa - texto original: "This may sound paradoxical, but the machine, which is thought to be cold and inhuman, can help to realize what is most subjective, unattainable, and profound in a human being." (sem numeração de página)

Manifestação Artística ou Abordagem

Parte importante do processo desta pesquisa consiste na identificação das práticas e estratégias associadas à expressividade procedural, e como elas se localizam em relação às demais formas artísticas. Uma questão presente desde o início deste estudo era a compreensão da própria natureza destas práticas. A princípio, já era possível definir com certa clareza como e onde elas se localizavam no contexto da arte digital, bem como das artes em geral. Porém, o conceito da expressividade procedural teve que ser constantemente aprofundado e atualizado ao longo da pesquisa, de forma a dar conta das elaborações teóricas e estudos práticos sendo realizados.

O processo envolvido neste amadurecimento conceitual pode ser condensado na seguinte pergunta: a expressividade procedural deve ser considerada como (1) uma **manifestação artística** ou expressiva em si; ou (2) como uma **abordagem criativa**, que pode ser adotada por - ou aplicada a - diferentes formas expressivas, inclusive as tradicionais?

Esta questão também se encontra intimamente relacionada à discussão sobre a conexão entre a expressividade procedural e a Arte Conceitual, também presente deste as etapas iniciais desta pesquisa, e mais recentemente no diálogo com Roman Verostko, apresentado anteriormente. Isto é, considerando que as estratégias procedurais de criação são definidas pela sua natureza conceitual, como elas se distinguem das práticas artísticas conceituais tradicionais, como a Arte Declarativa, ou a própria Arte Conceitual de uma maneira geral?

Não temos uma resposta definitiva ou única para esta questão. Por exemplo, é possível notar no discurso dos Algoristas que os seus participantes consideram esta como uma nova forma artística (isto é, a opção 1 apresentada acima). Nesta pesquisa, investigamos a expressividade procedural em situações que se encaixariam tanto na primeira como na segunda opção, dependendo do contexto, da intenção do artista ou criativo e mesmo da perspectiva a partir da qual a análise está sendo realizada. Isto é, ao contrário de buscar uma resposta para esta questão, tomamos ela como guia para as elaborações conceituais, reflexões e estudos práticos realizados ao longo da pesquisa.

2.2.2 Inteligência Artificial Expressiva (Michael Mateas e outros)

No drama interativo Façade (Michael Mateas e Andrew Stern, 2005), o jogador controla o protagonista em uma visita ao casal de amigos Trip e Grace (fig. 6, abaixo), personagens controlados pelo sistema de Inteligência Artificial (IA). É possível conversar com cada personagem individualmente e participar das conversas do casal, que está passando por uma crise conjugal. Ainda hoje, mais de dez anos depois da sua produção, este trata-se de um dos casos mais bem sucedidos do emprego de estratégias procedurais de criação para a produção de uma experiência neste gênero.[31]

Figura 6 – Façade (Mateas e Stern, 2005)

Nesta seção, examinaremos como Façade toma como base as investigações de Mateas e Stern na intersecção entre Inteligência Artificial e arte, a qual eles denominam como Inteligência Artificial Expressiva (*ou "Expressive Artificial Intelligence"*). Apresentaremos uma reflexão a respeito dos paralelos entre o sistema de IA empregado em Façade e os engines de física, presentes em muitos dos jogos modernos, e como isso levou Mateas ao seu projeto seguinte - um jogo baseado em uma simulação de "física social". Também introduziremos o conceito de "intensidade procedural", do pesquisador e designer de jogos Chris Crawford, através do qual é possível avaliar a proporção de elementos procedurais em relação aos elementos estáticos em uma dada produção.[32]

[31] Chris Crawford descreve Façade como "o avião dos irmãos Wright da narrativa interativa" (ROHRER; CRAWFORD, 2009). Esta comparação pode ser interpretada a partir do fato de que ambos são pioneiros em suas respectivas áreas, embora ainda não cumpram plenamente seus objetivos fundamentais.

[32] O conceito de **narrativa** não é tão claro ou absoluto no contexto das mídias digitais, uma vez que as produções neste meio podem diferir de maneira fundamental das estórias em meios tradicionais (FERNÁNDEZ-VARA, 2011b). Nesta pesquisa, empregamos o conceito da narrativa de uma maneira relativamente flexível, em referência à noção mais ampla da existência de conexões entre eventos, aspectos ou outros elementos do sistema, e que não precisam ser necessariamente de natureza temporal ou sequencial.

O Jogo

De forma a contextualizar e situar as investigações e elaborações teóricas que realizaremos sobre Façade, apresentaremos a seguir uma breve descrição de uma sessão típica de interação. O jogo começa com a tela completamente escura, e o som de uma mensagem de secretária eletrônica, de Trip convidando o protagonista para um jantar na casa dele. O nervosismo em sua voz, aliado a uma trilha sonora soturna, gera um clima de ansiedade. Da perspectiva do protagonista, o jogador se vê diante da porta de entrada do apartamento de Trip e Grace. É possível ouvir o casal discutindo. Trip abre a porta após um tempo (ou se o jogador bater na porta antes).

Os controles para jogar são simples: as setas no teclado servem para caminhar, o mouse permite interagir com objetos e pessoas e, ao digitar frases e pressionar "Enter", o protagonista "fala" o que foi digitado (não há som para as falas do protagonista).[33]

A sequência de acontecimentos nos primeiros momentos após o jogador adentrar o apartamento é geralmente a mesma, com algumas poucas variações. Trip cumprimenta o protagonista, e em seguida vai chamar Grace, que geralmente está na cozinha. Ela então vem até a sala e também cumprimenta a visita. Após alguns instantes, Trip volta da cozinha também. Esta sequência pode variar, principalmente se o interator fizer algum tipo de interação inesperada (por exemplo, ir direto até a cozinha logo após entrar no apartamento). De certa forma, estas primeiras interações podem ser vistas como uma oportunidade (embora breve) para interagir com cada um dos personagens individualmente antes que as discussões e intrigas comecem.

Façade é uma experiência frenética, por conta da rapidez com a qual Trip e Grace se comunicam entre si e com o protagonista, constantemente propondo (ou impondo) novos tópicos de discussão, fazendo perguntas e lançando comentários. Muitas vezes o interator tem dificuldade para conseguir responder ou comentar alguma coisa que foi dita pelos personagens antes que outra fala o interrompa. Além disso, existem as outras formas de interação à disposição do interator e dos personagens, como o deslocamento físico pelo apartamento, a interação com os objetos e com as outras pessoas. De forma a conseguir agir e participar dos diálogos, o interator precisa pensar e digitar rápido, conhecer bem os comandos e, principalmente, não ter receio em interromper Trip e Grace.[34]

Ao longo da sessão, as conversas, discussões e situações geralmente são iniciadas por Trip ou Grace. Por exemplo, Grace pode fazer um comentário sobre a decoração da sala e pedir a opinião do

[33] No caso da interação física através do mouse, o ícone se transforma de acordo com a ação possível em cada contexto. Por exemplo, se o cursor está sobre a maçaneta da porta, a ação é "abrir", se está sobre Trip ou Grace, a ação pode ser "abraçar", "confortar" ou "beijar", de acordo com a localização exata do cursor no corpo do personagem.

[34] Este ritmo acelerado parece ser uma escolha intencional de design, de forma a evitar interações muito complexas, que pudessem denunciar as limitações do sistema (por exemplo, como veremos, o sistema de interpretação de texto não é capaz de lidar com frases longas e elaboradas).

protagonista. Assim, as interações do jogador geralmente acontecem no contexto de situações que já estão em andamento. Mesmo quando a participação do interator pode contribuir para determinar qual será o desfecho de uma determinada situação (por exemplo, se o casal se reconcilia ou não após uma discussão), como veremos, fica evidente que isso acontece dentro de um universo limitado de opções.

O comportamento individual de Trip e Grace é relativamente dinâmico a maior parte do tempo, incluindo na maneira como os personagens interpretam e reagem às ações do interator. É possível notar, entretanto, que o sistema que controla o comportamento destes personagens não é capaz de lidar com situações que fogem daquelas predefinidas. Por exemplo, quando Trip oferece uma bebida ao protagonista, se este responde que não pode beber por algum motivo, o anfitrião continua insistindo. Fica claro que o sistema não prevê este tipo de desfecho para esta situação.

Mesmo no caso de situações psicologicamente mais complexas, como uma discussão envolvendo a fidelidade de um dos personagens, a participação do protagonista sempre se limita a um conjunto de possibilidades predeterminadas. Isto é, não é possível contribuir com informações ou situações novas, apenas participar daquelas que já são esperadas pelo sistema em cada contexto específico.

Quando uma determinada interação não é compreendida por algum motivo, seja por fugir ao tópico ou por conta de algum problema na interpretação do sistema, Trip e Grace simplesmente ignoram o fato, eventualmente reagem com interjeições de confusão ou desinteresse, e em seguida puxam outro assunto. A princípio esta é uma solução eficaz para impedir que o jogador note estas limitações, principalmente se for a sua primeira sessão. Entretanto, após mais algumas sessões, já é possível identificar os padrões e repetições na maneira como Trip e Grace reagem às diferentes situações, assim como nos tópicos de conversação, frases e momentos específicos que ocorrem.

Na maior parte das sessões com Façade, o final do enredo (isto é, a sua conclusão) nunca chega, pelo menos não de acordo com o arco narrativo imposto pelo sistema.[35]

[35] Após o término de uma sessão, Façade gera um roteiro (ou "*stageplay*") da narrativa que ocorreu, incluindo as falas dos personagens e do próprio protagonista. A duração de uma sessão que cumpre o arco narrativo previsto pelo sistema geralmente dura cerca de vinte minutos.

De uma maneira geral Façade é uma experiência interessante e geralmente divertida, mesmo após múltiplas sessões. Um dos motivos disso é a variabilidade relativamente alta na combinação de possíveis situações em cada sessão (de acordo com os desenvolvedores, no máximo cerca de 25% do conteúdo pode ser visto a cada nova sessão). Porém o que parece contribuir de maneira mais significativa para o apelo de Façade é a constante sensação de que os personagens estão reagindo às interações do protagonista, considerando sua presença e as suas contribuições, mesmo quando na maior parte do tempo isso seja apenas uma ilusão.

Uma Linguagem de Comportamento

A programação dos personagens Trip e Grace em Façade é realizada a partir de uma notação desenvolvida especialmente para o projeto, chamada Uma Linguagem de Comportamento (em Inglês, "A Behavior Language", abreviada pela sigla ABL). A partir desta linguagem são definidas as "personalidades" dos personagens e as possíveis ações associadas a eles, incluindo as combinações e relacionamentos entre elas.

O sistema também leva em consideração aspectos relacionados aos objetivos dos personagens, através de critérios como sucesso e fracasso, bem como a prioridade entre eles e o contexto no qual eles podem (ou devem) ocorrer.

Existe um módulo específico para a interpretação das interações comunicacionais do jogador, incluindo as frases que são digitadas para o protagonista "falar" com os outros personagens. A interpretação desta comunicação verbal é realizada de uma maneira relativamente flexível, e sem uma exigência muito rigorosa quanto à estrutura ou semântica do que é digitado.

Assim, por exemplo, se no jogo Grace demonstra estar brava com Trip e o jogador quer apaziguar as coisas, a princípio não há diferença entre digitar uma fala completa e bem estruturada como "Grace, não fique brava com Trip" ou uma série de palavras soltas sugerindo a mesma ideia, como "Grace não brava". As palavras mais significativas da frase são identificadas e o conjunto delas é interpretado pelo sistema de acordo com o contexto em que ela ocorre (além disso, a reação de cada personagem a uma mesma interação pode ser diferente). A frase é então classificada pelo sistema em categorias, como "entendimento", "desentendimento", "aliança", "provocação" e assim por diante.

O Sistema de IA e o Engine de Física: uma analogia

A área de pesquisa em Inteligência Artificial, considerada de uma maneira mais ampla, dedica-se ao desenvolvimento de modelos e técnicas para a simulação da mente humana, ou de determinados aspectos ou elementos do seu funcionamento. Estes modelos também podem ter aplicações práticas, como a criação de agentes virtuais ou sistemas especialistas.

No caso da Inteligência Artificial Expressiva, as técnicas de IA são empregadas para fins artísticos ou expressivos, com o objetivo de implementar personagens com determinados comportamentos e características, e que atendam a uma proposta ou projeto de criação particular. Enquanto nos filmes ou livros os personagens existem de maneira estática ou fixa, apresentados na forma de descrições verbais, imagens ou situações predefinidas, um personagem de IA é descrito na forma de características e regras de comportamento ou personalidade. Este agente virtual existe de maneira dinâmica, de certa forma "viva", na memória do sistema computacional e pode interagir com outros elementos da simulação (como o cenário, os objetos e outros agentes virtuais), e eventualmente com entidades externas (como um interator, ou outros elementos que existem fora da simulação). Esta abordagem expressiva ao uso da IA é adotada principalmente nas artes, como por exemplo nas produções experimentais de "gamearte" da artista Suzete Venturelli (VENTURELLI e MACIEL, 2004).

Com o objetivo de esclarecer como se dá na prática a aplicação dos conceitos de IA no contexto de uma simulação computacional, Mateas sugere uma analogia com o funcionamento de um engine de física (ou "*physics engine*"). De uma maneira geral, o "engine" é o nome dado à parte de um programa ou algoritmo que contém a - ou que é responsável pela - formalização dos comportamentos e propriedades fundamentais de um dado sistema. Assim, o "engine de física" seria o sistema responsável pela simulação dos elementos "físicos" do ambiente virtual. Além deste, podem haver outros sistemas responsáveis por determinar outros elementos ou aspectos da simulação, tais como a forma de apresentação visual e sonora, a interface com o usuário ou as regras específicas do sistema.[36]

Mateas explica que o sistema de Inteligência Artificial e o engine de física tem em comum o fato de ambos serem fontes de **comportamento** (MATEAS, 2003). Esta analogia é útil como recurso didático, pois permite identificar e evidenciar determinados aspectos relacionados ao emprego das técnicas de IA - bem como de outras estratégias procedurais de criação - para fins expressivos.

[36] A rigor, o termo "engine" pode ser empregado tanto em referência a cada um destes sistemas particulares, ou ao conjunto deles, tomados como um todo. O conceito do engine reflete a lógica da abordagem modular, que caracteriza a programação orientada a objetos (ou "Object-Oriented Programming"). Alguns autores escrevendo em Português empregam a tradução literal "motor" para o termo. Nesta pesquisa é adotado o termo original em Inglês (sem itálico).

De forma a ilustrar esta comparação a partir de exemplos práticos, serão apresentados a seguir dois jogos, escolhidos principalmente por conta da objetividade e simplicidade pela qual representam cada uma das áreas em questão. Os jogos são Angry Birds, cuja mecânica central se baseia em um engine de física, e o clássico Pac-Man, no qual a jogabilidade é determinada por um sistema de IA. Em seguida, será feita uma comparação entre os dois sistemas, de acordo com a analogia proposta por Mateas.[37]

Angry Birds (Rovio, 2009-) é uma série de jogos casuais apresentando uma estética visual de desenho animado, na qual o objetivo de cada fase é utilizar um estilingue para lançar projéteis - os "pássaros" - de forma a atingir e eliminar inimigos - os "porcos" -, bem como derrubar as estruturas nas quais eles se protegem. O comportamento dos diversos elementos do jogo, como a trajetória do pássaro ao ser arremessado, e o efeito nos porcos atingidos, são definidas por um engine de física 2D.[38]

A interação com o jogo é relativamente simples, bastando definir o ângulo e a velocidade com a qual o pássaro irá deixar o estilingue a cada arremesso. Uma vez atingidos, os elementos que compõem as estruturas que abrigam os "porcos" podem se deslocar, cair e eventualmente sofrer fraturas ou quebrar, de acordo com o material da sua composição (como pedra, vidro, madeira e assim por diante). Por exemplo, na situação representada na figura da próxima página (fig. 7; p. 49), o porco (à direita na imagem) está protegido dentro de uma estrutura relativamente simples e frágil, composta principalmente de partes de madeira. O pássaro (à esquerda na imagem) foi arremessado de forma a atingir a parte de cima da estrutura, o que vai levá-la a desmoronar, atingindo o porco no processo. Como esta é a primeira fase, para facilitar o processo há ainda um bloco de pedra no topo da estrutura, tornando-a mais frágil por conta do peso adicional.

Embora os personagens do jogo apresentem reações, na forma de animações de expressões faciais - os porcos demonstram "medo" antes do arremesso, e "satisfação" se não forem atingidos -, este aspecto da experiência não tem efeito no sistema do jogo propriamente dito. Isto é, em Angry Birds a mecânica do jogo, os desafios e as estratégias que podem ser empregadas para vencer cada fase, se baseiam exclusivamente na lógica do seu engine de física.

[37] Estes exemplos também foram escolhidos por não apresentarem elementos artísticos e expressivos significativos. Desta maneira é possível focar a análise sobre os mecanismos fundamentais empregados em cada caso, independentemente do seu uso específico.

[38] A premissa temática do jogo é que os pássaros querem se vingar dos porcos por terem roubado seus ovos. Na prática, este enredo não interfere ou afeta de maneira significativa a mecânica do jogo. Angry Birds foi diretamente inspirado no jogo Crush the Castle (Armor Games, 2009), cuja mecânica é praticamente idêntica, diferindo apenas pela temática medieval.

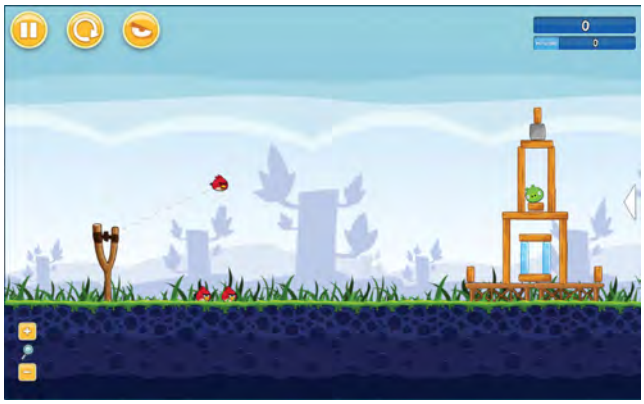


Figura 7 – Angry Birds (Rovio, 2009-)

Figura 8 – Pac-Man (Namco, 1980)

Pac-Man (Namco, 1980) é um jogo clássico no qual o interator controla um personagem (Pac-Man) por um labirinto de forma a capturar todos os pontos (ou pastilhas) distribuídos ao longo do caminho, ao mesmo tempo em que evita ser alcançado pelos "fantasmas" de diferentes cores (fig. 8, acima).[39]

O comportamento de cada fantasma é definido por um algoritmo simples de IA, diferente para cada um deles - o vermelho persegue o jogador diretamente, o rosa e o azul se deslocam de forma a se aproximar pela frente de Pac-Man e o laranja se move de maneira aleatória. No labirinto também existem *power pellets* (ou "pastilhas de poder"), os pontos maiores e em quantidade limitada, que quando capturados tornam todos os fantasmas temporariamente vulneráveis (quando isso acontece, o comportamento dos fantasmas também muda, e eles passam a fugir do jogador).[40]

Apesar da simplicidade das regras e dos algoritmos individuais de cada fantasma, há um equilíbrio cuidadoso entre estes elementos. Isso leva à emergência de situações variadas e complexas, resultando em um jogo divertido e desafiador.

Em Pac-Man, o jogador precisa tomar decisões rapidamente, de maneira quase instintiva. Por exemplo, no momento representado na figura acima (fig. 8), Pac-Man (localizado no canto inferior direito) já capturou alguns pontos e está se deslocando para cima. O fantasma vermelho está próximo, e deslocando-se para baixo, em sua direção. O jogador tem duas alternativas: pode aguardar até que o fantasma "escolha" um caminho, e então rapidamente mudar de direção se necessário, ou pode decidir ir direto até o *power-pellet*, mas arriscar não chegar nele antes do inimigo.

Todos estes fatores, ou pelo menos aqueles conhecidos pelo jogador, fazem parte do seu processo de tomada de decisão, mesmo que de maneira inconsciente.

[39] A imagem de Pac-Man representada nesta página foi obtida a partir de um registro em vídeo de uma emulação da versão original.

[40] O algoritmo exato para o comportamento de cada um dos fantasmas em Pac-Man ainda é motivo de debate.

Sem os fantasmas, Pac-Man não teria dificuldade - bastaria capturar todos os pontos no labirinto. Ou então, se todos os fantasmas tivessem o mesmo comportamento, o jogo também teria pouco apelo. Assim, o que caracteriza Pac-Man e define a experiência do jogo é principalmente o sistema de IA que governa os fantasmas.

Retomando a analogia proposta por Mateas, enquanto que em Angry Birds as leis descritas nos algoritmos do **engine de física** determinam o comportamento dos elementos físicos no jogo (colisões, quedas e fraturas), em Pac-Man as regras e instruções contidas no **sistema de IA** definem o comportamento dos fantasmas (como cada um deles se desloca pelo labirinto).

Em ambos os sistemas ocorre a emergência de situações complexas e imprevisíveis a partir de regras simples. Em Angry Birds é impossível prever todos os possíveis arremessos e as suas respectivas consequências no ambiente do jogo (por conta disso, muitas vezes a melhor solução para uma determinada fase é diferente de um jogador para o outro, ou mesmo da solução prevista pelos próprios criadores do jogo). Os fantasmas de Pac-Man também estão constantemente surpreendendo mesmo os jogadores mais experientes.

Cada um destes jogos explora aspectos diferentes da experiência (o **físico** em Angry Birds e o "**psicológico**" no comportamento dos fantasmas em Pac-Man), entretanto, do ponto de vista da sua implementação, a diferença entre estes dois sistemas é essencialmente arbitrária. A maior parte dos conceitos, técnicas e estratégias envolvidas no ato de projetar e programar um engine de física são comuns a um sistema de IA.

Isto é, da perspectiva do processamento computacional, não há grande diferença entre a queda de um bloco de pedra atingido por um pássaro em Angry Birds ou a "tomada de decisão" de um fantasma em ir de encontro a Pac-Man, por exemplo.

Em ambos os casos o sistema lida com uma situação na qual (1) há uma entidade em uma situação de equilíbrio (o bloco em repouso; o fantasma antes de sair do centro do labirinto) e (2) sofre uma interferência externa (o bloco é atingido por um pássaro, por exemplo; o fantasma "percebe" Pac-Man), levando-o a (3) seguir um impulso fundamental (a queda do bloco devido à gravidade, a perseguição de Pac-Man de acordo com a sua "personalidade").[41]

[41] Esta é apenas uma entre tantas outras perspectivas possíveis de comparação entre os elementos e comportamentos citados. A intenção é justamente de ilustrar a arbitrariedade das associações entre o código de um dado sistema computacional e os comportamentos que dele resultam (e os significados a eles associados).

Alguns dos termos no parágrafo acima estão entre aspas para evidenciar que tratam-se de analogias livres, entre as atividades ou conceitos da ordem da mente humana, e os processos computacionais empregados na sua simulação. Isto é, um fantasma do Pac-Man não "percebe" de fato, pelo menos não da mesma forma que um ser humano. Entretanto, a rigor o mesmo acontece com relação à suposta existência de um aspecto "físico" na simulação. Isto é, da mesma forma que um fantasma em Pac-Man não "decide" tomar um determinado caminho, um objeto em Angry Bird nunca "cai" de fato.

Mesmo na mente do designer ou programador, estes conceitos não existem de maneira absoluta e objetiva. Por exemplo, a modelagem do comportamento de um personagem em uma narrativa com base em conceitos como "agressividade" ou "atenção", não significa necessariamente que o designer acredita que estas mesmas categorias existem na mente humana. Tratam-se de modelos a partir dos quais é possível criar uma simulação da realidade - e é justamente na definição das diretrizes destes modelos que reside a autoria procedural.

Por exemplo, alguns designers tomam como base conceitos de redes neurais para a modelagem do comportamento de personagens, simulando os processos mentais humanos em um nível mais fundamental. O resultado é algo como as Evolved Virtual Creatures de Karl Sims (1994), criaturas digitais que lentamente aprendem a se locomover, realizar determinadas atividades e mesmo interagir entre si, a partir de algoritmos genéticos.

Este projeto serviu de inspiração para a implementação do comportamento dos personagens em Incidente em Varginha (Perceptum, 1998), considerado o primeiro jogo de ação em primeira pessoa produzido no Brasil. Marcos Cuzziol, um dos designers do jogo, explica que optou por esta alternativa devido às limitações e dificuldades enfrentadas para criar e balancear os modelos de comportamento de forma a obter o resultado desejado (CUZZIOL, 2007:49).[42]

Esta mesma solução poderia ser inadequada para outro projeto, ou outro designer. Por um lado, o emprego de algoritmos genéticos permite a emergência de sistemas potencialmente mais complexos e consistentes, dando acesso a estruturas e funcionalidades que não existiriam normalmente. Porém, isso também introduz uma camada adicional de abstração no processo, diminuindo o controle criativo sobre certos aspectos e elementos da simulação. É escolha do designer qual solução é mais adequada, tanto do ponto de vista prático quanto expressivo.

[42]Marcos Cuzziol organizou a exposição Gameplay em 2009 no Itaú Cultural, em São Paulo, bem como inúmeros outros eventos relacionados a arte e tecnologia.

Física Social em Prom Week

Esta analogia entre um sistema de Inteligência Artificial e um engine de física é sugerida no conceito de "**física social**" (ou "*social physics*"; MATEAS, 2011), no qual se baseia o jogo Prom Week (2012). Produzido no Expressive Intelligence Studio (University of California, Santa Cruz), sob orientação de Michael Mateas e Noah Wardrip Fruin, o jogo simula as relações sociais entre os alunos de uma escola durante a semana que antecede a festa de formatura.[43]

Em Prom Week é possível decidir como cada um dos personagens deve agir com relação aos demais. O jogador tem acesso aos "pensamentos" dos alunos, incluindo como eles se sentem uns em relação aos outros. Isso leva a situações complexas, geralmente envolvendo intrigas, como é de se esperar no contexto de uma escola. Por exemplo, se um determinado aluno está apaixonado por uma garota, e ele descobre que ela está brava com um outro aluno, ele pode tentar conquistar a afeição dela falando mal deste aluno, criando assim uma cumplicidade.

Diferentemente de Façade, no caso de Prom Week as ações e diálogos não podem ser controlados diretamente pelo jogador, que se limita a escolher opções predefinidas em uma lista, como "Convidar fulana para sair" ou "Falar mal de fulano".[44]

Convém notar que, apesar dos criadores de Prom Week empregarem o termo "física social" para descrever a mecânica de jogo, há uma diferença fundamental entre este sistema e um engine de física moderno, que é o nível de granularidade da simulação.

Em um dado sistema, o conceito de granularidade refere-se à unidade mínima de conteúdo sobre a qual determinadas operações podem ser aplicadas. Enquanto que Prom Week representa as relações sociais de uma maneira simbólica, em um engine de física não existem (ou não precisam existir) abstrações para elementos ou eventos particulares, uma vez que há um alto grau de granularidade na descrição das propriedades e regras da simulação

Por exemplo, em uma simulação de física, para derrubar uma caixa que está sobre uma mesa não é necessário ter à disposição um comando do tipo "derrubar caixa". Isto é, a simulação de física não precisa conhecer abstrações como a ideia de "derrubar" algo, ou o conceito de uma "caixa" (ou a abstração de uma caixa específica). Basta aplicar uma força ao objeto, de forma que a ação esperada ocorra.

[43] Prom Week emprega um sistema de IA próprio, chamado "*Comme il Faut*" (CiF).

[44] Esta mudança de abordagem reflete os problemas enfrentados com o sistema de interpretação de linguagem natural empregado em Façade, que se mostrou frustrante para os jogadores. Como veremos mais adiante, naquele caso a liberdade oferecida ao jogador não era compatível com a capacidade do sistema em interpretar a interação.

Em Prom Week, por outro lado, a granularidade é muito menor, e o nível de abstração das representações é mais alto. Por exemplo, uma das opções que o jogador tem para se relacionar é o ato de "se gabar" (ou contar vantagem) para alguém.[45]

De forma a apresentar um nível de granularidade mais próximo ao de um engine de física, Prom Week precisaria oferecer uma complexidade maior de funcionalidades. Por exemplo, poderia oferecer ao interator a opção de fazer um personagem comunicar a outro uma característica de sua personalidade, e incluir um mecanismo de avaliação pelo qual os personagens podem julgar as diferentes características disponíveis. Com isso, de forma a fazer um personagem "se gabar" para outro, o jogador precisaria fazê-lo comunicar uma característica sua que ele sabe que é valorizada pelos demais personagens. Note que, nesta implementação hipotética, o sistema não precisaria conter uma abstração para o conceito de "se gabar", uma vez que esta operação passa a ser realizada em um nível de abstração mais baixo.

De certa forma Façade funciona de maneira similar a este exemplo, embora com uma série de limitações. Como visto anteriormente, a maneira como o jogador trata Trip e Grace influencia as suas "opiniões" sobre ele, de uma maneira gradual e relativamente granularizada. Assim, por exemplo, se o jogador sempre defender Grace nas suas discussões com Trip, ela passa a sentir mais empatia por ele. Com isso, aumenta a chance de ela tratar bem o protagonista em uma interação futura, por exemplo. Isso também torna mais provável que Grace se volte contra Trip, uma vez que ela tem a validação do jogador para tal.[46]

Porém, esta granularidade não é encontrada na camada do sistema de conversação de Façade. Por um lado, há uma certa abertura na maneira como o sistema interpreta as falas do jogador. Por exemplo, se o interator deseja conversar com os outros personagens a respeito de uma foto que está pendurada na parede, basta fazer uma pergunta ou menção sobre o assunto, com perguntas como "Esta foto é de onde?" ou "E esta foto na parede?".

Entretanto, em Façade o nível de granularidade **sugerido** neste tipo de interação é muito maior do que a granularidade **real** do sistema por trás da simulação. Na prática, embora o jogador possa digitar uma infinidade de frases para se comunicar sobre um certo tópico, todas ativam uma mesma situação, de maneira similar a um diálogo baseado em itens fixos em um menu.

[45] O conceito de níveis (ou graus) de abstração é aprofundado mais adiante na tese (p. 156).

[46] Isso não muda o fato de que o sistema de Façade só reconhece situações previstas pelos seus criadores. Por exemplo, por um lado, ofensas geralmente são reconhecidas - xingar os personagens leva a reações negativas deles. Por outro lado, o jogo não reconhece o ato de gabar-se, presente em Prom Week, por exemplo. Assim, qualquer tentativa de fazê-lo é ignorada (ou interpretada de outra maneira) pelo sistema. A princípio, esta limitação deriva justamente das escolhas de design do projeto. Isto é, enquanto que o ato de se gabar é relevante no contexto de Prom Week, em Façade não é.

Isto é, na situação descrita anteriormente, todas as frases digitadas pelo jogador relacionadas à foto equivalem a escolher uma mesma opção em um menu, como é comum nos adventures (ou "aventuras", gênero de jogo baseado em quebra-cabeças e diálogos), por exemplo, ou no próprio Prom Week.[47]

A princípio, a razão pela qual um artista ou criativo decide adotar estratégias procedurais de criação não é necessariamente a ampliação das possibilidades e estados que uma simulação pode assumir, ou do volume e complexidade de conteúdo presente no sistema (embora este seja um uso comum para esta abordagem). A diferença fundamental entre a criação procedural e a tradicional consiste na própria natureza destas duas abordagens, e nas possibilidades criativas e expressivas distintas oferecidas por cada uma.

A intenção dos criadores de Façade não era desenvolver uma simulação capaz de reconhecer e produzir toda e qualquer situação imaginável - isso seria impraticável. O objetivo era que, entre as situações que de fato pudessem acontecer em uma dada sessão de jogo, a maior parte delas fosse resultante do sistema, ao invés de diretrizes e roteiros predefinidos. Isto é, embora o conjunto de possíveis eventos e situações seja limitado, a maior parte do que acontece emerge a partir de um processo relativamente granularizado.

[47]Nesta tese nos referimos ao gênero "adventure" pelo próprio termo em Inglês. Trataremos sobre o gênero do adventure em maior profundidade mais adiante, ao examinar a questão da coerência na mecânica de jogo (p. 241).

Arte Cinética Digital

A analogia que Mateas propõe entre o sistema de IA e um engine de física também reflete a maneira como ele considera a aplicação do seu conceito de Inteligência Artificial Expressiva. O autor afirma que o engine de física lida com "a parte 'morta' do jogo, os processos puramente mecânicos e casuais que não operam no nível da **intencionalidade** e que portanto não respondem às ações do jogador no nível do **significado**" (MATEAS, 2003:3).[48]

Esta afirmação sugere que a porção física de uma simulação (isto é, o engine de física) não pode conter intencionalidade ou significado. Isso implicaria que este aspecto da simulação também não pode ser empregado de maneira expressiva, o que nos parece ser uma distinção questionável.

É possível traçar uma comparação com as obras de Arte Cinética, cuja expressividade se encontra nas propriedades físicas, movimentos e interações potenciais entre os diferentes elementos que compõem cada objeto ou escultura (e eventualmente as interações com o público também). Da mesma forma, os objetos "físicos" em uma simulação digital também podem estar associados a uma intenção expressiva.

Há ainda um potencial expressivo adicional desta abordagem no contexto do meio digital, uma vez que neste caso o criador tem acesso praticamente ilimitado às propriedades físicas dos objetos na simulação (não há limitação quanto à escala ou às proporções entre objetos, por exemplo), bem como às próprias leis às quais estes elementos estão submetidos.

Embora um engine de física geralmente tenha como objetivo simular com realismo as leis da física (consideradas as inevitáveis limitações e ocasionais simplificações), a princípio não há limites sobre como estas regras podem ser implementadas - é possível modificar a força da gravidade, incluir novas forças e leis arbitrárias e assim por diante. Com isso, abre-se um nível adicional de abstração, e consequentemente novas oportunidades expressivas.[49]

[48] Citação: grifo nosso; tradução nossa - texto original: "the 'dead' part of the game, the purely mechanical, causal processes that don't operate at the level of intentionality and thus don't respond to player activity at the level of meaning"

[49] Um exemplo deste tipo de abordagem é o jogo A Slower Speed of Light (MIT Game Lab, 2012), tratado mais adiante (p. 177).

Uma Forma Híbrida: entre o procedural e o estático

O conceito de Inteligência Artificial Expressiva proposto por Mateas toma como base as estratégias procedurais de criação de forma a produzir o que ele denomina como um "drama interativo". Entretanto, como pode ser verificado em Façade, este processo não envolve apenas elementos procedurais, mas também estratégias expressivas tradicionais, de natureza estática e linear. O resultado disso é uma espécie de "forma híbrida".

Chris Crawford, designer de jogos e pesquisador, propõe o conceito de intensidade procedural (ou intensidade de processo) como uma maneira de mensurar o quanto um determinado programa ou produção digital explora a proceduralidade. Segundo Crawford, esta medida representa "o grau em que um programa enfatiza processos ao invés de dados." (CRAWFORD, 1987)[50]

Como será visto mais adiante, em um escopo local (ou de curto prazo), Façade apresenta um grau de intensidade procedural relativamente alto. Por conta disso, é comum a ocorrência de situações pontuais nas quais os personagens reagem de maneira coerente com as interações do jogador. Por exemplo, há situações nas quais Trip e Grace estão discutindo e, por conta de algo que o jogador diz, eles param de brigar e fazem as pazes. Momentos como estes transmitem ao jogador a ideia de que as suas ações influenciam e tem impacto no universo da simulação, caracterizando um alto grau de agência.

Porém, como vimos, ao considerar Façade a partir de um escopo mais amplo, determinadas limitações do sistema se revelam. Por exemplo, torna-se aparente que há determinados eventos e situações recorrentes no jogo, como um diálogo que sempre tende a tomar um mesmo rumo (independentemente da interação do jogador). Da mesma forma, é possível notar que muitas ações e falas, que seriam perfeitamente coerentes em uma determinada situação, não são reconhecidas pela simulação. Torna-se evidente que a simulação se baseia em um repertório limitado de possíveis situações, assuntos e ações que podem ser realizadas.

Em um escopo ainda mais amplo, é possível notar a camada de controle referente à narrativa global de Façade, que direciona o comportamento dos personagens de forma a garantir que um determinado enredo se realize, atendendo a um arco narrativo clássico.

Assim, a experiência em Façade é melhor definida como uma forma híbrida, apresentando uma abordagem procedural e dinâmica em um escopo local, e uma narrativa mais tradicional, de natureza linear e estática, em um escopo médio e global.

[50]Citação: tradução nossa - texto original: "Process intensity is the degree to which a program emphasizes processes instead of data" (sem numeração de página)

2.2.3 Processamento Expressivo (Noah Wardrip-Fruin)

O pesquisador Noah Wardrip-Fruin tem um estudo sobre o emprego expressivo dos processos computacionais, com ênfase na área da literatura digital, mais especificamente no gênero da ficção. Porém, suas investigações não se limitam a estas práticas, sendo que grande parte das suas elaborações se dá a partir da análise de jogos digitais. Sua intenção é de contribuir na compreensão e reflexão sobre os diferentes tipos de produção criativa e expressiva em meio digital.[51]

É possível identificar intersecções entre aspectos e elementos dos estudos de Wardrip-Fruin, e a nossa própria pesquisa. Ao mesmo tempo, existem divergências fundamentais entre as duas abordagens. Como veremos a seguir, o processo de examinar estas similaridades e diferenças permite reforçar e enriquecer a definição do nosso próprio objeto de pequisa.

Abordagem Metodológica

A pesquisa de Wardrip-Fruin se dá sobre um universo de análise relativamente amplo, englobando produções artísticas e acadêmicas, além de diferentes técnicas e ferramentas de criação. Sua proposta é de desenvolver uma tipologia sobre as práticas criativas em mídias digitais, através da definição e identificação de conceitos, terminologias, e categorizações.

Por exemplo, o autor define categorias de forma a distinguir entre obras que são unicamente **autoradas** digitalmente (ou *"digitally-authored"*) e aquelas nas quais o **suporte** computacional também é necessário para a sua exibição e recepção (denominadas como *"digital media literature"*, ou "literatura de mídia digital", por conta do seu foco de estudo na literatura).

Outra distinção importante feita em sua pesquisa é entre as produções nas quais os processos levam a variações comportamentais dentro de um determinado sistema, e aquelas em que isso não ocorre (relacionada ao conceito de agência).

Wardrip-Fruin também utiliza diagramas de forma a ilustrar alguns dos conceitos tratados no seu estudo, e para tornar mais claras as intersecções e o relacionamento entre os diferentes elementos e aspectos envolvidos.

[51] A literatura digital (ou eletrônica) é uma área relativamente ampla, englobando produções literárias baseadas em computadores tanto na etapa da criação quanto da exibição. Assim, inclui desde produções que empregam pouco ou nenhum elemento ou estratégia procedural (como *hyperlinks* ou animações simples), até variadas formas de obras generativas. Temas na área da Inteligência Artificial também são presentes na pesquisa de Wardrip-Fruin, principalmente a partir do seu emprego na modelagem e simulação de personagens virtuais.

Por exemplo, na figura abaixo (fig. 9) o gradiente vertical no círculo representa o equivalente ao conceito do nível de intensidade procedural, mencionado anteriormente (p. 56), e as duas metades na horizontal indicam o tipo de mídia - à esquerda, as produções em suporte digital, e à direita aquelas autoradas digitalmente (WARDRIP-FRUIN, 2006:15).

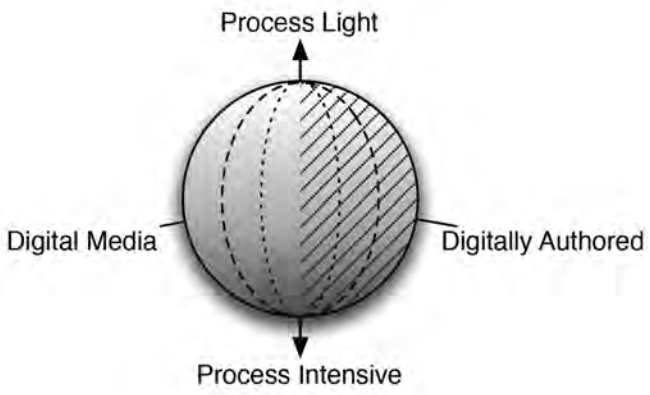


Figura 9 – Diagrama: tipos de mídia (WARDRIP-FRUIN, 2006)

O processo de localizar uma determinada produção digital na superfície do diagrama permite identificar e evidenciar de maneira clara e objetiva seus aspectos e características, bem como traçar comparações e relacionamentos com outras produções. Para Wardrip-Fruin, o estudo das mídias digitais se beneficia desta abordagem de natureza metódica e sistemática.

Expressividade Procedural / Processamento Expressivo

Apesar da similaridade entre a terminologia empregada por Wardrip-Fruin para denominar o seu objeto de estudo - **processamento expressivo** (ou "*expressive processing*") - , e aquela utilizada nesta pesquisa - **expressividade procedural** -, existem diferenças fundamentais no significado de cada um destes conceitos.

A primeira distinção, e a mais evidente, é a inversão do papel que o conceito da expressividade tem em cada um dos termos. Enquanto na denominação empregada nesta pesquisa a expressividade é o substantivo e representa o elemento central, no termo de Wardrip-Fruin ela é secundária, sendo atribuída como qualidade ao "processamento" (ela é o adjetivo).[52]

[52]Esta diferença terminológica é sutil, e não implica necessariamente em uma divergência fundamental entre os conceitos. Note que no idioma Inglês as palavras que compõem os termos se invertem ("expressive processing" e "procedural expressivity"), pois neste caso o adjetivo precede o substantivo. Isso não muda, entretanto, o sentido e a função gramatical dos termos.

Esta diferença na ênfase dada em cada um dos termos reflete perspectivas distintas de análise entre as duas pesquisas. As investigações de Wardrip-Fruin tomam como ponto de partida o meio digital, examinando como a computação pode ser empregada de maneira expressiva. A nossa pesquisa, por sua vez, toma como base o conceito de uma forma expressiva, e investiga como ela pode se realizar no contexto das mídias digitais, através das estratégias procedurais de criação.

Outra diferença entre o nosso recorte e aquele adotado por Wardrip-Fruin é a palavra escolhida para representar o elemento computacional em cada caso. No estudo de Wardrip-Fruin, esta palavra é "processamento" (ou "*processing*"), enquanto que nesta pesquisa usamos o termo "procedural" (relativo a "procedimento"). Esta diferença terminológica reflete algumas distinções adicionais entre os dois conceitos.

A primeira refere-se à etimologia das palavras. De um lado, a palavra "**procedural**" está associada ao conceito objetivo de "procedimento", que significa uma maneira específica de fazer alguma coisa. Este é um termo familiar na área da computação, denominando uma função ou determinada série de instruções.

A palavra "**processamento**", por sua vez, tem um escopo mais amplo, pois pode se referir tanto ao processamento de uma função ou algoritmo no contexto da computação (isto é, de um "procedimento"), quanto a um "processo" (de natureza "processual"), em uma perspectiva mais geral.

Outra distinção está no fato de que, no termo empregado por Wardrip-Fruin, há um foco explícito no momento da execução do algoritmo ou programa, isto é, no "processamento". No termo empregado na nossa pesquisa, esta ênfase não é sugerida, refletindo a intenção de investigar a autoria procedural da perspectiva do artista ou criativo, sem focar no momento da exibição ou recepção da obra (ou mesmo do momento da geração, no caso do produto final ser de natureza estática).[53]

[53] Há ainda outra interpretação possível sobre o termo escolhido por Wardrip-Fruin, que é considerar a "expressividade" como uma característica do processamento computacional em si, ao invés de se referir à intenção do criador. Este não é, entretanto, o sentido pretendido pelo autor (e tampouco em nossa pesquisa, como vimos).

Terminologia e Conceitos

As mídias digitais ainda são uma tecnologia relativamente nova, em comparação com outros meios tradicionais. Embora exista uma terminologia comum sendo estabelecida entre os autores da área, e naquelas relacionadas, ainda existem muitas discrepâncias e lacunas. Estas divergências são parte integrante do processo científico, que geralmente se beneficia da existência de diferentes perspectivas, bem como dos debates e diálogos entre elas.

Wardrip-Fruin emprega em sua pesquisa uma série de conceitos e termos que possuem uma definição relativamente bem estabelecida. Entretanto, ele também constrói uma terminologia parcial própria. O autor introduz determinados conceitos e palavras que lhe permitem descrever e representar com precisão as suas proposições e investigações. Alguns destes termos tratam-se de novas interpretações ou mesmo redefinições de conceitos já existentes.

Um destes termos é a noção de uma "mídia fixa" (ou "*fixed media*"), usado em referência a suportes como o filme ou o livro, que a princípio é o equivalente ao que é denominado nesta pesquisa como um meio de natureza "estática". Wardrip-Fruin contrapõe este conceito ao de mídia digital (ou *"digital media*"), uma comparação compreensível e justificada no contexto em que é apresentada.

Em nossa pesquisa, entretanto, consideramos que a diferença entre estes dois conceitos não é tão clara ou rígida. Como vimos, na sua qualidade de metamídia, o computador tem a capacidade de emular (ou simular) formas tradicionais, incluindo a exibição de conteúdo de natureza fundamentalmente estática - isto é, a princípio um livro, fotografia ou filme podem ser exibidos através de um computador, de maneira análoga aos seus respectivos suportes nativos.

Neste sentido, o conteúdo estático que reside na memória de um computador pode ser considerado como "mídia fixa", uma vez que sua natureza - isto é, sua composição - é de fato fixa. Wardrip-Fruin reconhece esta questão, apontando que um filme (estático, linear e fixo) exibido em um computador **não** pode ser considerado como "mídia digital" só por conta do suporte ser de natureza digital.

Wardrip-Fruin também propõe um modelo para descrever o meio digital, que ele representa no diagrama reproduzido abaixo (fig. 10). O modelo inclui a entidade do(s) autor(es), e o meio digital representado pelos seus dois aspectos, os dados e o processo, além de uma camada denominada "superfície", através da qual se dá a Interação com o público e com os processos e fontes de dados externas (WARDRIP-FRUIN, 2006:12).[54]

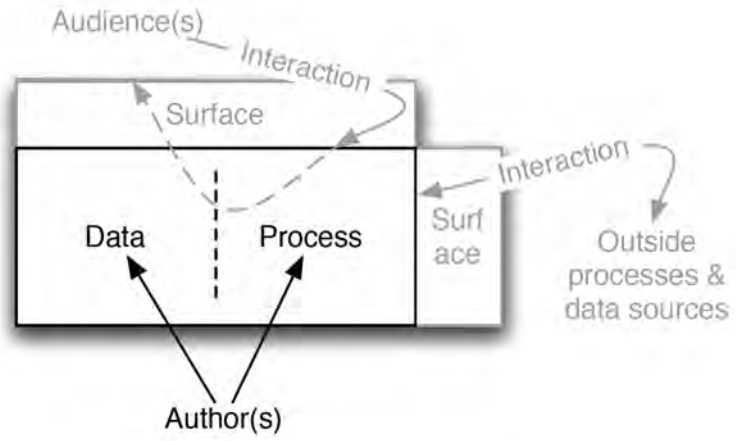


Figura 10 – Diagrama: modelo do meio digital (WARDRIP-FRUIN, 2006)

O autor toma este modelo como base para ilustrar o foco da sua investigação, que se dá sobre os processos computacionais, em oposição à "superfície". Entretanto, no contexto deste modelo, o conceito de superfície engloba não apenas a interatividade (ou os estímulos e dados externos), mas também todo tipo de processamento computacional que se dá no momento da recepção. A princípio, isso implica que o universo de análise da pesquisa de Wardrip-Fruin se limita a produções cuja forma final é de natureza estática, como o meio impresso, por exemplo.

[54] Tradução nossa a partir do texto na figura: Autor (Author); Dados (Data); Processo (Process); Superfície (Surface); Interação (Interaction); Público (Audience) e Processos e fontes de dados externas (Outside processes & data sources).

2.2.4 Unidades Operacionais (Ian Bogost)

O pesquisador e designer de jogos Ian Bogost propõe uma metodologia de estudo sobre as manifestações comunicacionais e expressivas em mídias digitais, com ênfase nos videogames, a partir do conceito de "unidade operacional" (BOGOST, 2006). A unidade operacional é definida como um elemento ou ideia recorrente em uma determinada obra, e que se encontra de alguma maneira condensada (ou codificada) simbolicamente - como veremos, Bogost também emprega o termo "unidade de significado".[55]

O autor explica o conceito tomando como exemplo a situação social do "encontro casual" - isto é, o encontro fortuito entre desconhecidos, a princípio com conotações românticas ou sexuais.

Bogost examina como esta situação é representada em quatro situações diferentes: (1) na poesia de Charles Baudelaire, (2) no trabalho de Charles Bukowski, (3) no filme O Fabuloso Destino de Amélie Poulain (Jean-Pierre Jeunet, 2001) e (4) na série de jogos The Sims (Maxis, 2000-) - a figura abaixo ilustra os quatro exemplos (fig. 11).[56]

Figura 11 – Ilustração representando mulher no século XIX (esq.), desenho de Robert Crumb para texto de Bukowski, cena de Amélie Poulain e tela do jogo The Sims

[55]Para a formulação da sua teoria, Bogost traça paralelos entre conceitos da Teoria Literária e da Computação, além de outras áreas, como Filosofia, Psicologia e Semiótica. Optamos pela tradução "unidades operacionais" para o termo em Inglês "unit operations". Porém, em determinados contextos no livro a tradução mais literal "operações unitárias" também se aplica. Pela clareza, em nossa análise enfatizamos o termo "unidade de significado" ("units of meaning").

[56]Fonte das imagens: primeira (esq., reprodução) - URL: `etc.usf.edu/clipart/70100/70129/70129_waist_dress.htm`; segunda (Crumb) - URL: `www.brainpickings.org/index.php/2012/10/08/r-crumb-illustrates-bukowski`; Amélie Poulain e The Sims (divulgação).

Segundo Bogost, em cada um destes casos, o "encontro casual" se encontra representado em diferentes graus de formalização, desde a rebuscada descrição na poesia do século XIX, até a sua codificação na forma dos algoritmos por trás de personagens em uma simulação computacional.

Convém notar que, em sua análise, Bogost não propõe buscar por formas ou estratégias análogas ou similares entre as diferentes referências (autores, linguagens e suportes), mas sim identificar a maneira como um determinado tema (no caso, o "encontro casual") é representado em cada uma delas, a partir de seus próprios mecanismos expressivos.

Ao longo desta seção, realizaremos uma análise crítica sobre a teoria proposta por Bogost em sua pesquisa. Também adiantaremos alguns dos temas que serão tratados mais adiante nesta tese, como a ideia da tradução (ou comparação) entre linguagens e formas de diferentes naturezas, e o potencial expressivo do engine de um jogo.[57]

De Baudelaire a The Sims

No poema A une passante, escrito no meio do século XIX, Charles Baudelaire descreve a experiência do eu lírico ao deparar-se com uma mulher nas ruas da cidade. Esta é a estrofe final do texto:[58]

> Somewhere else, far from here! Too long! Maybe *never*!
>
> I don't know where you're running, nor do you me,
>
> Oh you who I would have loved! And you who knew it too!

Segundo Bogost, mais do que descrever uma situação específica, o poema expressa uma "unidade de significado", que representa um encontro casual. O texto mostra como as cidades modernas promovem encontros incidentais entre os seus habitantes, e formaliza estratégias para lidar com a alienação da vida moderna.

[57] Optamos por apresentar os trechos de texto em Inglês, de maneira a preservar a forma e ritmo do conteúdo empregado por Bogost em sua pesquisa. Incluímos nossa própria tradução informal para o Português como rodapé em cada caso.

[58] Existem múltiplas traduções para este poema, tanto em Inglês quanto em Português. Algumas das versões diferem substancialmente da tradução de Bogost, apresentada aqui (BOGOST, 2006:76). Segue uma tradução nossa para o Português, a partir da versão em Inglês de Bogost: "Em algum outro lugar, distante daqui! Tempo demais! Talvez *nunca*! / Eu não sei para onde você corre, nem você sabe de mim, / Oh você que eu teria amado! E você que, aliás, sabia!"

Esta próxima poesia, Woman on The Street (1996), de Charles Bukowski, se baseia na mesma temática do encontro casual. Porém neste caso a situação aparece de uma maneira mais familiar e condensada. Segue um trecho:[59]

her shoes themselves
would light my room
like many candles.

she walks like all things

shining on glass,
like all things
that make a difference.

she walks away.

Bogost argumenta que, enquanto em Baudelaire o encontro casual na cidade é uma situação nova, em Bukowski ela já se transformou em uma "unidade de significado cultural" estabelecida. De fato, em Bukowski a linguagem é mais direta e compacta. Muda também a postura do eu lírico, que não parece tentar entender ou resolver a situação - ao contrário, tudo indica que ele aceita a sua inevitabilidade.[60]

Consideremos agora o filme Amélie Poulain, dirigido por Jean-Pierre Jeunet, que narra as tentativas da sua protagonista em orquestrar encontros entre pessoas. Bogost observa que aqui o "encontro casual" se vê de tal forma codificado que o próprio filme funciona como uma metáfora deste processo. A personagem Amélie atua como uma espécie de "designer de encontros". Isto é, mais do que fazer o bem para os outros, ela deseja realizar sua fantasia do "encontro ideal".

A introdução do filme, na qual os personagens principais são apresentados através de cenas curtas, rápidas e objetivas, ilustra a diferença entre o discurso tradicional, mais descritivo e elaborado, para uma forma mais moderna, direta e condensada.

[59]Este trecho da poesia de Charles Bukowski foi obtido da sua reprodução no livro de Bogost (BOGOST, 2006:78). Tradução nossa para o Português: "os próprios sapatos dela / iluminariam o meu quarto / como muitas velas // ela caminha como todas as coisas // brilhando no vidro / como todas as coisas / que fazem a diferença. // ela vai embora."

[60]Outros termos análogos ao conceito de "unidade operacional" empregados por Bogost são "unidade de memória cultural" e "unidade de moeda cultural" (tradução livre para "*unit of cultural currency*").

Por último, Bogost examina a representação da situação do encontro casual na série de jogos The Sims (Maxis, 2000-). Nos jogos, é possível criar uma casa e os seus habitantes (os "Sims"), controlando diversos aspectos das suas "vidas virtuais". Existe uma expansão (módulo adicional que acrescenta funcionalidades e elementos ao jogo) chamada "*Hot Date*" (ou "Encontro Quente"), que contém situações específicas relacionadas a encontros amorosos.

Em comparação com os três casos anteriores, o jogo representa um caso extremo de formalização da situação do "encontro casual". Para Bogost, esta unidade operacional se encontra literalmente codificada no jogo: "Diferentemente de 'A une passante' e 'A Woman on the Street', que oferecem significado poético pelas suas características formais, o significado em *The Sims* vem unicamente do efeito generativo de um grande número de conjuntos de regras codificadas." (BOGOST, 2006:86)[61]

Isto é, em The Sims, a situação do "encontro casual" é apresentada como um **sistema**, não uma experiência fixa ou linear (como é o caso nas poesias e no filme). Trata-se de uma forma de representação única, que só poderia ser realizada desta maneira neste meio. O autor conclui afirmando que The Sims representa o colapso daquela unidade de significado original, herdada de Baudelaire.[62]

Unidades Procedurais de Significado

Bogost traça paralelos entre o seu conceito de "unidades de significado" e a própria essência do processamento computacional, descrevendo a proceduralidade como "a prática de encapsular comportamentos específicos do mundo real em representações programáticas." (BOGOST, 2006:13)[63]

Segundo o autor, qualquer produção humana pode ser pensada como potencialmente procedural. Ele exemplifica a partir do filme The Terminal (Steven Spielberg, 2004), que conta a estória de um homem obrigado a passar diversos meses morando em um aeroporto por conta de conflitos diplomáticos. Para Bogost, o ritmo lento e a estrutura pouco convencional deste filme se deve ao fato de que esta não se trata de uma narrativa tradicional, mas sim de uma **exploração procedural**

[61] Bogost evidencia a interatividade, explicando que este elemento passa a ser possível a partir do momento em que a unidade de significado existe como um sistema, ressaltando que nos poemas isso não é possível. A palavra "generativa" foi escolhida como tradução para o termo "*generative*" pela similaridade com o termo. Outra alternativa é a palavra "gerativa". As duas traduções são válidas, ambas tendo como origem a palavra "gerar". Citação: tradução nossa - texto original: "Unlike 'A une passante' and 'A Woman on the Street', which offer poetic significance by their formal characteristics, meaning in *The Sims* comes solely from the generative effect of numerous codified rule sets."

[62] Em nossa pesquisa, como vimos, os meios e linguagens tradicionais, como o Cinema e a Literatura, são descritos como estáticos (isto é, fixos ou lineares) unicamente no que se refere à materialidade da obra em si, e em oposição às mídias digitais. Não consideramos a abertura de interpretação do receptor. Por exemplo, os personagens de um livro podem se tornar entidades "vivas" no pensamento do leitor, existindo na forma de modelos e sistemas. A diferença no meio digital é que, neste caso, também existem sistemas na própria composição da obra, codificados nos seus algoritmos.

[63] Citação: tradução nossa - texto original: "the practice of encapsulating specific real-world behaviors into programmatic representations."

sobre o tema da "espera". Isto é, The Terminal não deve ser visto como uma narrativa estática e linear, mas sim como um sistema (ou "unidade de significado") explorável, uma experiência muito mais próxima daquela do videogame ou do software.

Bogost observa que o meio digital se distingue por ter estas "unidades operacionais" como **principal** meio de representação. Como vimos, embora a abordagem procedural à expressão e a construção de significados possa ser reconhecida em outras linguagens tradicionais (ideia sugerida por Roman Verostko; p. 37), nas mídias digitais isso acontece de maneira única, e serve de base para a realização deste meio como uma forma expressiva própria.

Ao analisar o processo do design de jogos, o ludologista Jesper Juul identifica duas possíveis abordagens: a primeira, baseada na "progressão" (na qual os desafios se apresentam de maneira linear e fixa, ou predeterminada, pelo designer), e a segunda na "emergência" (na qual o designer cria um sistema do qual as situações de jogo emergem).

> Emergência é a estrutura de jogo primordial, na qual um jogo é especificado como uma pequena quantidade de regras que se combinam e produzem um grande número de variações de jogo, e os jogadores então criam estratégias de design para lidar com elas. É uma situação entre o designer especificar completamente o que pode acontecer, e deixar tudo nas mãos do usuário/leitor/jogador.
>
> Progressão é a estrutura historicamente mais nova, que adentrou o jogo de computador através do gênero da aventura. Nos jogos de progressão, o jogador precisa realizar uma série predefinida de ações de forma a completar o jogo. (JUUL, 2002)[64]

Juul sugere que a abordagem da progressão não é interessante porque impõe uma linearidade que contrasta com a natureza dinâmica e procedural do meio digital, uma posição compartilhada por Bogost: "Há motivos para privilegiar a **emergência** sobre a **progressão** como estratégia de design; a primeira requer inerentemente uma jogabilidade mais configurativa, e portanto a princípio **maximizaria as possibilidades expressivas do meio [digital].**" (BOGOST, 2006:121)[65]

[64]Citação: tradução nossa - texto original: "Emergence is the primordial game structure, where a game is specified as a small number of rules that combine and yield large numbers of game variations, which the players then design strategies for dealing with. [It is] somewhere between a designer completely specifying what can happen, and leaving everything to the user/reader/player. Progression is the historically newer structure that entered the computer game through the adventure genre. In progression games, the player has to perform a predefined set of actions in order to complete the game." (sem numeração de página)

[65]A citação original emprega a palavra "*affordances*", que significa a qualidade de permitir ou favorecer uma ação (neste sentido, tem similaridades com o conceito de agência). Este termo, particularmente comum na área do design, não possui uma tradução direta em Português (alguns autores preferem manter o termo em Inglês). Neste texto a

A expressividade não precisa estar associada à emergência, mesmo no contexto do meio digital. O que Bogost quer dizer é que a expressividade procedural tem características e qualidades únicas a serem exploradas. Isto é, "[o] tipo, não o grau, de emergência é o fator decisivo no potencial expressivo de um sistema complexo." (Idem:151)[66]

Convém notar que, embora o objeto principal de estudo de Bogost seja o videogame, as suas teorias, reflexões e argumentações a respeito do emprego expressivo da proceduralidade nas mídias digitais também se aplicam a outros tipos de produção neste meio.

Um Engine Para Emoções

Vimos que o engine de uma simulação ou sistema trata-se da porção do código que contém a formalização dos seus elementos e comportamentos (p. 47). Uma das características principais do engine é que ele não engloba as instâncias específicas dos elementos que populam o ambiente virtual. Assim, em um simulador de corrida de carros, por exemplo, o engine determina o comportamento dos veículos (incluindo como eles interagem entre si), a maneira como a experiência é apresentada ao interator (o estilo dos gráficos e sons pode ser realista ou estilizado, por exemplo) e as regras que regem a competição em si (como os critérios de vitória e derrota). Mas o engine não contém cada carro ou pista específica, ou informações sobre cada um deles, como gráficos e sons individuais – a princípio este conteúdo existe em outra parte do programa.[67]

O conceito de engine ganhou certa evidência na época em que os jogos de tiro em primeira pessoa (os "*First-Person Shooters*", ou FPS) se popularizaram, através de títulos como Doom (1993) e Quake (1996), ambos da produtora Id Software. A partir deste momento, mesmo o público leigo - os consumidores destes jogos - passaram a compreender as diferentes camadas de tecnologia que existiam por trás destas produções. Esta abordagem modular característica dos engines tem sido cada vez mais comum no desenvolvimento de jogos, mesmo em projetos menores e menos complexos. Atualmente muitas produtoras vendem ou licenciam os engines utilizados em seus jogos como um produto independente, inclusive para outras produtoras. Há ainda os engines independentes ou gratuitos, como é o caso do Blender (1998), especializado em jogos 3D, e o Flixel (Adam Saltsman, 2009), para produções 2D em Flash.

---

palavra foi traduzida como "possibilidades". Citação: grifo nosso; tradução nossa - texto original: "There are reasons to privilege **emergence** over **progression** as a design strategy; the former inherently requires more configurative gameplay and therefore would appear to **maximize the expressive affordances of the [digital] medium**."

[66] Citação: tradução nossa - texto original: "[T]he type, and not the degree, of emergence is the deciding factor in the expressive potential of a complex system."

[67] Na prática, a distinção entre o engine e o restante do conteúdo de uma simulação é essencialmente arbitrária. No código de sistemas mais simples, principalmente de simulações e jogos mais antigos, estes dois aspectos geralmente se encontram integrados. Esta separação se torna ainda mais difusa no caso do conteúdo gerado proceduralmente.

A separação entre o engine e o restante do conteúdo do jogo permite uma metodologia de produção mais eficiente e elegante. Desta forma, programadores lidam apenas com os aspectos técnicos ou práticos, enquanto que os designers, artistas e outros criativos podem se concentrar nos aspectos expressivos ou artísticos, como gráficos, música, enredo e diálogos, por exemplo.

Por outro lado, esta separação também contribui para o que muitos consideram como uma estagnação criativa nos jogos digitais (ou então, ela reflete esta situação). Afinal, a partir do momento em que o artista ou criativo atua de maneira isolada, sem ter contato ou influência nos elementos procedurais, como a mecânica do jogo, o resultado é uma dissonância entre estes diferentes aspectos da experiência.[68]

Em muitos casos este distanciamento não tem um impacto tão significativo no produto final (por exemplo, em determinados gêneros de ação ou simuladores de esporte, nos quais há pouco ou nenhum elemento expressivo ou narrativo). Entretanto, como será visto mais adiante, no caso de designers buscando formas de explorar a expressividade no contexto destes sistemas, a proximidade com os elementos procedurais é essencial.

Na prática, a maior parte dos elementos e aspectos expressivos ou artísticos encontrados em produções digitais, como narrativas e jogos, ainda são predominantemente baseados em conteúdo estático. O engine serve principalmente uma função técnica. Por outro lado, assim como qualquer tipo de formalização ou modelo, o engine inevitavelmente representa uma perspectiva específica sobre uma determinada situação ou conceito.

Por exemplo, considere os jogos no estilo *sandbox* (ou "caixa de areia"), um gênero predominantemente baseado em sistemas. Nestes jogos, o interator tem um alto grau de agência, podendo decidir por conta própria como prefere agir. Entretanto, sempre haverá determinados elementos ou aspectos que são fixados ou predefinidos no sistema.

[68] Existem engines que oferecem uma interface de scripting, através da qual designers podem definir a mecânica do jogo, criando comportamentos e regras sem lidar com a programação em um nível mais baixo. Entretanto, esta prática não é tão comum, e geralmente se restringe a aspectos e elementos específicos do jogo. De qualquer forma, mesmo nestes casos, embora a separação entre o design do jogo e a programação dos seus sistemas seja menor, ela continua existindo.

No simulador de planejamento urbano Sim City (Maxis, 1989-), por exemplo, no qual é possível criar uma cidade e administrar o seu funcionamento, há uma postura política particular associada à lógica por trás do sistema de arrecadação e investimento dos impostos (fig. 12, abaixo). No jogo Grand Theft Auto (GTA; Rockstar Games, 1997-), por sua vez, no qual o objetivo é roubar carros e praticar uma variedade de crimes, qualquer tipo de interação se dá no contexto da violência - esta questão é abordada pelo *machinima* My Trip to Liberty City (Jim Munroe, 2003), criado dentro do GTA, no qual o autor simplesmente "passeia" pelos ambientes do jogo e se nega a participar de atos violentos (fig. 12).[69]

Figura 12 – Sim City (esq.), My Trip to Liberty City (GTA) e Tetris

Nem sempre é claro qual é o significado representado em um sistema. O clássico Tetris (Alexey Pajitnov, 1984), por exemplo, a princípio é apenas um jogo de quebra-cabeças (fig. 12, acima). Mas a simplicidade da sua mecânica, aliada ao alto nível de abstração dos seus gráficos, permite que o jogo seja interpretado a partir de uma infinidade de perspectivas. A pesquisadora Janet Murray, por exemplo, vê o jogo como uma metáfora para a correria da vida moderna.

Mesmo quando existe um consenso relativamente universal sobre o significado de uma determinada simulação ou sistema, nem sempre é possível saber as verdadeiras intenções por trás desta escolha. Por exemplo, o mundo de The Sims é claramente capitalista (o sucesso no jogo está associado à posse de bens materiais), porém é incerto se esta escolha tem um viés crítico ou não.[70]

[69] *Machinima* é o nome dado para uma produção em vídeo baseada em conteúdo gerado a partir da interação com um jogo digital.

[70] Bogost também trata sobre o uso da abordagem procedural como uma forma de conhecer, e como um olhar sobre o mundo. Este é um dos temas tratados na nossa pesquisa anterior de mestrado (FERREIRA, 2011:145).

Esta capacidade que os modelos digitais tem de representar significados pode ser explorada por artistas e criativos buscando se expressar a partir da autoria procedural. Como veremos, a mesma abordagem empregada na criação de um algoritmo para gerar um desenho ou uma composição musical, por exemplo, pode ser aplicada para desenvolver um engine, ou determinados aspectos ou elementos dele (p. 170).

Bogost também chama atenção para a similaridade que existe entre o conceito de engine e o de gênero, no sentido que se usa para a classificação na Literatura e no Cinema, por exemplo. Ambos descrevem em linhas gerais um tipo de objeto cultural, mas sem definir os seus aspectos ou elementos específicos. Entretanto, como vimos, há uma diferença fundamental entre eles: enquanto ambos podem conter múltiplas "unidades de significado", o engine é praticamente **definido exclusivamente** por elas**.**

Por exemplo, no gênero de filmes de ação, as unidades de significado geralmente incluem o herói, determinados conflitos, um interesse romântico e assim por diante. Mas estes elementos, embora essenciais para este gênero específico, não bastam. É necessário que um roteirista, tomando estes elementos como base, crie a narrativa. Por outro lado, em um jogo do tipo FPS, as unidades de significado definem diretamente a experiência, pois estas se encontram formalizadas nos próprios algoritmos que compõem o engine.

Baudelaire x The Sims

Retomado o percurso descrito por Bogost, mostrando a representação de uma unidade de significado desde a literatura no século XIX até as mídias digitais, examinaremos a seguir com maior profundidade a decisão do autor por escolher o jogo The Sims como a versão digital do "encontro casual" encontrado no poema de Baudelaire.

Como vimos, nos poemas e no filme há uma interpretação artística da situação do encontro casual. Em The Sims, por outro lado, o objetivo é diferente - embora o jogo expresse esta mesma ideia, a partir da simulação do comportamento dos personagens, dos movimentos e gestos físicos envolvidos nos seus encontros, não há uma intenção expressiva no design desta experiência.

Em The Sims, os jogadores são os principais responsáveis por tornar a experiência de jogo interessante e divertida, ao construir mentalmente as suas próprias narrativas, atribuindo intenções e

significados por trás do comportamento dos habitantes da simulação. De fato, há relatos de que, durante a produção do primeiro título da série, nos testes de jogabilidade foi detectado que o sistema de Inteligência Artificial empregado para simular os personagens era complexo demais. Tratava-se de um sistema detalhado e realista, porém que confundia os jogadores, que consideravam o comportamento dos personagens inverossímil, ou pouco natural. Por conta disso, o sistema de IA foi simplificado, resultando em uma representação mais caricatural dos personagens. Esta abstração, ou distanciamento do real, foi fundamental para permitir que os jogadores se identificassem com os personagens e o seu comportamento, imaginando uma "vida interna" e narrativas individuais para cada um deles.[71]

De qualquer maneira, The Sims não é, e tampouco pretende ser, uma produção artística. Trata-se de um jogo – seu objetivo principal é de entreter e desafiar.

Seguindo a evolução sugerida por Bogost, desde a *"passante"* de Baudelaire até a simulação do encontro casual em um jogo digital, não ocorre apenas uma condensação, ou formalização, da unidade operacional do "encontro casual". Para o autor, juntamente a este processo, também diminui necessariamente a carga poética (ou artística) no tratamento sobre o tema.

De fato, uma determinada unidade de significado tem a tendência a se tornar familiar com o passar do tempo, levando até mesmo ao surgimento de um certo cinismo em relação a ela. Enquanto que Baudelaire descreve em detalhes seu lamento pelo desencontro, Bukowski aceita a inevitabilidade da situação. Em Amélie Poulain, por sua vez, o tema do encontro casual é tratado quase como um experimento científico.

Entretanto, é importante notar que nestes três casos o "encontro casual" está sendo processado pelo olhar do artista ou criativo, algo que não acontece em The Sims. Sob esta perspectiva, há uma certa discrepância na escolha deste jogo, ao invés de uma produção artística ou expressiva baseada em estratégias procedurais, como exemplo para representar o meio digital neste contexto. Em outras palavras, nos parece que The Sims não serviria como argumento para justificar esta suposta diminuição inevitável da carga poética, em comparação com as demais linguagens analisadas.

[71] A reação dos jogadores à primeira versão de The Sims, descrita neste parágrafo, está relacionada ao fenômeno denominado como "*uncanny valley*" (ou "vale da estranheza"). Trata-se da situação quando a representação de um elemento do mundo físico é realista, porém ainda preserva aspectos artificiais, tornando-se repulsiva. Will Wright também declara ter tomado como referência os estudos de Scott McCloud, teórico e criativo em arte sequencial (ou quadrinhos). McCloud explica que a abstração é necessária para criar empatia com o leitor (ou receptor) - uma representação detalhada demais não oferece espaço para que o receptor possa desenvolver a sua própria interpretação sobre o personagem (MCCLOUD, 2011). Informações obtidas dos documentos de design originais do The Sims: Jefferson - Compilação de documentos publicada por Don Hopkins em 2012. Diversos arquivos em PDF. 1996 (data aproximada).

Reiteramos que, embora The Sims seja predominantemente voltado para o entretenimento, em oposição a uma produção artística, ainda assim o jogo emprega a proceduralidade de maneira expressiva. Como vimos, o jogo pode ser interpretado como um modelo do capitalismo, por exemplo - este conceito é expressado no jogo através da sua mecânica, de maneira procedural.

Outros exemplos similares incluem produções editoriais como Cutthroat Capitalism (WIRED, 2009), mencionado anteriormente, um jogo sobre a pirataria na Somália, e RIOT, um "simulador de manifestações" (fig. 13, abaixo). A intenção nestes casos é de apresentar uma pauta, ou expressar uma ideia.[72]

Figura 13 – Cutthroat Capitalism (esq.) e RIOT

[72]Cutthroat Capitalism foi criado como acompanhamento para um artigo na revista WIRED. RIOT trata-se de um jogo independente, ainda em desenvolvimento.

2.3 Considerações e Esclarecimentos

Vimos que há uma diferença fundamental entre, de um lado, o emprego do meio digital para lidar com conteúdo de natureza estática, e do outro, para lidar com conteúdo procedural - isto é, algoritmos.

De uma maneira geral, em termos quantitativos, a maior parte dos usuários de mídias digitais as empegam como uma maneira de emular meios tradicionais, para produzir, consumir e transferir conteúdo estático, como textos, planilhas, imagens, vídeos, música e assim por diante. Todo usuário de computador lida com aplicativos e sistemas, porém apenas uma minoria programa eles. Da mesma forma, entre os artistas e criativos atuando em mídias digitais, os que utilizam a programação em seu trabalho também são a minoria.

Acreditamos que esta situação pode levar a algumas confusões e mal-entendidos sobre a natureza das mídias digitais, suas características e funcionalidades, bem como sobre as atividades realizadas neste meio, e o conteúdo nele produzido ou exibido (além disso, esta situação também é em parte causada justamente por esta percepção). Embora estas confusões se encontrem principalmente associadas ao público leigo, veremos que em alguns casos elas também são compartilhadas entre profissionais experientes, incluindo artistas e criativos atuando em mídias digitais, bem como críticos e pesquisadores estudando sobre este meio.

Examinaremos algumas destas questões nas próximas páginas, com o propósito de elaborar e definir de maneira mais completa e precisa a expressividade e a autoria procedural, distinguindo esta abordagem de outros empregos das mídias digitais, bem como de outros conceitos e ideias (dentro e fora deste meio).[73]

[73] Todas as considerações apresentadas a seguir se aplicam à expressividade procedural. Além disso, algumas delas também são compartilhadas com a autoria procedural de uma maneira mais ampla (isto é, aquelas práticas criativas sem intenção expressiva). Isso se deve ao fato de que, como vimos, muitas das estratégias criativas empregadas por artistas também fazem parte do repertório de criativos em outras áreas.

A expressividade procedural não se limita a manifestações, aplicações ou conceitos específicos. Como vimos, o uso da programação de computadores por artistas e criativos pode ser aplicada em uma diversidade de áreas e para diferentes fins (p. 25). Entretanto, muitas vezes este conceito é simplificado e associado a manifestações ou instâncias particulares da sua aplicação. Por exemplo, a determinadas **práticas** artísticas (como a Arte Generativa), **estilos** particulares (como a estética minimalista, por exemplo) e mesmo certos **artistas, criativos** e **obras** específicas.[74]

Vimos que o aspecto da **interatividade** também tende a ser associado de maneira incondicional às estratégias procedurais de criação, embora este seja secundário a estas práticas, e nem sempre esteja presente (p. 19). Esta percepção pode ser ilustrada a partir do comportamento do público em exposições de arte e tecnologia, para o qual "[f]azer um gesto na frente de uma obra que não possui esse grau de interatividade já é algo comum." (SOGABE, 2011)[75]

O mesmo pode acontecer com determinadas **ferramentas**. Embora exista uma infinidade de aplicativos, ambientes e linguagens de programação disponíveis para a realização da autoria procedural, algumas tendem a ser priorizadas sobre outras. Este foi o caso do aplicativo Flash no final dos anos 90, e depois a linguagem Processing, bem como outros aplicativos especializados (como aqueles para criação de jogos).

Também é comum a simplificação da abordagem procedural a uma **técnica** específica, como por exemplo o uso do ruído (ou *noise*), ou de geradores de números aleatórios (ou random) em geral, aplicados à produção ou a manipulação de conteúdo. Embora o ruído seja um elemento fundamental nas estratégias procedurais de criação, principalmente em determinadas manifestações mais específicas, esta técnica (ou função) não é essencial a estas práticas.[76]

Dependendo do contexto, a autoria procedural também tende a ser associada a produções de uma determinada **natureza** (por exemplo, visual, sonora, textual, narrativa e cinética), **tipo** (como motion design e jogos) ou em um **suporte** específico - suportes comuns são a tela do computador ou a impressão; porém há inúmeras outras opções, como por exemplo as esculturas de areia produzidas a partir de algoritmos da série Sand As Medium (1997-2002), do francês Jean-Pierre Hébert.

Também existem determinados **conceitos** que tendem a ser associados às estratégias procedurais de criação, apesar de não se tratarem de elementos fundamentais ou mesmo exclusivos a estas práticas.

[74] A Arte Generativa, por exemplo, geralmente se refere especificamente ao uso do algoritmo como meio para gerar algo, como uma imagem ou objeto. Além disso, esta prática tende a estar associada especificamente a processos matemáticos ou sistemas biológicos complexos, explorando as noções de emergência e aleatoriedade.

[75] Sem numeração de página.

[76] Outro exemplo é o uso de sistemas de Inteligência Artificial para modelar o comportamento de personagens em narrativas ou jogos digitais. Apesar dos benefícios evidentes em empregar este técnica para este propósito, como vimos, esta é apenas uma entre as inúmeras possíveis abordagens que podem ser exploradas.

É o caso da emergência (quando um comportamento complexo ou inesperado ocorre a partir de um conjunto de regras simples) e do caos (ou aleatoriedade), por exemplo.

Por fim, embora as estratégias procedurais de criação possam ser aplicadas em diferentes **etapas** do processo criativo, há uma tendência a associar estas práticas à sua ocorrência mais comum: quando a produção final é gerada a partir da execução do algoritmo. Entretanto, como vimos anteriormente, esta abordagem pode ser aplicada a outras etapas do processo criativo, como por exemplo na fase inicial de planejamento. Outro exemplo é o caso da companhia de dança Random Dance, mencionada anteriormente, que emprega estratégias procedurais no processo de elaboração de algumas das suas coreografias - isto é, como ferramenta de auxílio ao processo criativo, mas não diretamente na produção ou exibição de uma obra.[77]

A expressividade procedural não visa resultados objetivos, absolutos ou definitivos. Segundo o filósofo Luigi Pareyson, "a única lei da arte é o critério do êxito." (PAREYSON, 1989:139) Isto é, não existe uma maneira única, ou "correta", de materializar uma determinada proposta poética, ou projeto de criação. A natureza objetiva do meio digital pode sugerir que existe uma solução "mais adequada" ou "otimizada" para representar uma determinada ideia, porém a princípio isso não se aplica à atividade expressiva ou artística, que não se guia por critérios de precisão e eficiência.[78]

Durante o processo criativo de uma maneira geral, e durante o próprio ato de materialização de uma determinada produção, o embate com a linguagem, o meio e as ferramentas empregadas pode gerar novos rumos e contornos para o projeto. Assim, da mesma forma que na Pintura cada novo gesto pode influenciar os gestos seguintes, e na Literatura muitas vezes as palavras guiam o escritor, na expressividade procedural o autor pode amadurecer o seu projeto de criação ao mesmo tempo em que constrói e executa os algoritmos da sua composição.

Este caráter dinâmico e aberto se reflete de maneira particularmente evidente na prática do Live Coding, na qual o artista cria e edita o código ao mesmo tempo em que o programa é executado, muitas vezes exibindo o processo para o público em uma projeção, como uma performance.

[77] Trataremos sobre o emprego da abordagem procedural nas diferentes etapas do processo criativo na próxima seção (p. 84).

[78] A rigor, mesmo quando não há uma intenção expressiva por parte do autor, a implementação de sistemas envolve escolhas arbitrárias de modelo ou recorte sobre o objeto a ser representado.

A expressividade procedural não é necessariamente complexa ou rebuscada. Do ponto de vista técnico, as mídias digitais são mais complexas do que outras máquinas ou ferramentas mais simples. Entretanto, isso não significa que todo conteúdo em mídias digitais, ou que qualquer tipo de emprego ou produto deste meio, seja necessariamente complexo.

Além disso, o potencial expressivo de um algoritmo não se encontra necessariamente relacionado à sua complexidade, da mesma maneira que uma pintura não é necessariamente mais expressiva se o artista utilizar uma maior variedade de cores ou criar um número maior de traços ou elementos.

A expressividade procedural não se refere a criações de natureza essencialmente estática. Como vimos, as estratégias procedurais de criação se distinguem das demais práticas em mídias digitais por serem realizadas através da programação - isto é, por se basearem fundamentalmente na proceduralidade deste meio. É a diferença entre os traços gerados pelo algoritmo na obra Algorithmic Poetry for a Three Story Wall (Roman Verostko, 2011) e uma ilustração produzida manualmente por um desenhista em um aplicativo gráfico.[79]

Outra forma artística que tende a ser confundida com a expressividade procedural é a situação na qual a obra final é o algoritmo em si, ao invés do produto ou resultado da sua execução (COX; WARD, 2008). Embora uma produção como essa tenha fortes pontos de contato com a autoria procedural, ela aproxima-se mais de práticas como a Arte Declarativa.

Lembrando que uma produção de natureza procedural também pode ser materializada em um suporte estático. Este é o caso do trabalho de Verostko, mencionado acima, que resulta em uma impressão. Da mesma maneira que o fato de uma ilustração ter sido criada em meio digital não significa necessariamente que trata-se de "arte digital" (e tampouco Arte Algorítmica), uma obra criada a partir de algoritmos e materializada em uma forma física não perde a sua natureza procedural.[80]

[79] A obra de Verostko citada aqui faz parte da composição que ilustra a capa desta tese (parte esquerda da imagem).

[80] Além disso, uma determinada produção pode ser híbrida, apresentando tanto elementos procedurais quanto estáticos - na prática, este é o caso da maior parte das obras envolvendo elementos procedurais. Como veremos, também convém considerar que a existência de elementos procedurais em uma determinada obra não significa que a expressividade predominante desta produção seja de natureza procedural. Finalmente, lembramos que os procedimentos (os algoritmos) são eles próprios compostos fundamentalmente de elementos estáticos (os caracteres em um código, por exemplo), a partir dos quais a proceduralidade emerge (após a compilação e execução).

A expressividade procedural não se refere ao emprego do computador para fins unicamente práticos. Embora esta seja uma questão evidente, convém reiterar que a expressividade procedural refere-se especificamente aos casos em que a intenção de quem cria é artística ou expressiva. Isso exclui o uso da programação para propósitos práticos (por exemplo, para desenvolver um aplicativo utilitário), incluindo pela sua capacidade de automatizar tarefas.[81]

Note que esta distinção se dá predominantemente a partir da intenção do autor, principalmente no caso de determinadas situações limite. Por exemplo, dois artistas podem criar um algoritmo similar para gerar o desenho de uma árvore, porém um pode fazê-lo por motivos práticos (talvez para economizar o tempo ou esforço necessários para desenhar cada uma das folhas individuais), e o outro por motivos expressivos (por exemplo, para produzir uma estética particular).

Por conta disso, também não faz sentido considerar a expressividade procedural como sendo uma alternativa "superior" ou "mais prática" às estratégias tradicionais, uma vez que estas tratam-se de abordagens de natureza fundamentalmente distintas. Pelo mesmo motivo, também não faz sentido considerá-las como uma abordagem "inferior".

Vimos que a expressividade procedural compartilha de estratégias criativas que podem ser empregados para atividades e fins não expressivos. O mesmo acontece com qualquer outro meio ou linguagem. Por exemplo, muitas das técnicas e ferramentas utilizadas para pintar uma parede para fins práticos também podem ser empregadas para uma produção com fins expressivos. Por conta disso, nem todos os objetos de estudo tratados nesta pesquisa são de natureza artística ou expressiva. Independentemente disso, o nosso foco se encontra na situação particular na qual estas estratégias procedurais são empregadas com intenção expressiva.[82]

[81] O emprego do computador na arte unicamente para automatizar tarefas de certa forma se localiza entre a forma expressiva tradicional e aquela própria do meio digital. Tratamos sobre este tema anteriormente (p. 37).

[82] Por exemplo, anteriormente, de forma a exemplificar os mecanismos do engine de física, foi feita uma análise do jogo Angry Birds, uma produção voltada para o entretenimento (p. 47). Neste jogo as qualidades artísticas ou expressivas não são predominantes, entretanto o exemplo serve de base para a compreensão do potencial expressivo das estratégias procedurais de criação.

2.4 Metodologia de Análise (uma proposta)

> Quem está fazendo os desenhos? Quem é o responsável?
> O programa é um artista? **Que parte disso tudo é arte?**
>
> (COHEN, 1982:4)

Em nossa pesquisa anterior elaboramos uma metodologia de análise, que empregamos para identificar, investigar e descrever as obras, práticas, artistas, criativos e outros elementos e aspectos relacionados ao emprego criativo do meio digital. Esta metodologia consiste na análise das diferentes **influências e contribuições** envolvidas ao longo do processo criativo, desde a formulação da proposta ou projeto na mente do criador, passando pelos meios e ferramentas físicas e intelectuais empregados, até a materialização da obra ou produção final.[83]

A elaboração desta metodologia foi um elemento fundamental daquele estudo, não apenas do ponto de vista prático, por ter contribuído para consolidar a abordagem que adotaríamos nas análises, mas também como uma maneira de apresentar este processo no contexto mais amplo da pesquisa. Empregamos ela novamente nesta pesquisa, adaptada e reestruturada com base nos novos desenvolvimentos, e recontextualizada de acordo com as particularidades do recorte deste estudo.

Influências e Contribuições

Ao tratar sobre o artista que atua em mídias digitais, Arlindo Machado fala sobre a noção dos **graus de contribuição** dos diferentes elementos envolvidos no processo criativo (MACHADO, 1993). Tomamos esta ideia como ponto de partida para desenvolver uma classificação, apresentada na página seguinte, representando as diferentes **camadas** envolvidas no processo criativo em mídias digitais. Note que, como o propósito desta classificação é de compreender as práticas criativas de uma maneira abrangente, as camadas vão desde as mais fundamentais, relativas ao mundo físico (incluindo o corpo e mente do criador), passando pela sociedade e as ciências, até as ferramentas simples, mecânicas e eletrônicas, culminando nas mídias digitais.

A intenção desta classificação é de fornecer uma base para identificar, analisar e localizar as diferentes influências e contribuições sobre cada etapa e elemento envolvido no processo criativo.[84]

---

[83] Epígrafe: grifo nosso; tradução nossa - texto original: "Who is making the drawings? Who is responsible? Is the program an artist? **What part of all this is art**?"

[84] A classificação inclui elementos e temas mais gerais (como as ferramentas simples, por exemplo) e de outras áreas (como a Física e a Química). Estes devem ser considerados principalmente como contexto e suporte para a análise das práticas criativas.

Classificação: Camadas de Influência

1. Natureza		
	Fenômenos físicos	gravidade, eletromagnetismo, eletricidade etc.
	Propriedades da matéria	composição química, resistência, densidade etc.
	Aspectos ambientais	em grande escala - clima, geografia etc.
2. Ser humano		
	Mente	lógica inata,mecanismos internos
	Corpo	estrutura, aspecto e funcionamento do corpo
3. Sociedade		aspectos sociais, culturais (arte, linguagem), econômicos etc.
4. Ciências		biológicas, humanas e exatas
5. Ferramentas		
	Ferramentas simples	e instrumentos artesanais - pincel, violino etc.
	Máquinas mecânicas	prensa, máquina fotográfica, filmadora etc.
	Máquinas eletrônicas	vídeo, fax etc.
6. Mídias Digitais		hardware, software etc.

Devido à abrangência desta classificação, ela permite relacionar e localizar as estratégias procedurais de criação em relação às demais manifestações artísticas e criativas, tanto em meio digital quanto de formas expressivas tradicionais, e mesmo a outras atividades humanas mais práticas (resgataremos este tópico no próximo capítulo, ao comparar o processo criativo na produção de narrativas e jogos digitais com aqueles de outras formas e meios tradicionais).

A estrutura da classificação é relativamente simples: os itens numerados à esquerda apresentam o nome das seis camadas principais (em alguns casos também são listadas subcamadas), e à direita há uma breve descrição de cada camada ou subcamada, incluindo exemplos.

Esta é apenas uma entre as inúmeras perspectivas possíveis sobre os processos criativos em meios digitais. O objetivo não é buscar uma classificação definitiva, mas sim as reflexões que decorrem da sua formulação e aplicação.

Nesta classificação, de uma maneira geral as camadas no topo (as primeiras) são mais fundamentais do que aquelas mais abaixo (as últimas). Além disso, as camadas mais fundamentais geralmente fazem parte ou estão relacionadas de alguma maneira a aquelas que a seguem, e que as precedem historicamente (por exemplo, o corpo humano existe independentemente do computador; mas o contrário não é verdade).

A influência da camada **Natureza** e da subcamada **Corpo** (uma das subdivisões da camada Ser humano) é muito mais direta e significativa nas atividades humanas que envolvem lidar com o mundo físico, como é o caso de um escultor, por exemplo. No caso de atividades predominantemente intelectuais ou simbólicas, como a Literatura, a Matemática, a Música e as práticas em mídias digitais, estes elementos tem uma influência menor.[85]

A **Mente** (parte da camada Ser humano) é onde se dá a construção intelectual da proposta do artista ou criativo. Como mencionado acima, no caso das estratégias procedurais de criação, o próprio ato de programar um computador pode ser interpretado como predominantemente intelectual (apesar de envolver o mundo físico), por tratar-se de uma atividade fundamentalmente simbólica.

As demais camadas diferem das duas primeiras no sentido em que representam criações ou construções humanas, tanto intelectuais quanto físicas. A camada **Sociedade** representa os conceitos,

[85] Cada uma das atividades citadas neste parágrafo tem uma determinada proporção de influência da camada mencionada, por diferentes motivos. A Música, por exemplo, pode depender mais da fisicalidade do que a Literatura, nos casos em que as características e limitações dos instrumentos envolvidos são um fator relevante. O meio digital, por sua vez, apesar da sua natureza essencialmente simbólica, tende a ser empregado para emular meios e linguagens tradicionais, incluindo as suas características e limitações (como o comportamento de um pincel no Photoshop, por exemplo). Ao simular a materialidade do mundo físico, o computador de certa forma também pode herdar as suas influências e contribuições fundamentais.

convenções e ideias que moldam as atividades do indivíduo no contexto social (ou sob o efeito dele). A linguagem se encontra nesta camada, e sua influência nas ações humanas é evidente, principalmente no aspecto intelectual. O meio digital é particularmente influenciado por esta camada, uma vez que a lógica do computador se baseia fundamentalmente no pensamento ocidental, pragmático e estruturado, e nas suas convenções (TEDRE; EGLASH, 2008) - por exemplo, isso se reflete no fato de que a maioria das linguagens de programação tem a estrutura e lógica similares à da linguagem verbal.

O mesmo pode ser dito das **Ciências**, a camada seguinte, uma vez que a tecnologia por trás do computador se baseia em uma infinidade de disciplinas, desde as leis da Física e Química nas placas e circuitos até a Lógica das teorias da computação que guiam os processos da máquina. A influência da Ciência também é evidente nas Artes Plásticas (na forma de uma determinada técnica ou material, por exemplo), no desenho (através de conceitos da geometria, por exemplo) e na manifestação particular da Arte Generativa baseada em teorias da genética ou da biologia.

A camada das **Ferramentas** se refere às diferentes materializações de técnicas e saberes científicos na forma de objetos, que podem ser empregados para a realização de atividades específicas. Um exemplo de ferramenta simples é o compasso utilizado para o desenho, que pode ser considerado como uma materialização física da descrição paramétrica de um círculo.

As máquinas mecânicas, por sua vez, apresentam uma maior complexidade. Por exemplo, a máquina fotográfica é um dos primeiros casos da convergência de um grande conjunto de saberes, técnicas e ciências em um único dispositivo de representação. Seu funcionamento é influenciado pela lógica e características de uma variedade de disciplinas, como a Química, a Ótica e a Mecânica, entre muitas outras.

Desta camada em diante, os avanços técnicos são exponenciais, primeiro com o surgimento das máquinas eletrônicas, e finalmente as **Mídias Digitais**. Nos computadores, o aumento de complexidade se deu de uma maneira tão significativa que esta mudança, a princípio unicamente qualitativa, representou também um salto qualitativo. O meio digital, como vimos, se diferencia das demais máquinas e ferramentas - bem como de qualquer linguagem e forma tradicional anterior - pela sua capacidade de executar processos e algoritmos (isto é, seu aspecto procedural).

Perspectivas de Análise

Nossa proposta de metodologia de análise tem como objetivo lançar um olhar informado e abrangente sobre as práticas criativas em mídias digitais, sem enfatizar um aspecto específico - por exemplo, os elementos técnicos - acima dos demais. A intenção é fornecer uma ferramenta teórica para investigar o processo criativo do ponto de vista do artista ou criador, a partir das implicações e significados associados aos mecanismos e características deste meio, e no seu potencial expressivo.[86]

Em paralelo à classificação de influências e contribuições, apresentada anteriormente, a metodologia envolve também uma série de perspectivas de análise pelas quais é possível investigar a criação digital, a partir: (1) dos diferentes elementos e aspectos que fazem parte de uma produção; (2) das etapas do processo criativo; (3) da intenção e das contribuições dos artistas e criativos envolvidos; e (4) de determinados contextos, pontos de vista e recortes.

Elementos e Características da Obra

Nesta perspectiva de análise, são tratados individualmente os diferentes elementos e características de uma determinada obra ou processo criativo, bem como as influências e os relacionamentos entre eles, e o seu papel na proposta ou projeto de criação.

Por exemplo, o aspecto material pode ter diferentes influências sobre uma determinada produção, de acordo com a natureza da sua proposta. Assim, a carga poética de uma obra literária geralmente não é muito influenciada pelas ferramentas empregadas na sua criação, ou pelo formato escolhido para a sua materialização; porém, a expressividade de uma escultura geralmente é influenciada fortemente pela matéria-prima empregada na sua produção.

No caso das estratégias procedurais de criação, uma análise a partir desta perspectiva permite verificar a carga expressiva dos diferentes elementos procedurais de uma determinada obra. Por exemplo, uma determinada produção pode ter um alto grau de intensidade procedural, sem que este elemento seja explorado expressivamente, e vice-versa.

[86] Os aspectos técnicos são relevantes às nossas investigações, porém sempre a partir da sua relação com o aspecto expressivo.

Etapas do Processo Criativo

Além da investigação da obra ou produto final, a análise das diferentes etapas do processo criativo também permite perspectivas importantes para a compreensão de uma determinada prática artística. Ao examinar os elementos e gestos envolvidos, desde a formação da proposta de criação, passando pelas ferramentas empregadas, até a obra ou produção final, também é possível identificar a origem de muitas das influências e contribuições envolvidas.

A abordagem procedural pode ser realizada ou aplicada em diferentes momentos do processo criativo:

1. a **obra final** pode ser de natureza procedural (por exemplo, uma instalação, ou uma produção de Arte de Software);[87]
2. a criação procedural pode ser um **estágio intermediário** no processo - o autor cria um programa que, por sua vez, gera a obra final;[88]
3. a estratégia procedural pode ser empregada como **ferramenta auxiliar** ao processo criativo, mas não diretamente na produção da obra.[89]

Note que a divisão entre estes três momentos não é rígida - por exemplo, a primeira situação pode ser vista como um caso particular da segunda, se for feita a distinção entre o programa em execução, de natureza procedural, e seu algoritmo, de natureza estática (nas notas de rodapé desta página são mencionadas algumas perspectivas alternativas).[90]

Intenção e Contribuições

A identificação da autoria dos elementos expressivos e artísticos que definem ou compõem uma determinada obra ou produção nem sempre é clara. Nas mídias digitais, como vimos, esta questão é ainda mais evidente, por conta da complexidade deste meio.

Considere por exemplo o emprego de aplicativos gráficos, como o Photoshop (Adobe), por parte de desenhistas e ilustradores. Este trabalho envolve o uso intenso de uma variedade de ferramentas e

[87] Neste caso, embora a poética seja de natureza procedural, a criação em si segue um processo estático - isto é, a digitação do código (a não ser que o algoritmo seja ele próprio gerado a partir de um programa - uma situação complexa e pouco comum, envolvendo dois níveis de metacriação).

[88] Neste caso, geralmente o produto final é de natureza estática, como a imagem no caso de Aaron (Harold Cohen), por exemplo. Note que o conteúdo gerado pode ser ele próprio de natureza procedural, como na situação mencionada na nota de rodapé anterior.

[89] Por exemplo, é o caso do aplicativo empregado pela companhia de dança Random Dance, mencionado anteriormente (p. 31).

[90] A expressividade procedural também pode servir exclusivamente como inspiração para uma determinada obra. Por exemplo, um músico tradicional pode criar uma composição tomando a estética da Arte Generativa como inspiração e referência, porém não empregar ele próprio estas estratégias. Não consideramos que estes casos se configurem como estratégias procedurais propriamente ditas.

funções do aplicativo, como os diferentes tipos de "pincéis", efeitos e filtros (como o *blur*, que causa um desfoque na imagem, ou a aplicação de sombras).

Mesmo no caso do artista ou criativo atuando a partir de estratégias procedurais de criação, este emprega funções, estruturas lógicas e bibliotecas de uma ferramenta ou linguagem de programação preexistente. Alguns criativos se dedicam a criar as suas próprias ferramentas, engines e mesmo linguagens de programação, mas ainda assim camadas inferiores de abstração continuam existindo (como veremos mais adiante na tese).

Este processo significa que a autoria se desloca de maneira significativa, passando a envolver, além do autor propriamente dito, os criadores e desenvolvedores das ferramentas, linguagens e sistemas empregados (direta ou indiretamente). Da perspectiva do autor, a contribuição expressiva que cada um destes indivíduos terá sobre a obra final depende de uma série de fatores, como por exemplo a sua intenção ao usar a ferramenta, sua consciência sobre a natureza desta contribuição, seus significados e as implicações do seu uso.

Contexto, Ponto de Vista e Recorte

A análise de uma determinada produção ou processo criativo também pode ser realizada em diferentes contextos e pontos de vista, de acordo com o tipo de investigação sendo feita, seus objetivos e recorte.

Por exemplo, uma determinada produção musical pode ser analisada como composição (uma atividade criativa predominantemente intelectual), como performance (envolvendo elementos externos à mente, como o corpo e mesmo o ambiente) ou mesmo do ponto de vista da linguagem (isto é, contextualizada em relação às convenções da notação musical).

As práticas artísticas em meio digital, como vimos, também podem ser consideradas como predominantemente intelectuais (nos casos em que a proposta independe de um suporte específico de atualização), neste sentido aproximando-se do tipo de abordagem adotado na Arte Conceitual ou Declarativa, por exemplo.

Além disso, assim como existe uma infinidade de fatores que influenciam as diferentes etapas e elementos do processo criativo, e múltiplos pontos de vista possíveis sobre eles, há uma quantidade ainda maior de combinações entre todas estas possíveis situações.

Assim, por exemplo, um artista ou criativo pode criar sob a influência de todas as camadas da classificação, porém em proporções reduzidas, outro pode ter seu trabalho predominantemente influenciado

por apenas um dos fatores citados na classificação, mas de maneira extremamente determinante, e assim por diante.

Outro fator que pode ser considerado na análise de uma obra ou processo criativo é que nem sempre o efeito de uma determinada influência ou contribuição é proporcional à sua visibilidade no contexto da expressividade da obra final. Por exemplo, na Arte Generativa geralmente a estética da obra denuncia a sua origem procedural, muitas vezes de maneira intencional. Por outro lado, na área dos efeitos especiais para o Cinema o oposto tende a acontecer. Isto é, os elementos ou efeitos computacionais tendem a ser criados de forma a produzir uma ilusão, realista ou fantástica, sem chamar atenção para os seus próprios mecanismos.[91]

[91] Um caso extremo de tornar visível os mecanismos do processo criativo é a prática do *Live Coding*, mencionada anteriormente, na qual a relação entre o algoritmo produzido pelo artista e as imagens e sons gerados se revela de maneira clara e direta (p. 75).

3 Playing Beowulf (Estudos Práticos)

Neste capítulo apresentaremos os estudos práticos realizados junto ao projeto Playing Beowulf (ou "Jogando Beowulf"), no qual trabalhei como parte da equipe de pesquisa e desenvolvimento do MissionMaker, um software para a criação de jogos e narrativas digitais baseadas no poema épico Inglês Beowulf. Estes estudos foram realizados durante o período sanduíche desta pesquisa, na University College London (UCL), no Reino Unido.

Embora o MissionMaker também ofereça ferramentas para a criação de conteúdo de natureza estática, minha contribuição ao projeto se deu principalmente sobre as funcionalidades do aplicativo referentes às estratégias procedurais de criação. Estes estudos práticos também contribuíram para o desenvolvimento de uma nova versão do aplicativo, principalmente na definição do novo paradigma de autoria e fluxo de trabalho, e das novas funcionalidades da ferramenta.

No contexto desta tese, estes estudos práticos tiveram como objetivo principal servir de base para a nossa investigação principal, sobre a expressividade procedural.

Neste capítulo, primeiro trataremos **sobre o projeto** Playing Beowulf (p. 91), apresentando o poema, o aplicativo e a oficina piloto, na qual se baseou a maior parte dos estudos práticos.

Em seguida, apresentaremos a **análise dos jogos** produzidos pelos participantes da oficina (p. 103). As análises tiveram como foco a forma como os diferentes elementos, situações e personagens do poema original foram reinterpretados e representados pelos participantes através das estratégias procedurais de criação oferecidas pelo MissionMaker.

Depois, apresentaremos a **análise do aplicativo** em si (p. 149), com ênfase na maneira como a prática da autoria procedural é implementada e disponibilizada ao designer nos mecanismos e funções do MissionMaker. Elaboraremos sobre como estes recursos são organizados dentro da gramática particular deste aplicativo, a partir do ato da composição. Também examinaremos este processo a partir das suas intersecções com outras formas e linguagens tradicionais, buscando paralelos e comparando os recursos e estratégias compartilhadas entre estas práticas. Para esta análise, além da nossa própria metodologia de análise (apresentada anteriormente), também empregaremos uma outra abordagem denominada Análise Multimodal.

Por fim, realizaremos **algumas considerações** sobre estes estudos práticos (p. 203), e apresentaremos brevemente a nova versão do aplicativo - MissionMaker2 (em fase alpha) -, bem como alguns dos **futuros desenvolvimentos** previstos para as suas próximas iterações (p. 207).

3.1 Sobre o Projeto

Playing Beowulf é um projeto realizado em parceria com a Biblioteca Britânica, voltado ao ensino da programação para crianças e adolescentes, através da criação de jogos. Ele faz parte do DARE Collaborative, um grupo de pesquisa baseado no London Knowledge Lab (LKL), no Institute of Education da University College London (IOE UCL).

O projeto envolve o desenvolvimento do MissionMaker, um aplicativo para criação de jogos baseados no poema anglo-saxão Beowulf. Esta ferramenta é voltada para o uso em escolas e universidades, bem como museus e bibliotecas, tanto como uma abordagem alternativa à leitura do poema original, quanto como uma forma de aprender sobre criação de jogos, programação e autoria procedural de uma maneira mais ampla (BURN, 2010).[92]

Os estudos práticos apresentados neste capítulo foram desenvolvidos dentro deste projeto, no qual trabalhei integrando a equipe de pesquisa e desenvolvimento, sob orientação do Prof. Andrew Burn e da Dra. Alison Gazzard (IOE UCL).[93]

Convém reiterar que, embora esta parte da pesquisa tenha sido realizada junto a um projeto com viés educacional, e aproveitando-se de inúmeros aspectos e elementos viabilizados por conta deste contexto, o nosso foco de investigação não se encontra no aspecto pedagógico, mas sim nos aspectos criativos e expressivos da autoria procedural.

[92]Playing Beowulf é parte do programa Digital Transformations, financiado pela Arts and Humanities Research Council UK. Outros parceiros incluem a University of Sydney e o espaço GameCity - URL: `gamecity.org`; Nos referimos ao projeto pelo título original em Inglês ao invés da tradução em Português, "Jogando Beowulf". O nome completo do projeto inclui o subtítulo "Gaming the Library". A referência à biblioteca ("library") se deve à ideia de propor releituras de textos literários. O termo "gaming", além de fazer referência ao jogo digital, também se relaciona ao ato de desafiar os mecanismos ou lógica de um determinado sistema.

[93]Esta pesquisa "sanduíche" teve duração de um ano (ou três "termos" acadêmicos no Reino Unido), de Outubro de 2014 a Setembro de 2015, e foi financiada pela bolsa do programa Ciência Sem Fronteiras (CNPq). Minhas atividades durante este período incluíram, além da minha participação ativa no projeto: (1) sua apresentação na conferência Storytelling in the Medieval World (realizada em Abril de 2015 na UCL), juntamente com as colegas Theo Bryer e Jane Coles (IOE UCL); (2) a produção de dois artigos baseados na oficina piloto (FERREIRA, 2014a e 2014b); e (3) um texto mais longo explorando os aspectos teóricos do estudo relacionados à Análise Multimodal (FERREIRA, 2015).

3.1.1 Beowulf

Beowulf é um poema épico Inglês escrito entre os séculos VIII e XI por um autor anônimo, com base em uma estória construída a partir da tradição oral. Trata-se de um dos textos mais importantes escritos no que hoje denomina-se como Inglês arcaico (ou "Old English"). Seu manuscrito original, queimado parcialmente em um incêndio em 1731, é conservado atualmente na Biblioteca Britânica. A figura abaixo (fig. 14) ilustra a primeira página do manuscrito, incluindo um detalhe ampliado, à direita na imagem (reprodução em escala de cinza).

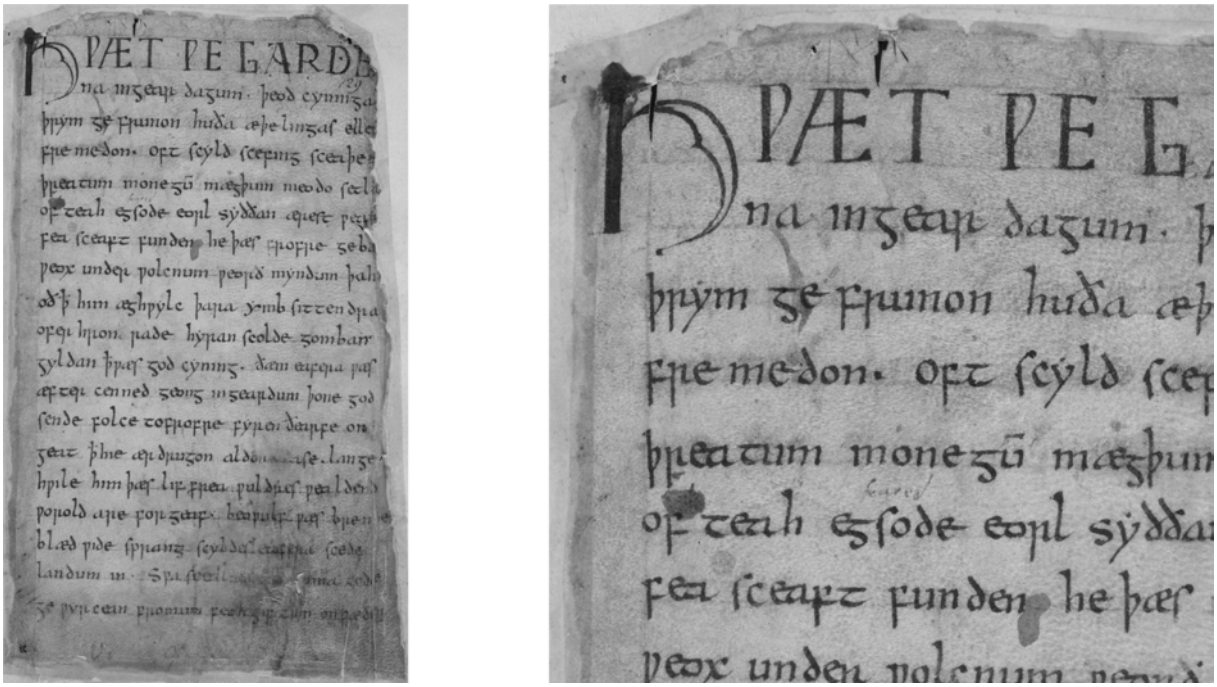


Figura 14 – Beowulf: primeira página do manuscrito original (dir.: detalhe)

O poema conta a estória do herói Beowulf que, atendendo ao pedido de ajuda de Hrothgar, o rei dos Danos, vai até o seu reino para protegê-lo de um monstro chamado Grendel. O enredo completo do poema é longo e complexo, e envolve inúmeros personagens, mas para o propósito desta pesquisa iremos nos concentrar apenas em alguns destes elementos, de forma a contextualizar os estudos práticos apresentados aqui.[94]

[94] A versão do poema utilizada como referência principal no projeto Playing Beowulf é a tradução para o Inglês moderno de Frances B. Gummere, de 1909, uma vez que a compreensão do texto original é difícil e demanda conhecimento do Inglês arcaico. A referência em Português empregada nesta tese é a tradução de Ary Gonzalez Galvão, de 1992, que inclui introdução e notas.

De forma a familiarizar o leitor com a linguagem empregada no poema, segue uma reprodução dos seus primeiros versos, da tradução para o Inglês moderno de Frances B. Gummere (GUMMERE,1909):[95]

> *LO, praise of the prowess of people-kings*
> *of spear-armed Danes, in days long sped,*
> *we have heard, and what honor the athelings won*
> *Oft Scyld the Scefing from squadroned foes,*
> *from many a tribe, the mead-bench tore,*
> *awing the earls. Since erst he lay*
> *friendless, a foundling, fate repaid him [...]*

Mais adiante neste capítulo apresentaremos sete jogos, criados pelos participantes da oficina piloto Playing Beowulf, representando determinadas situações e cenas do poema, bem como alguns dos seus principais personagens e temas. Em todos os jogos, o interator é colocado no papel de **Beowulf**, o protagonista da estória, um herói extremamente forte e poderoso. O antagonista principal na maioria dos jogos é o monstro **Grendel**, ou então sua mãe (geralmente referenciada como "**mãe de Grendel**").

Os locais nos quais a ação se passa nestes jogos incluem o castelo dos Danos e **Heorot**, o grande salão de banquetes, alvo de ataque dos monstros. Outras áreas importantes do poema representadas nos jogos são o **pântano** (ou lago), no qual Beowulf mergulha para chegar até a caverna onde Grendel e sua mãe se escondem, além do próprio **covil** dos monstros. Alguns dos jogos também representam os restos mortais de **Æschere**, um dos guerreiros dos Danos, que no poema é morto pela mãe de Grendel em vingança ao assassinato do seu filho.

Qualidade Sistêmica

Beowulf é mais popularmente conhecido pelas aventuras do seu protagonista, e seus embates com os monstros e dragões. Entretanto, um dos aspectos mais fortes desta obra reside na construção detalhada do seu universo ficcional, incluindo elementos da Escandinávia do século V, desde guerras e lendas até jogos de poder e relacionamentos amorosos.

A riqueza do universo ficcional de Beowulf torna este texto particularmente aberto a reinterpretações criativas e, como veremos, especialmente apropriado para a tradução na forma de sistemas e regras

[95] Sem numeração de página.

em mídias digitais. Podem ser identificados paralelos em diversos níveis, desde o mais superficial - ou evidente -, do enredo e das situações retratadas, passando pela estrutura do poema e os temas sobre os quais a estória se baseia, até a própria estrutura e lógica da linguagem empregada, incluindo as especificidades do Inglês arcaico.

Mais do que os eventos e personagens particulares que compõem a estória de Beowulf, a principal força do poema vem dos temas e princípios nele contidos, nas ideias de coragem, lealdade e honestidade. Como explica o tradutor Ary Gonzalez Galvão, o poema é "valioso como grande documento social sobre o ideais de uma raça e de uma nação." (GALVÃO, 1992:9) Galvão explica que o "bardo anônimo" autor do poema "conhecia muito bem a vida da corte dos castelos medievais, os códigos rígidos da cavalaria, as lendas e tradições do seu povo [...]" (Idem).

De certa forma qualquer produção artística ou objeto cultural tem o potencial de refletir as estruturas e formas mais amplas nas quais se baseiam ou estão inseridas, bem como aspectos e características da época e contexto da sua concepção. Por outro lado, este aspecto tende a ser mais evidente em determinados gêneros e formas, como é o caso de um poema épico como Beowulf, transmitido a partir da tradição oral

Consideremos os paralelos entre os temas e estrutura do poema e aqueles de um jogo. Segundo Galvão, em Beowulf a forma da narrativa segue a clássica trajetória do herói - ritos de passagem, separação, iniciação, retorno, morte do herói (volta à condição humana) (GALVÃO, 1992). Esta trajetória vai além de simplesmente definir uma estrutura para a narrativa. Trata-se de uma forma que se define em torno da figura central de um herói, cujas ações determinam o desenrolar dos eventos, e a partir do qual a perspectiva do público diante da estória é estabelecida - isto é, de maneira similar ao avatar (a figura que representa o jogador em uma simulação) em uma narrativa ou jogo digital.

Em Beowulf também existe uma lógica claramente definida na contraposição clássica entre o bem e o mal. Em um lado, o conhecimento, a beleza e a honra dos guerreiros liderados por Beowulf; no outro, a ignorância que Grendel e sua mãe tem sobre estes valores. Esta dualidade funciona como uma diretriz com base na qual o mundo de Beowulf é construído, desde os elementos mais amplos do enredo e dos temas, até aqueles mais pontuais e específicos.

Isso se reflete nas duas localidades principais do poema, na contraposição entre o salão de banquetes Heorot, representando alegria e segurança, e o pântano que serve de covil para os monstros, representando as trevas e a ameaça aos ideais humanos. Esta mesma lógica se reflete de diferentes formas na personalidade de cada personagem, e consequentemente nos eventos que compõem a estória - por exemplo, na decisão de Beowulf em lutar contra Grendel (o bem contra o mal). Mas esta mesma dualidade pode levar a situações mais complexas e sofisticadas.

Por exemplo, na primeira grande batalha do poema, entre Beowulf e Grendel, o guerreiro dispensa a espada e decide enfrentar o monstro com suas próprias mãos. Este gesto representa a existência de uma camada adicional de complexidade neste personagem, que vai muito além da simples ideia da batalha entre o bem e o mal.

Esta decisão específica de Beowulf em não utilizar uma espada contra Grendel, baseada na sua percepção do fato de que o monstro ignora os códigos de conduta do guerreiro, representa de maneira prática os princípios do protagonista. Segue o trecho (GALVÃO, 1992:57):

> E Beowulf, orgulhoso, desarmou-se do escudo,
> despiu-se da cota de malha, entregou ao escudeiro
> a espada - a melhor de todas as lâminas - e pediu-lhe
> que guardasse os apetrechos de guerra; então
> pronunciou estas palavras - a bravura de um grande
> guerreiro - antes de se deitar: 'Imagino que minha
> força no embate seja pelo menos tão poderosa quanto
> a de Grendel. Não o matarei com minha espada -
> tarefa tão simples essa! Grendel não tem a
> habilidade em combate de responder na arte
> da luta e de vibrar seus golpes no meu
> escudo - perspicaz que ainda possa ser nesses
> ataques noturnos. Não, à noite não usarei
> arma alguma, se desarmado ousa
> enfrentar-me em combate. O Senhor em Sua
> sabedora há de honrar - o Todo-Poderoso -
> aquele que mais apto Lhe parecer'.

Em qualquer forma expressiva tradicional, é função do criador garantir a consistência e coesão da sua obra ou produção. Em Beowulf, isso envolve tanto o autor do poema em si, quanto toda a tradição oral em torno da estória. O resultado é uma unidade formal e temática para o universo ficcional, para cada um dos eventos e gestos realizados por seus personagens.

Em uma narrativa ou jogo digital, o designer também precisa garantir a coerência e a consistência da experiência. Porém, neste caso isso se dá em uma etapa diferente do processo criativo, no momento da autoria do sistema ou simulação.

Anteriormente citamos o jogo Dwarf Fortress (Bay 12 Games, 2006-), ilustrando o emprego de técnicas de geração procedural de conteúdo. Este jogo, além de apresentar muitas similaridades temáticas com o universo ficcional de Beowulf (se passa em um mundo de fantasia com castelos, calabouços, dragões e guerreiros), ele se baseia em uma simulação que também se assemelha ao caráter sistêmico do poema. Por exemplo, assim como há diversas lendas na estória de Beowulf, habitantes do mundo de Dwarf Fortress possuem a sua própria história, e podem ser admirados ou temidos por outros personagens na simulação.

A pesquisadora Sherry Turkle faz um paralelo similar entre os sistemas dos mundos nos jogos digitais e as regras de RPGs ("role-playing game", ou "jogo de representação de papéis") de tabuleiro como Dungeons & Dragons (Gary Gygax e Dave Arneson, 1974) (TURKLE, 2003).

Mais adiante neste capítulo, investigaremos como esse tipo de sistema pode ser implementado em uma narrativa ou jogo digital. Examinaremos alguns exemplos de jogos criados no MissionMaker que exploram este tipo de abordagem (embora, como veremos, de maneira relativamente limitada), e analisaremos os recursos e funcionalidades oferecidos por esta ferramenta.

3.1.2 MissionMaker

MissionMaker é um aplicativo voltado para a criação de jogos 3D em primeira pessoa, sendo desenvolvido dentro do projeto Playing Beowulf. O editor permite a criação de jogos compostos de diferentes ambientes, objetos e personagens, bem como a definição de regras, comportamentos e propriedades para estes elementos. Além do editor em si, o projeto também inclui um engine proprietário (IMMERSIVE EDUCATION, 2010).

Este aplicativo se distingue da maioria das ferramentas similares por conta do seu propósito didático - a intenção do projeto, como vimos, é o ensino da programação. O MissionMaker permite que mesmo usuários novatos criem jogos completos e totalmente funcionais quase que imediatamente. Daí em diante, é possível customizar e adicionar novos elementos gradualmente.

Por conta deste propósito didático, o desenvolvimento do MissionMaker se norteia pela busca de um equilíbrio entre a simplicidade e a função. De um lado, o objetivo é criar uma ferramenta acessível, o que inevitavelmente significa a imposição de certas limitações e simplificações. Do outro lado, busca-se preservar as funcionalidades e características fundamentais que se espera de um sistema de autoria procedural.

Convém notar que este é um projeto em desenvolvimento, e que este equilíbrio ainda não foi alcançado em determinados aspectos e elementos do aplicativo. Além disso, a ferramenta ainda apresenta alguns problemas e falhas pontuais. Como veremos ao longo dos estudos práticos, embora o MissionMaker ofereça a maior parte das funcionalidades pretendidas, e seja relativamente estável, alguns dos usuários tiveram dificuldades em lidar com determinados aspectos da interface, e para a realização de certas tarefas.

De forma a contextualizar os estudos práticos apresentados neste capítulo, realizaremos a seguir uma breve introdução ao fluxo de trabalho do MissionMaker, bem como os principais elementos da interface do aplicativo.[96]

[96] A versão do MissionMaker utilizada na oficina, bem como para os estudos práticos de uma maneira geral, se encontrava licenciada para fins educacionais e de pesquisa.

Editor: Interface e Fluxo de Trabalho

A imagem abaixo mostra os principais elementos da interface do MissionMaker (fig. 15). Neste aplicativo, geralmente a criação de um novo projeto começa com a adição de **localidades** no mapa, de forma a compor a topologia do espaço. Isso é feito arrastando ícones representando as áreas desejadas, do menu visual no topo, até a perspectiva de mapa.[97]

Figura 15 – MissionMaker: interface

Depois disso, é possível incluir **objetos** nos ambientes, como Props (ou "objeto de decoração"), Active Props (objetos ativos, que permitem algum tipo de interação ou comportamento) e Characters (os personagens). As propriedades e ações individuais associadas a cada objeto podem ser acessadas nos painéis que surgem na metade inferior da tela, à direita.

Cada tipo de objeto tem diferentes propriedades disponíveis - por exemplo, personagens tem propriedades relativas a aparência, força e estilo de caminhada, e itens de energia tem o nível de saúde (Health) que será obtido com seu uso. Além disso, cada tipo de objeto permite diferentes modos de edição para o designer, afetando a liberdade de deslocamento, redimensionamento e rotação, por exemplo.

[97] O usuário também pode iniciar a criação a partir de outros aspectos ou elementos do jogo, como customizar as propriedades globais do projeto, do jogador, e assim por diante. Entretanto, geralmente isso é feito depois que algumas localidades e objetos já tenham sido criados.

Os tipos de objeto também podem permitir diferentes usos de acordo com o contexto. Por exemplo, Props geralmente são objetos meramente decorativos, entretanto, como veremos mais adiante na tese, também podem influenciar (ou ser influenciados) pelo sistema do jogo, de acordo com a maneira como a sua função for programada.

Também é possível criar objetos de **mídia** (Media), compostos de áudio, vídeo, voz sintetizada ou "pop-ups" - telas, ou mensagens que podem conter texto, imagem ou ambos. Estes objetos podem ser empregados para transmitir informações ao jogador, fornecer algum tipo de feedback, ou para produzir algum efeito expressivo.

Por exemplo, um pop-up pode ser programado de forma a mostrar uma mensagem na introdução ou no final do jogo, ou um arquivo de áudio pode ser reproduzido se um personagem for ferido. Para implementar este tipo de lógica no MissionMaker é necessário utilizar as regras, tratadas a seguir.

Até agora falamos principalmente da criação e edição de elementos **estáticos**. De forma a definir regras ou comportamentos é necessário criar elementos **procedurais**. Para criar (ou definir) uma regra (Rule), primeiro deve-se escolher qual é a ação (Action) resultante que se deseja, e em seguida a condição que irá ativar ela.

Por exemplo, é possível criar uma regra que faz com que um certo personagem diga alguma coisa (a ação) quando o jogador entrar em um ambiente (o ativador, ou Trigger). Ou então, uma porta pode ser programada para abrir (a ação) assim que um determinado período de tempo tiver passado (o ativador).[98]

Estas são as principais funcionalidades e operações presentes no editor do MissionMaker, a partir das quais cada usuário pode definir o seu próprio processo criativo ou fluxo de trabalho para desenvolver o seu jogo.

[98] Para alguns usuários do aplicativo este processo para criação de regras segue uma lógica "invertida", como veremos mais adiante (p. 143). De fato, embora exista um item para a criação de regras no menu NEW (no topo da tela à direita), ele não tem função prática. O usuário precisa sempre selecionar primeiro a ação que deve acontecer.

Jogo: Interface e Controles

Durante a execução dos jogos criados no MissionMaker, além da perspectiva em primeira pessoa do ambiente virtual, há uma interface pela qual o jogador pode visualizar algumas informações (esta interface também é conhecida como "HUD" - sigla em Inglês para "heads-up display"). Esta interface pode ser vista nas imagens dos jogos, mostradas mais adiante neste capítulo (por exemplo, na fig. 20; p. 107).

A interface consiste em um inventário (mostrando os itens de posse do jogador, bem como algumas ações que podem ser realizadas com eles), mostradores numéricos para a pontuação e um marcador de tempo. Também existem mostradores visuais (as barras verticais) representando três economias - isto é, qualidades quantificáveis -, simbolizadas com os ícones de um coração, um braço (ou bíceps) e um sanduíche. A função específica de cada um destes valores e economias no jogo pode variar de acordo com a sua implementação.[99]

Para deslocar-se, o jogador deve utilizar as teclas direcionais. O mouse (ou "trackpad") serve para direcionar o olhar ao redor. Também é possível utilizar determinadas teclas para realizar ações, como interagir com algum objeto, pular, agachar e assim por diante.

[99] O HUD pode ser configurado, porém entre os jogos analisados neste capítulo, nenhum dos criadores modificou a interface padrão do aplicativo.

3.1.3 Oficina Piloto

A oficina piloto Playing Beowulf foi realizada como uma maneira de explorar algumas das ideias sendo discutidas para o projeto naquele momento, incluindo as possibilidades didáticas da leitura ativa de um texto (o poema Beowulf) através da realização de atividades criativas e expressivas, entre as quais o exercício de criação de jogos.

O público da oficina era composto predominantemente de "professores aprendizes" - estudantes do programa de formação de professores em Inglês e Drama da UCL. Os participantes foram estimulados a explorar o poema épico Beowulf, analisando e discutindo o texto, sua linguagem, estrutura narrativa e temas, bem como a maneira como os personagens, ambientes e situações são representadas.[100]

Também foram discutidas as múltiplas adaptações do poema em diferentes meios e linguagens. No Cinema, por exemplo, uma das adaptações foi o filme animado em computação gráfica, dirigido por Robert Zemeckis (2007; fig. 16, abaixo). Na Arte Sequencial (histórias em quadrinhos, ou HQs), uma das principais adaptações do poema foi na "graphic novel" (ou "novela gráfica") de Gareth Hinds (2008; fig. 17). Também foram apresentados exemplos de adaptações na Escultura, na Música e nas Artes Visuais.[101]

Figura 16 – Beowulf (Robert Zemeckis, 2007)

Figura 17 – (Gareth Hinds, 2008)

[100] O termo empregado para descrever o aluno de um curso de formação de professores no Reino Unido é "student teacher" (em tradução literal, "professor aprendiz"). A coordenação da oficina envolveu diversos membros da equipe do projeto Playing Beowulf, incluindo Andrew Burn, Jane Coles, Theo Bryer (responsável pelos exercícios de drama e improviso), Simon Thomson (pesquisador em literatura inglesa e contador de estórias) e Abel Drew (programador do projeto), além de mim. A oficina piloto aconteceu no prédio central do IOE UCL, em Outubro de 2014, e teve a duração de dois dias.

[101] Fonte das imagens: divulgação.

Posteriormente, esta variedade de linguagens e meios se refletiu nas atividades criativas propostas para os participantes da oficina. Foram realizados exercícios criativos em teatro de improviso (fig. 18, abaixo), redação criativa, desenho e filme, entre outros.

Figura 18 – Oficina: improviso em dramaturgia

O segundo dia da oficina foi dedicado exclusivamente para a atividade de criação de jogos digitais (ou videogames), usando o MissionMaker para criar aventuras baseadas em certas partes do poema (fig. 19, abaixo).[102]

Figura 19 – Oficina: criação de jogos com MissionMaker

[102]Após esta primeira oficina houveram outros eventos similares, e que também informaram os nossos estudos. Entretanto, nesta pesquisa nos concentramos na análise dos jogos produzidos na oficina piloto. Foi produzido um vídeo documentando a oficina: Beowulf Story: Embellishing Beowulf (youtube/Andrew Burn; Theo Bryer, 2015) - URL: `youtu.be/ARdgIOQZSKI`

3.2 Análise dos Jogos

Nesta seção serão apresentadas as análises de sete jogos produzidos utilizando o MissionMaker, no contexto da oficina piloto Playing Beowulf. Para estes estudos práticos, nos interessa principalmente a forma como cada designer escolheu representar as situações e temas do poema original, com ênfase no emprego dos recursos e estratégias procedurais de criação oferecidas pelo aplicativo.

Antes de apresentar as análises em si, descreveremos a seguir de maneira sucinta o processo adotado, incluindo os critérios utilizados para a seleção e investigação dos projetos, bem como o guia empregado para as entrevistas com os criadores, e a estruturação deste registro.

No final desta seção, após as análises dos jogos, também apresentaremos alguns dos temas e tópicos recorrentes identificados ao longo das entrevistas (p. 203).

Processo das Análises

Inicialmente, todos os jogos produzidos na oficina foram analisados seguindo dois critérios principais. Primeiro, o uso que cada criador fez das ferramentas e funcionalidades oferecidas pelo aplicativo, em termos quantitativos (a variedade e quantidade de funcionalidades utilizadas) mas principalmente qualitativos (a maneira como as ferramentas foram utilizadas, e o domínio que o usuário tinha delas).

O segundo critério foi o grau de finalização e acabamento do projeto. Isto é, até que ponto o jogo oferece, além dos elementos fundamentais para uma experiência completa (em termos de estrutura e funcionamento), algum tipo de acabamento ou personalização. Isso inclui o refinamento das regras e comportamentos dos personagens e elementos, bem como a inclusão de efeitos visuais, sons e música, por exemplo.

Na oficina foram criados cerca de vinte jogos no total, alguns deles em dupla. Conversei individualmente com os criadores de sete dos jogos, durante cerca de uma hora cada.[103]

[103] As entrevistas foram conduzidas entre os dias 21 e 28 de Novembro de 2014, no prédio central do campus do Institute of Education. Convém notar que a grande maioria dos participantes da oficina eram do sexo feminino - entre os jogos analisados, apenas o criador de um deles é do sexo masculino.

As entrevistas foram baseadas livremente no seguinte guia:

1. **Introduzimos** os objetivos do projeto Playing Beowulf, bem como desta pesquisa, incluindo uma descrição dos tópicos a serem discutidos. Além disso, como as entrevistas aconteceram algumas semanas depois da oficina, nos momentos iniciais dos encontros relembramos as atividades realizadas na ocasião, bem como as principais discussões que aconteceram.

2. **Jogamos e analisamos** o projeto lado a lado com o seu criador. Esta etapa podia acontecer apenas no início do encontro, ou ao longo da entrevista, durante a conversa. Em alguns casos, o arquivo de projeto do jogo também era explorado no editor.

3. Conversamos a respeito do **processo criativo** no design e implementação do jogo utilizando o MissionMaker. Também discutimos eventuais correções e ajustes que poderiam ser feitos nos jogos.

4. Discutimos sobre os aspectos mais **teóricos** relacionados à experiência na oficina, a atividade de criação de jogos e outros tópicos relevantes ao projeto. Também tratamos sobre eventuais **reflexões** que o entrevistado pudesse ter sobre os temas discutidos.

Depois de cada encontro, todo o material e informações obtidas eram compilados e organizados, incluindo a gravação em áudio das entrevistas e as anotações em papel. Imagens e vídeos da interação com os jogos também eram capturadas, como forma alternativa de registro.[104]

O primeiro estudo prático, apresentado a seguir, é mais detalhado do que os demais, uma vez que também serve como uma introdução ao tipo de análise que será feita sobre os jogos, bem como a muitos dos temas e elementos que serão tratados ao longo do capítulo. Por exemplo, como todos os entrevistados eram professores aprendiz, um tema recorrente em todas as entrevistas era a aplicação da criação de jogos como um recurso pedagógico em escolas. Outra discussão recorrente era a respeito dos paralelos entre os jogos digitais e outros meios e linguagens tradicionais. Lembrando que abordamos estes temas sempre a partir do nosso recorte de pesquisa, isto é, a investigação do processo criativo do designer que exerce autoria procedural com fins expressivos.

[104]Foram produzidos dois vídeos mostrando trechos dos jogos analisados: Andrew Smith (gameplay video; youtube/Andrew Burn, 2015) - URL: `youtu.be/D-TV1S8lV4M`; e Six Games (gameplay video compilation; youtube/Andrew Burn; Daniel Ferreira, 2015) - URL: `youtu.be/uYQEZi3WcIA`; Como vimos, este processo também resultou em dois artigos, descrevendo cada estudo prático, identificando e elaborando sobre as ideias e tópicos mais relevantes (FERREIRA, 2014a; 2014b).

Convém notar que, como uma parte significativa destes estudos práticos se baseia em entrevistas, a linguagem que empregamos neste capítulo é mais pessoal e informal do que no restante da tese. Além disso, como todos os jogos foram baseados no mesmo material de referência, muitos deles compartilham dos mesmos elementos, personagens e temas - praticamente todos os jogos incluem Beowulf e Grendel (ou sua mãe), a maioria termina com o jogador enfrentando o monstro em seu covil, alguns fazem menção aos restos mortais de Æschere, e assim por diante.

Dito isso, para as análises em si, demos ênfase nos elementos e aspectos originais encontrados em cada um dos projetos. Da mesma forma, nas entrevistas procuramos identificar e aprofundar nos temas e tópicos que refletissem a individualidade de cada participante. Também apontamos eventuais conexões e similaridades que notamos entre os projetos, bem como nas falas e reflexões realizadas pelos entrevistados.

Antes de cada uma das análises, apresentamos uma imagem representativa do conteúdo do jogo, bem como uma descrição sucinta dos seus elementos principais, incluindo os personagens, locais e mecânicas envolvidas.

3.2.1 O Servo Fiel (Andrew Smith)

Figura 20 – O Servo Fiel (Andrew Smith)

No jogo "O Servo Fiel" (ou "The Faithful Flunky"), criado por Andrew Smith, o jogador, no papel de Beowulf, recebe a missão do rei dos Danos (fig. 20) para matar a mãe de Grendel e encontrar a sua coroa. No covil do inimigo há um obstáculo adicional, que é o servo do monstro, dedicado a protegê-la de Beowulf a todo custo. O jogador vence se o monstro for morto.

A maioria dos jogos criados no contexto da oficina Playing Beowulf segue uma progressão relativamente linear, com uma série de tarefas e ações que precisam ser completados em uma determinada ordem, de forma a alcançar o final (e a vitória). Embora o jogo de Andrew siga um esquema similar, ele também incluiu diversos elementos que são opcionais à aventura principal, como é o caso dos dispositivos narrativos ambientais (ou "environmental narrative devices"). Estes são elementos de natureza geralmente estática e passiva, que quando observados ou manipulados podem contribuir para a construção do universo ficcional e da narrativa.

Por exemplo, existe um livro que pode ser lido pelo jogador, reforçando a narrativa do jogo. Também existem alguns ratos mortos que podem ser encontrados em um dos corredores do covil do monstro. Quando o jogador examina os ratos, surge uma mensagem na tela representando os pensamentos do protagonista, refletindo sobre o ambiente em que está, no estilo de escrita do texto do poema original.

Processo Criativo

A experiência de Andrew com jogos digitais se deu principalmente através de franquias de estratégia, como Command & Conquer (Electronic Arts, 1995-) e Civilization (MicroProse, 1991-), e mais recentemente jogos de ação, como Grand Theft Auto V (RockStar Games, 2013). Ele também teve uma breve experiência com criação de jogos quando era mais jovem, com a ferramenta Klik & Play (Clickteam, 1994).

Esta experiencia e familiaridade com jogos digitais, bem como a compreensão da sua lógica e funcionamento, ajudaram Andrew no processo de criar e planejar o seu próprio jogo na oficina. Por exemplo, o fluxo do seu jogo foi planejado de forma a garantir um efeito dramático e narrativo específico, através do uso de regras condicionais simples. A mãe de Grendel ataca o jogador apenas se a coroa for retirada do seu covil, o servo agride Beowulf apenas se ele ameaçar o monstro, e a topologia do mapa é alterada dinamicamente através do surgimento de passagens novas entre determinados ambientes, exigindo do jogador novas estratégias e ampliando a tensão da experiência.

Ao começar a planejar o seu jogo, Andrew tinha uma estrutura narrativa mais linear em mente. Entretanto, à medida em que experimentou as funcionalidades disponíveis no aplicativo, ele percebeu o potencial para explorar uma abordagem mais livre e dinâmica. A versão final do seu jogo ainda revela alguns aspectos desta abordagem inicial mais linear e estática. O diálogo com o rei no início, por exemplo, apresenta as tarefas específicas a serem completadas por Beowulf - assassinar a mãe de Grendel e encontrar a coroa. Entretanto, o jogo final contém uma série de outros caminhos e elementos para explorar.

Vimos que os dispositivos narrativos ambientais são bastante presentes neste jogo. Convém notar que elementos como este, embora tenham um grande potencial expressivo, são de natureza predominantemente estática, uma vez que não tem efeito funcional no sistema do jogo (isto é, não afetam o comportamento da simulação). Por exemplo, no caso dos ratos mortos mencionados anteriormente, eles podem ser examinados pelo jogador, mas isso não influencia nenhum dos outros elementos do jogo, tampouco o comportamento dos outros personagens.[105]

Alguns jogos baseiam sua mecânica principal neste tipo de abordagem. É o caso de Cozinheiro das Almas (Poéticas Digitais, 2006), no qual o jogador visita ambientes da trama de O Perfeito Cozinheiro das Almas deste Mundo, o diário da garçonnière mantida por Oswald de Andrade entre 1918 e 1919 (PRADO et al., 2006).

[105]Esta distinção, entre os elementos expressivos procedurais e os de natureza estática, é explorada em maior profundidade mais adiante na tese.

Esta abordagem tem sido cada vez mais presente nos últimos anos, em jogos como Gone Home (The Fullbright Company, 2013; fig. 21, abaixo), Dear Esther (The Chinese Room, 2012; fig. 22, abaixo) e Everybody's Gone to the Rapture (Idem, 2015). Neste tipo de jogo, descrito por alguns críticos como "walking simulators" (ou "simuladores de caminhada), o papel principal do jogador é de **explorar** o ambiente do mundo virtual, ao invés de **afetá-lo**.

Figura 21 – Gone Home (2013)

Figura 22 – Dear Esther (2012)

Note que a forma expressiva que caracteriza esta abordagem criativa no emprego das mídias digitais tem muito em comum com outros tipos de manifestação artística tradicionais, como a arte de Instalação ou mesmo com a Escultura. Isto é, a fruição nestes casos se dá principalmente a partir do deslocamento espacial do receptor em torno do objeto, ou dentro do espaço, explorando e observando as suas características e propriedades.

Para Andrew, as atividades criativas realizadas no primeiro dia da oficina ajudaram no processo de design e criação do seu jogo. Exercícios como os de escrita criativa e teatro de improviso exigiram dos participantes a capacidade de imaginar como os diferentes personagens da estória pensariam e agiriam em uma diversidade de situações. Isso permitiu um maior entendimento das cenas e dos personagens, incluindo as suas motivações, comportamento e o relacionamento entre eles.

Acabamento

Como nenhum dos participantes da oficina tinha experiência prévia com o MissionMaker, a maioria precisou de pelo menos algumas tentativas antes que os jogos funcionassem da maneira esperada. Durante os períodos de teste dos jogos, nos quais os designers experimentavam os jogos uns dos outros, era comum encontrar regras incompletas, objetos "esquecidos" e personagens com comportamentos estranhos ou problemáticos.

Como veremos, outro desafio era o fato de que o aplicativo em si, por conta de tratar-se de um projeto em desenvolvimento, apresentava alguns problemas funcionais e de usabilidade, bem como alguns "bugs" (isto é, falhas técnicas).

Documentos do Processo

Além de entrevistar os participantes, também tivemos acesso aos arquivos dos seus projetos no MissionMaker. Isso nos permitiu um olhar único sobre a forma como cada indivíduo estruturou e organizou o seu processo criativo, bem como a maneira como os elementos e regras foram implementados no sistema, incluindo a ordem na qual isso foi feito. Trata-se de uma abordagem de análise similar à da crítica genética tal qual descrita por Cecilia Almeida Salles, na qual o objeto de análise não é a obra finalizada, mas sim o processo de sua criação, a partir do que a autora chama de "documentos de processo" (SALLES, 2009).

No caso do jogo de Andrew, por exemplo, o acesso a este histórico do processo criativo nos revelou que um dos últimos elementos adicionados no seu jogo foram alguns sons representando vocalizações do jogador, em reação a determinados eventos. Isso pode sugerir que, em um dado momento, Andrew já estava satisfeito com o seu jogo, e decidiu dedicar algum tempo para incrementar a experiência.

Esta possibilidade de acessar os arquivos de projeto também contribuiu na compreensão de determinados problemas e erros encontrados nos jogos. Por exemplo, identificamos uma regra no projeto de Andrew que, por conta de um erro na implementação, não estava funcionando na prática. Como o efeito pretendido para esta regra em particular era sutil, ele não havia percebido que ela não estava funcionando.

Consistência e Coerência

A seguir examinaremos alguns comportamentos e regras implementadas por Andrew em seu jogo a partir da sua consistência e coerência com o restante da experiência. Veremos que certas situações que podem acontecer, embora sejam relevantes e significativas no contexto do universo ficcional estabelecido, não afetam o sistema, e tampouco são reconhecidas por ele.[106]

Por exemplo, o jogador pode ferir ou mesmo assassinar o rei, e isso não tem consequência alguma no contexto do jogo (além da mudança de estado do personagem do rei). Outra exemplo se dá na dinâmica entre o monstro e o seu servo, que podem matar um ao outro acidentalmente (por exemplo, se um estiver na linha de fogo do outro) sem repercussões na lógica do sistema.

[106] O tópico da consistência na expressividade procedural é retomado em alguns momentos da tese, e aprofundado mais adiante (p. 241).

Note que o problema aqui não se trata da possibilidade da ocorrência destes eventos em si, mas sim da sua coerência no contexto do jogo.

Por exemplo, se o servo do monstro atinge ela acidentalmente, não existe nenhuma reação ou indicação de que o sistema reconhece este evento de maneira significativa (isto é, tendo um significado no contexto do jogo). Seria natural que o servo demonstrasse arrependimento, ou que o monstro o repreendesse (mesmo que isso acontecesse através de uma simples vocalização ou mensagem de texto). Também seria esperado que os personagens ao menos tentassem evitar novos acidentes, talvez alterando a sua estratégia, mas isso não acontecia. Pelo contrário, nestes casos geralmente o personagem seria atingido diversas vezes, e eventualmente morreria.

Note que esta questão é menos problemática no caso da situação inversa, isto é, da mãe de Grendel atingindo seu servo por acidente. Neste caso, a ausência de consequências para este ato poderia ser justificada no contexto do jogo a partir da relação de poder entre os personagens. Isto é, seria coerente justificar esta situação a partir do argumento de que o monstro não se importa com o servo.

Outro elemento incoerente relacionado a esta regra é o fato de que, se o monstro morre em decorrência destes ataques acidentais do seu servo, o jogador é congratulado como se fosse ele o responsável pela morte.

Há outra situação que produz uma incoerência no jogo de Andrew, relacionada à coroa, localizada no covil do monstro. Este é um objeto significativo no contexto da narrativa, uma vez que representa uma conquista da mãe de Grendel (ela roubou a coroa do rei). Entretanto, da mesma forma que os inimigos de Beowulf podem facilmente se agredir acidentalmente, a coroa também pode ser destruída por engano sem repercussões no sistema. Esta situação contraria a premissa estabelecida para o jogo, uma vez que não é do interesse de nenhum dos personagens que a coroa seja destruída (exceto talvez em uma situação estratégica muito particular - entretanto, este não é o caso neste jogo).

Andrew explicou que gostaria de ter evitado algumas destas inconsistências. Por outro lado, em muitos dos casos elas seriam inevitáveis, pois ele queria garantir o acontecimento de uma narrativa relativamente predeterminada. Assim, de qualquer forma jogador teria que concordar em contribuir para o desenrolar de uma determinada sequência de acontecimentos.[107]

O jogo de Andrew também apresenta algumas inconsistências menores e mais pontuais, cujo impacto na experiência é menos significativo ou perceptível. Por exemplo, não é claro para o jogador qual é

[107] No drama interativo Façade, examinado anteriormente, foi adotada uma postura similar. Apesar de tratar-se de um sistema relativamente aberto e flexível, existem determinadas linhas narrativas que o jogador deve atender de forma a realizar a experiência tal qual foi planejada pelos seus criadores.

o significado da contagem regressiva no mostrador de tempo (no HUD), e tampouco a razão pela qual o jogo é reiniciado se ela chega a zero. Também não é explicado o que as economias de Força e Nutrição representam, ou o relacionamento entre elas no jogo. Assim, por exemplo, no momento em que o jogador perde energia ao beber de um vinho envenenado, não é claro a que esta perda se refere exatamente no contexto da simulação.[108]

Mensagens e Feedback ao Jogador

Neste jogo também há algumas incoerências no uso de pop-ups - as mensagens que aparecem para o jogador -, tanto em relação à forma quanto ao conteúdo destes elementos. Isso pode prejudicar a plena compreensão do jogador a respeito dos estados do sistema.

Por um lado, estas mensagens tratam-se de elementos estáticos - elas são compostas de texto, e eventualmente imagens -, e portanto sem impacto no sistema da simulação, e sem um potencial expressivo procedural. Por outro lado, elas são importantes justamente como uma maneira de representar e informar ao jogador a respeito dos eventos do sistema, e do comportamento, estado e propriedades dos diferentes elementos que compõem a simulação.

Uma inconsistência no jogo de Andrew se refere à aparência e formatação destas mensagens. Todas as mensagens exibidas apresentam a mesma imagem de fundo, tipo e cor de fonte, apesar de muitas cumprirem funções narrativas diferentes. Por exemplo, uma mensagem pode se referir a um diálogo, enquanto que outra pode representar os pensamentos do próprio protagonista, e em algumas ocasiões a fala do narrador onisciente. Mesmo nos casos em que a natureza da mensagem pode ser deduzida a partir do seu conteúdo ou contexto, há inconsistências, ora internas (por exemplo, diferenças no tom ou estilo entre mensagens diferentes representando uma mesma voz) ora em relação às demais vozes, ou ao universo ficcional de uma maneira mais ampla.

Em um dado momento, o jogador se depara com um kit de primeiros socorros. A reação natural é de tentar pegar este item, como geralmente se faz em jogos de primeira pessoa envolvendo combate. Se isso acontece, surge uma mensagem representando o narrador, declarando que aquele item não é necessário, pois Beowulf já é forte o suficiente. A intenção de Andrew com este elemento narrativo foi de reforçar a caracterização de Beowulf como um herói poderoso, entretanto na prática ele contradiz uma ação escolhida pelo jogador, gerando uma inconsistência.

[108] Também existem certas ações disponíveis ao jogador, comuns a todos os jogos criados no MissionMaker, e que não são reconhecidas ou relevantes no contexto da maioria das produções da oficina. Por exemplo, o jogador pode arremessar objetos, porém em nenhum momento este gesto é necessário, ou leva a algum resultado significativo (o equivalente a isso na Literatura, por exemplo, seria um livro contendo a descrição de uma determinada característica para um personagem que é irrelevante para o restante da narrativa).

A ausência de *feedback* para o jogador também pode ser problemática quando o comportamento do sistema por si só não é capaz de se fazer ser percebido. É o caso da situação do vinho envenenado, vista anteriormente, que tira energia do avatar. Neste caso, Andrew pretendia enfatizar para o jogador a necessidade de se concentrar na missão, ao invés de se distrair com tentações. Porém, o efeito da regra é muito sutil, quase imperceptível. Se fosse exibida uma mensagem neste momento, seria possível chamar atenção para o evento, e desta forma valorizar e potencializar o significado pretendido com a implementação da regra.

Mensagens também podem ajudar a evitar a ambiguidade em certas situações. Por exemplo, no projeto de Andrew, assim que o jogador é vitorioso, o jogo automaticamente se reinicia. Houve confusão quanto à razão desta escolha. Alguns dos participantes da oficina que testaram o jogo pensaram ser um erro na programação, outros que era apenas uma escolha prática (sem muito significado) e houve aqueles que interpretaram este reinício como um elemento significativo ou expressivo (talvez sugerindo que Beowulf, o protagonista, estava sonhando, ou repercussões filosóficas mais amplas para os eventos do jogo). A inclusão de uma mensagem resolveria esta ambiguidade (neste caso, Andrew havia feito esta escolha por motivos práticos, assim bastaria uma mensagem congratulando o jogador pela sua vitória, por exemplo).

Da Escrita Criativa à Autoria Procedural

Com base na sua experiência na oficina, Andrew considera que a criação de jogos pode ser um recurso educacional valioso. Discutimos como ele usaria o MissionMaker com os seus próprios alunos, para o ensino da escrita criativa (tanto de maneira direta, quanto indireta, informando o processo de aprendizado).

Ele notou diversas similaridades entre o processo de design e implementação de um jogo, e a escrita criativa. Ambos envolvem a construção de um universo de elementos, que podem representar objetos físicos, comportamentos, relacionamentos, propriedades e assim por diante.

Além disso, ele nota que em ambos os casos o processo criativo é de natureza dinâmica e iterativa, envolvendo um forte elemento de experimentação. Por exemplo, o designer de uma narrativa digital pode criar um personagem e testar diferentes parâmetros para a sua personalidade. Ou então, pode experimentar diferentes relacionamentos e interações, não apenas entre personagens, mas também entre outros elementos do sistema, permitindo identificar possíveis problemas ou inconsistências. No caso do escritor, a diferença é que estes experimentos acontecem apenas na sua mente (e talvez na forma de anotações e desenhos) - no fim do processo, o resultado é um texto fixo, ou estático.

Também discutimos a respeito da questão da interatividade nas narrativas e jogos digitais, que Andrew vê como uma forma de expressar e comunicar ideias de uma maneira mais clara ou completa, em comparação com outros meios e linguagens tradicionais.

Ele explica que um filme, por exemplo, é sempre o mesmo, não importa quantas vezes ele é assistido (fazendo referência à natureza estática desta linguagem). Mesmo que as interpretações do receptor possam mudar a cada nova exibição, a materialidade do filme é sempre a mesma (os mesmos quadros, as mesmas falas, trilha sonora e assim por diante). Já nos jogos digitais, o receptor pode escolher diferentes caminhos e ações a cada nova interação, mudando não apenas a sua perspectiva sobre o sistema, mas o próprio comportamento da simulação, potencialmente revelando novos elementos ou aspectos da experiência.[109]

[109] Trata-se de uma ideia interessante, entretanto convém notar que um argumento similar também pode ser feito em relação à expressividade procedural de um maneira mais ampla, independentemente da presença da interatividade. Retomamos esta ideia mais adiante na tese (p. 161).

3.2.2 A Caveira de Æschere (Alice Hawkins e Soraya Shabaan)

Figura 23 – A Caveira de Æschere (Alice Hawkins e Soraya Shabaan)

O jogo "A Caveira de Æschere" ("Æschere's Skull"), criado pela dupla Alice Hawkins e Soraya Shabaan, representa a estória de Beowulf depois que Grendel já foi morto. O jogador controla o herói, que recebe do rei (fig. 23) a missão para matar a mãe de Grendel. É necessário encontrar a caveira de Æschere, de forma a chegar até o covil do monstro. Se o jogador mata ela, surge um baú repleto de ouro. Abrindo o baú, o jogador vence o jogo.

Alice e Soraya não tiveram muitas dificuldades em compreender a estrutura e o fluxo de trabalho do MissionMaker. Elas também se familiarizaram rapidamente com os conceitos envolvidos, como as variáveis, procedimentos, comportamentos e ativadores (triggers). Por conta disso, a dupla foi capaz de dedicar algum tempo no final da oficina para trabalhar no acabamento do jogo, incluindo detalhes sutis, como efeitos sonoros para ações secundárias.

Entre as dificuldades encontradas por elas, a maioria era compartilhada com os demais participantes da oficina. Por exemplo, um dos problemas mais comuns era lidar com a ausência de modelos 3D adequados para representar os personagens do poema original (principalmente o rei).

Uma dificuldade específica que Alice e Soraya tiveram foi sobre a programação do comportamento de um dos personagens do jogo, a mãe de Grendel. Elas queriam que o monstro permanecesse em seu covil até a chegada do protagonista, porém isso não estava acontecendo. O motivo era que

todos os personagens criados no MissionMaker tem um comportamento padrão (ou default) que os faz caminharem livremente pelo ambiente do jogo. Eventualmente elas descobriram que isso poderia ser evitado incluindo-se uma regra logo no início de cada partida, anulando este comportamento.

Processo Criativo

Segundo Alice, sua experiência anterior jogando videogames foi um dos principais motivos que facilitaram o seu aprendizado do MissionMaker. Ela citou especificamente a sua familiaridade em lidar com a navegação no espaço 3D, bem como manipular e posicionar objetos em relação a uma determinada topologia. Entre os títulos que ela costumava jogar, ela citou o jogo de corrida POD: Planet of Death (Ubisoft, 1997) e o popular "simulador de vida" The Sims (Maxis, 2000-), ambos baseados no paradigma de representação em 3D.

Ela também mencionou que teve experiência com programação quando criança, aprendendo a interpretar e escrever código básico em HTML (a linguagem na qual páginas na web são representadas). De fato, esta experiência pode ter contribuído em parte na sua compreensão de alguns conceitos envolvidos no uso do MissionMaker. Entretanto, convém notar que a intersecção não é tão significativa, uma vez que o HTML é principalmente uma linguagem de marcação (em Inglês, "markup language"), em oposição a uma linguagem de programação de fato, envolvendo procedimentos, funções e assim por diante.[110]

No seu processo criativo, a dupla explicou que elas decidiram evitar definir um enredo ou narrativa de antemão, pois não queriam que o jogo se limitasse a uma sequência linear de eventos e ações. Primeiro, elas escolheram os ambientes (ou locais), objetos e personagens que gostariam de incluir no jogo. Em seguida, definiram os relacionamentos e conexões entre eles. Por fim, um percurso lógico integrando todos estes elementos foi idealizado. O resultado foi uma narrativa mais flexível e dinâmica, e menos linear do que na maioria dos outros projetos.

Alice declarou ser uma grande apreciadora do poema Beowulf, e do estilo e linguagem empregados no texto. Por conta disso, inicialmente ela se sentiu compelida a incluir no jogo um volume grande de texto, na forma de descrições, diálogos e assim por diante - tanto na forma de citações do poema original, quanto de material original escrito por elas mesmas, no estilo do texto.

Entretanto, elas logo concluíram que esta abordagem estaria indo contra o propósito da atividade, que era justamente de adaptar o poema para outras linguagens - neste caso, para o meio digital, na

[110]Documentos em HTML podem conter em si elementos de programação ou *scripts* de outras linguagens, tais como JavaScript, por exemplo. Entretanto, o HTML em si trata-se de uma linguagem essencialmente estática de formatação, unicamente descrevendo (através de *tags*) a maneira como uma determinada informação deve ser exibida pelo navegador (a nova versão da linguagem, a HTML5, se assemelha mais a uma linguagem de programação propriamente dita).

forma de um jogo. Assim, elas decidiram que, mesmo que fossem incluídos alguns trechos do poema original, estes deveriam cumprir uma função secundária na experiência, predominantemente servindo à jogabilidade.

Linguagem

Segundo Alice, um dos aspectos mais proveitosos da experiência para ela foi ter que abandonar a sua própria zona de conforto. Embora ela tivesse alguma experiência criativa em outros meios, nunca havia feito nada parecido com a criação de jogos. Ao lidar com esta linguagem desconhecida, e empregar ferramentas com as quais não tinha familiaridade, Alice foi capaz de aprender uma nova abordagem criativa e expressiva. Entretanto, ela mesma reconheceu que este foi apenas um primeiro passo na sua compreensão sobre o potencial expressivo do meio digital, e mesmo particularmente das narrativas e jogos digitais.

A interatividade foi o aspecto que Alice considerou mais significativo no processo de idealizar e criar o seu jogo. O motivo foi principalmente pelo fato de que o jogo se basearia em Beowulf, cujo enredo gira em torno da figura de um herói. Segundo ela, isso se traduziu bem para a linguagem de um jogo digital, uma experiência que tradicionalmente exige que o jogador cumpra um papel similar.[111]

Por fim, Alice declarou que considera que esta experiência permitiu a ela uma nova leitura do poema original, trazendo para a sua atenção elementos e aspectos que ela nunca havia percebido. Por exemplo, ao planejar os comportamentos para a mãe de Grendel no jogo, e definir as regras sobre como ela deveria reagir a Beowulf, Alice acabou descobrindo novas nuances sobre estes personagens no poema.

[111] Convém reiterar entretanto que, embora a interatividade seja uma das maneiras mais diretas e intuitivas para fruir (ou "ler") um sistema, do ponto de vista funcional ela é secundária. Como vimos, nesta pesquisa consideramos que no meio digital a primazia é do aspecto procedural (p. 26).

3.2.3 Voltando Atrás (Harriet Piercy)

Figura 24 – Voltando Atrás (Harriet Piercy)

Em "Voltando Atrás" ("Backtracking"), projeto criado por Harriet Piercy, o jogador, no papel de Beowulf, encontra um baú contendo uma espada (fig. 24). Ao pegar a espada, surge uma mensagem com a missão para matar a mãe de Grendel. Em um dado momento do jogo, a topologia do espaço foi planejada de forma que o jogador precisa voltar atrás, de forma a progredir. Chegado ao covil, o jogador é atacado impiedosamente pela mãe de Grendel. Se o herói consegue matar o monstro, o jogo termina em vitória.

O jogo de Harriet é relativamente linear, uma característica que, como vimos, era comum na maioria das produções realizadas na oficina. Esta tendência em parte se justifica pela falta de experiência dos participantes com o processo criativo envolvido no design e criação de um jogo. Entretanto, pudemos verificar que na maioria dos casos os motivos para isso resultaram de fatores externos, da própria maneira como a oficina foi estruturada.

Um destes motivos, de natureza prática, foi que os participantes tiveram pouco tempo para aprender e explorar as possibilidades da ferramenta de maneira plena. Outro motivo é o fato de que, como mencionamos ao longo da tese, nas mídias digitais ainda há uma tendência a perpetuar formas e estratégias criativas tradicionais. Assim, na oficina, embora a intenção era de incentivar o emprego do meio digital a partir dos seus próprios mecanismos, inevitavelmente a atividade acabou sendo influenciada significativamente pela forma da obra original no qual ela se baseou.

Por exemplo, os participantes foram instruídos a escolher uma "cena" do enredo de Beowulf como base para os jogos. O conceito da "cena" é tradicionalmente associado a linguagens lineares - isto é, a cena pode ser definida como uma determinada sequência de eventos (além disso, um enredo pode ser definido como uma sequência de cenas). Um termo mais adequado neste caso seria "situação", por exemplo, que pode ser associado a conceitos mais dinâmicos como os de estado, condição, propriedades e comportamento, e assim teria mais em comum com a lógica sistêmica de um jogo digital.[112]

De qualquer maneira, embora os jogos produzidos na oficina fossem predominantemente lineares, a maioria deles apresenta pelo menos alguns elementos dinâmicos, aproveitando a natureza procedural do meio empregado.

No jogo de Harriet, por exemplo, existe uma situação na qual uma passagem aparentemente livre se fecha assim que o jogador tenta passar por ela. Trata-se de uma situação interessante e cuidadosamente planejada, e um dos poucos casos entre os jogos produzidos na oficina em que um quebra-cabeça (ou desafio) foi criado com base na sincronização entre o movimento do jogador e um evento programado. As regras envolvidas são relativamente simples, mas a lógica empregada na implementação desta situação revelou que Harriet compreendeu os conceitos fundamentais envolvidos na autoria procedural.

MissionMaker

Conversamos com todos os entrevistados a respeito da sua experiência anterior com jogos digitais. De uma maneira geral, como esperado, aqueles participantes que já tinham alguma familiaridade com esta linguagem tiveram maior facilidade com a atividade de criação de jogos. Entretanto, em alguns casos esta correspondência não se deu de maneira tão direta e evidente.

A experiência de Harriet com jogos digitais se deu principalmente com títulos cuja jogabilidade envolvia (ou exigia) algum tipo de criatividade por parte do usuário. É o caso de Theme Park (Bullfrog, 1994), um jogo de estratégia no qual o jogador é o responsável pela construção e administração de um parque de diversões. Assim como em outros títulos similares, como a série Sim City (Will Wright, 1989-) e The Sims, a mecânica de jogo em Theme Park é essencialmente criativa - o jogador tem ferramentas com as quais pode construir os diferentes elementos do seu parque, modificando e consertando partes sempre que necessário, e assim por diante.

[112] É interessante contrapor isto com a maneira como os demais exercícios criativos da oficina foram realizados. Por exemplo, na atividade de desenho, os participantes estavam livres para se expressar unicamente a partir de elementos visuais. Na atividade de criação de vídeo, por sua vez, foi encorajada a realização de produções mais subjetivas (isto é, não foi imposta a necessidade de incluir determinados elementos específicos do poema, como certos personagens ou diálogos).

Segundo Harriet, esta experiência certamente ajudou em determinados aspectos do fluxo de trabalho do MissionMaker, principalmente no que se refere à criação e manipulação de elementos, e a sua disposição no ambiente de jogo, a partir da interface visual da ferramenta. Por outro lado, ela teve muita dificuldade com relação aos elementos procedurais, como a definição de regras e comportamentos, funcionalidades que geralmente não se encontram à disposição do jogador em títulos como os mencionados anteriormente.

De fato, ao analisar o arquivo de projeto do jogo de Harriet no editor, pudemos identificar algumas das dificuldades que ela teve, principalmente nos elementos procedurais. Como vimos, as regras do projeto são armazenadas em uma lista, organizadas cronologicamente de acordo com o momento em que foram criadas. No projeto de Harriet a lista tinha diversas instâncias de regras que não funcionavam, geralmente porque faltava algum elemento ou associação necessária (por exemplo, havia uma regra associada a uma porta, que deveria fazê-la abrir, mas nada acontecia porque não havia sido definida uma condição para isso acontecer). Na mesma lista, pudemos verificar que eventualmente Harriet conseguiu criar regras funcionais, uma vez que as instâncias mais recentes estavam implementadas de maneira correta.

Comparando Linguagens

Apesar da sua dificuldade em lidar com a autoria procedural no MissionMaker, Harriet considerou o processo interessante, principalmente pela sua natureza sistemática. Ela observou o fato de que o designer (ou programador) precisa ser capaz de descrever em detalhes todos os elementos envolvidos em um projeto, incluindo todas as suas propriedades, comportamentos e a maneira como eles se relacionam entre si.

Harriet sugeriu uma comparação, explicando que nos demais meios e linguagens tradicionais é possível obter resultados parciais ao longo do processo, enquanto que nos jogos digitais isso não ocorre. Por exemplo, um desenhista pode começar fazendo um esboço, depois criar os traços principais, os detalhes e finalmente a cor - mas em todos os estágios deste processo já existia uma versão do desenho. Enquanto isso, segundo ela, na criação de um jogo digital todos os elementos envolvidos precisam ser programados por completo - e sem erros - antes que qualquer resultado pudesse ser produzido.

Em parte, esta observação de Harriet é válida, entretanto ela também revela uma confusão na sua percepção sobre a lógica computacional. Na prática, o programador também produz o seu trabalho de maneira gradual - deixando elementos ou aspectos do projeto em aberto enquanto trabalha em outros. A maior parte das linguagens de programação se baseia nesta lógica modular.

Se um certo elemento não for programado de maneira completa (ou correta), de fato ele pode não funcionar (por exemplo, o caso das primeiras regras criadas por ela em seu jogo) - embora isso não vá necessariamente invalidar o sistema como um todo (embora isso seja uma possibilidade). Deste ponto de vista, no meio digital existe a chance de que um engano ou esquecimento, mesmo que pequeno por si só, tenha um impacto significativo, enquanto que em meios tradicionais isso seria menos comum. Isto é, geralmente se um desenhista esquece ou erra um traço, a princípio o impacto disso na ilustração como um todo não é tão grande quanto se um programador esquece de digitar um determinado caractere em seu código, por exemplo.

É difícil generalizar esta questão, uma vez que há uma infinidade de situações nas quais o oposto acontece. Em alguns casos pode ser possível apagar múltiplas linhas (ou mesmo funções completas) do código de um jogo sem afetá-lo de uma maneira significativa - por exemplo, se for a parte do código que descreve um efeito visual. Enquanto isso, há casos em que a mudança de uma única nota em uma música, ou uma letra deslocada em um texto, podem alterar completamente a natureza da obra como um todo.

De uma maneira geral, entretanto, é verdade que a linguagem algorítmica tende a ser mais frágil ou instável neste sentido, em comparação com outros meios tradicionais. O alto grau de abstração possibilitado pelo meio digital significa que o potencial de distanciamento entre o gesto de quem cria e o produto final é muito maior do que em outros meios e linguagens. Embora um programador a princípio esteja livre para criar e organizar o seu código da maneira que desejar, ele precisa obedecer minimamente à sintaxe e a outros requisitos da linguagem sendo empregada, de forma que possa ser interpretado e executado pelo computador.

Autoria Procedural

Ao refletir sobre a lógica metacriativa por trás da abordagem procedural, Harriet propôs uma comparação entre as linguagens do Cinema e do Videogame, mais especificamente sobre o discurso em torno de obras particulares em cada uma delas. Ela observou que, ao conversar sobre um filme, as pessoas geralmente fazem referência a determinadas cenas ou diálogos. Enquanto isso, o discurso em torno dos jogos digitais envolve conceitos como comportamentos, propriedades e sistemas - não apenas a instâncias específicas de interação.

Ela também observou as similaridades entre a atividade de criação de jogos e a prática do improviso. No improviso, o ator não tem um roteiro (ou "script") contendo as suas falas, ou uma descrição das ações que deve realizar na cena. Ao invés disso, recebe instruções mais genéricas sobre como deve

agir e reagir a determinados tipos de situação, ou uma descrição da personalidade do seu personagem. De certa forma, estas instruções podem ser vistas como "algoritmos", que são "executados" pelo improvisador (embora evidentemente sem o mesmo nível de objetividade e precisão apresentados por um computador).

Harriet acredita que um aplicativo como o MissionMaker pode ser uma ferramenta valiosa na sala de aula. Ela atentou para a importância da presença de um educador capaz de decidir quais elementos e funcionalidades, e em que nível de complexidade, devem ser apresentados à turma, de acordo com a faixa etária dos alunos, entre outros fatores. Para ela, idealmente este tipo de atividade deveria ser introduzida no currículo o mais cedo possível, juntamente com outros conteúdos elementares, como o desenho, a escrita e a matemática. Isto é, ela vê a lógica computacional como um tópico de aprendizado fundamental, e não como uma disciplina ou habilidade voltada apenas para determinados fins específicos.[113]

[113] Recentemente a programação de computadores se tornou parte do currículo nacional no Reino Unido, para crianças a partir dos cinco anos de idade (p. 204).

3.2.4 Vingando Æschere (Katie Smith)

Figura 25 – Vingando Æschere (Katie Smith)

"Vingando Æschere" ("Avenging Æschere"), de Katie Smith, começa exibindo um texto explicando ao jogador (Beowulf) que ele precisa vingar a morte de Æschere. Há uma trilha sonora orquestral dramática, e podem ser ouvidas vozes de origem indefinida, endereçadas a Beowulf. As vozes o encorajam e dão conselhos sobre os perigos que o herói irá encontrar. Após encontrar a caveira de Æschere e percorrer mais alguns ambientes e portões (fig. 25), Beowulf chega ao covil da mãe de Grendel. Para vencer o jogo, é necessário matar o monstro e coletar uma medalha que ela deixa para trás.

Este jogo é um pouco mais complexo do que a maioria das outras produções da oficina. Katie dedicou atenção especial em definir um sistema para sincronizar a reprodução da trilha musical e dos efeitos sonoros com as ações e movimentos do jogador. Por exemplo, em determinados locais ouve-se interjeições e suspiros representando as reações de Beowulf a certos eventos.

É interessante contrapor, de um lado, a maneira como Katie emprega estas interjeições para caracterizar Beowulf, e do outro, a intenção de Andrew em fazer o mesmo, com a situação do kit de primeiros socorros (p. 112).

Vimos que, no jogo de Andrew, há um conflito entre a intenção do jogador e a voz do narrador. Ao ver o kit, a maioria dos jogadores decide pegá-lo, uma vez que ele representa um benefício evidente, e o gesto não tem custo algum. O item está localizado de forma que o jogador passaria por ele de

qualquer maneira, e nenhuma ação adicional é necessária para pegá-lo. Entretanto, ao fazê-lo, o sistema do jogo desconsidera a ação (o kit de primeiro socorros não é coletado) e o narrador faz um comentário sugerindo que Beowulf é forte demais para precisar do item.

A intenção de Andrew ao implementar este comportamento foi de enriquecer a caracterização do personagem. A eficácia desta situação é ainda maior se o jogador é familiarizado com a simbologia do kit de primeiros socorros em jogos de ação - isto é, se Beowulf dispensa um item como este, de fato deve ser muito forte.

Entretanto, enquanto o jogo de Andrew **contradiz** de maneira ativa a ação escolhida pelo jogador, o de Katie apenas **acrescenta** elementos novos à experiência - por exemplo, uma interjeição de susto ao se deparar com um inimigo. Mesmo que o jogador discorde da reação do avatar a um dado evento, esta não contradiz de maneira ativa uma ação ou decisão tomada no contexto do jogo.

Playtesting

Para Katie, de acordo com sua experiência na oficina, uma das etapas mais importantes no processo de criar um jogo é o tese de jogabilidade (em Inglês, o "playtesting"). Segundo ela, é mais fácil para um designer encontrar problemas e erros em outros jogos, uma vez que nos seus próprios projetos os desafios, regras e elementos são todos conhecidos. O jogador que não conhece o sistema por trás de um jogo tende a experimentar mais, e portanto tem uma maior chance de encontrar situações problemáticas.

Um exemplo disso aconteceu no próprio projeto de Katie. Ela havia incluído um portão fechado logo no início do jogo, e programado ele para abrir após o jogador realizar uma determinada série de tarefas. Entretanto, na prática, por algum motivo o portão podia ser simplesmente atravessado pelo jogador, como se ele não existisse. A maioria das pessoas que testaram o jogo descobriam isso imediatamente por acidente, ao tentar abrir o portão simplesmente encostando-se nele. Katie não tinha percebido o problema antes justamente porque sabia o que precisava ser feito para abri-lo.[114]

[114] A origem do problema em si - do portão ser atravessável mesmo estando fechado - era um erro no próprio engine do MissionMaker, que por algum motivo não estava aplicando as propriedades físicas adequadas para o estado do objeto. Katie apenas deixou de notar que este erro estava acontecendo.

3.2.5 Através do Pântano (Laura Scott)

Figura 26 – Através do Pântano (Laura Scott)

Em "Através do Pântano" ("Through the Mere"), de Laura Scott, a aventura começa com a mãe de Grendel desafiando Beowulf a ir em seu encontro. O jogador chega até o pântano (fig. 26), uma das localizações icônicas do poema, e ao redor dele encontra alguns itens importantes. Entre eles, está a arma que precisará usar para matar o monstro. O covil pode ser alcançado "através do pântano", onde está a mãe de Grendel. O jogador é vitorioso se matar o monstro.

O design deste jogo é simples e direto, envolvendo uma sequência linear de ações, sem desvios ou elementos secundários a serem explorados. Como veremos, esta estrutura acabou contribuindo para o surgimento de um problema, no qual o jogador poderia se encontrar em uma situação sem saída.

O tema do jogo de Laura faz referência a uma passagem específica do poema, na qual o protagonista mergulha em um pântano, atravessando ele para chegar ao covil dos monstros. A representação desta situação no MissionMaker é problemática, uma vez que no engine do aplicativo não existe a opção para que os personagens possam entrar na água ou nadar. Na maioria dos projetos contendo o pântano, de forma a representar o ato de mergulhar na água, foi empregado um teleportador (Teleporter), um elemento que funciona como um portal entre dois pontos diferentes do espaço do jogo - o portal era posicionado na superfície da "água", de forma a transportar o jogador para outro local assim que entrasse em contato com ele.

No jogo de Laura, como vimos, há uma sequência específica que o jogador precisa realizar. Primeiro é preciso pegar a arma, para só então poder atravessar o pântano até o covil para enfrentar o monstro. Entretanto, em uma versão anterior do jogo, o jogador poderia "mergulhar" no pântano sem ter encontrado a arma, levando a uma situação sem saída - não era possível combater o monstro, e tampouco voltar atrás para recuperar a arma.

Este problema teve uma solução simples, bastando modificar as propriedades do portal para que o estado inicial dele fosse inativo, e de forma que ele só seria reativado após o jogador pegar a arma. Entretanto, outra maneira de lidar com o problema seria tornar a estrutura do jogo mais flexível, de forma a acomodar a emergência de situações inesperadas como esta (ao invés de coibi-las). Por exemplo, ao invés de exigir que o jogador pegue a arma antes de chegar ao covil, poderia ser implementado um novo portal pelo qual é possível retornar ao pântano se necessário.[115]

Colagem

Laura observou semelhanças entre o processo de criar um jogo usando o MissionMaker e a prática artística da colagem. Ela explicou que em ambos os casos o processo é fundamentalmente baseado na combinação de elementos ou módulos preexistentes, em oposição a criar tudo "do zero" (na expressão em Inglês, "from scratch"). No Missionmaker, estes módulos seriam os ambientes, objetos e personagens, bem como as regras, enquanto que na colagem seriam os diferentes recortes disponíveis.

Na colagem, o trabalho do artista não está na autoria dos recortes, mas sim na sua organização e composição na imagem final. Da mesma forma, a criação de jogos no MissionMaker consiste predominantemente da composição de elementos preexistentes.[116]

Entretanto, na colagem o artista não necessariamente compartilha autoria com os autores das imagens dos recortes (como os fotógrafos e ilustradores). Enquanto isso, na criação de jogos, é mais comum considerar o uso de funções ou elementos predefinidos como uma colaboração, ou ao menos evidenciar a autoria original destes elementos. Por exemplo, em jogos nos quais a simulação de física tem um papel fundamental (geralmente no gênero da ação, mas também em jogos de quebra-cabeça ou aventura), é comum dar crédito quando um engine externo é empregado.[117]

[115] Note que esta solução seria mais coerente no contexto do jogo. Faz mais sentido o jogador ser capaz de retornar por um caminho que já usou, do que estabelecer uma limitação arbitrária como a que foi implementada no jogo de Laura.

[116] O tema da composição será aprofundado mais adiante na tese (p. 185).

[117] Neste aspecto, os jogos digitais tem mais similaridades com formas como o Cinema ou a Música, nas quais as produções geralmente apresentam créditos com os nomes de todos os envolvidos.

Granularidade

Também conversamos a respeito das diferenças entre as linguagens da poesia e do videogame. Para Laura, o meio digital é definido pela abertura de escolhas, enquanto que o poema é uma obra fixa e linear. Ela notou, entretanto, que há diferentes níveis nos quais isso pode ser considerado.

Ela mencionou os livros da série Choose Your Own Adventure (Bantam Books, 1979-1998), que representam narrativas que se bifurcam de acordo com a escolha do leitor. Nestes livros, apesar desta abertura, as bifurcações acontecem sempre nos mesmos pontos, e a natureza do conteúdo ainda é essencialmente estática e linear. Enquanto isso, Laura explica que, no meio digital e nos videogames, a abertura é muito maior e mais granularizada, no nível das ações, gestos e eventos.[118]

Entretanto, Laura reconhece que o próprio jogo criado por ela na oficina é predominantemente linear, até mesmo mais do que os livros Choose Your Own Adventure. A razão disso é que, embora no jogo a princípio a granularidade nas escolhas seja muito superior, isso não implica necessariamente que elas sejam significativas.

O jogo de Laura representa de maneira eficaz certas partes do enredo de Beowulf, entretanto isto é feito de maneira predominantemente linear. Se o jogador decide experimentar algo diferente do que acontece no enredo original, estas variações não são previstas no sistema, e em alguns casos podem até mesmo "quebrar" o jogo (levar a situações sem saída, imprevistas ou que causam falhas no jogo).

Isto é, não basta oferecer ao receptor múltiplos caminhos ou ações, se as repercussões ou significados destas escolhas não forem previstos pelo autor, e também comunicados para o receptor, na forma de consequências e mudanças no estado dos elementos do sistema. Em outras palavras, como veremos mais adiante, um alto grau de intensidade procedural não necessariamente implica em um alto grau de expressividade procedural.

MissionMaker

Laura teve dificuldades em aprender a usar o MissionMaker, principalmente para programar as regras e comportamentos. Ela relata que levou mais tempo aprendendo a usar a ferramenta, testando e experimentando as diferentes funcionalidades e opções, do que criando o jogo propriamente dito. Muitos dos outros participantes relataram experiências semelhantes.

Em nossas análises, verificamos que um dos motivos para esta dificuldade, de caráter mais prático, se refere ao fato de que muitos dos usuários consideraram a interface da ferramenta e o seu fluxo

[118] A série da Bantham foi muito popular nos Estados Unidos nos anos 80, e ainda é relativamente bem conhecida, inclusive no Reino Unido. Note que este tipo de livro tem essencialmente a mesma estrutura bifurcada dos "filmes interativos", mencionados anteriormente.

de trabalho confusos. Por conta disso, embora a ferramenta em princípio permitisse uma série de operações avançadas, nem sempre era intuitivo para o usuário como fazê-lo.

Outra dificuldade que identificamos foi a compreensão de certos conceitos, como a noção de variáveis, funções, condicionais e operações booleanas, bem como aqueles mais específicos, como as propriedades e comportamentos dos personagens, a função do avatar, a noção de ambientes ("rooms") e objetos, e a maneira como estes diferentes elementos comunicam os seus estados entre si de forma a estabelecer a lógica do jogo.

Em alguns casos, mesmo quando um determinado conceito era claro para o usuário, a sua implementação particular na interface do aplicativo era confusa. Por exemplo, Laura compreendeu o conceito da regra (ou condicional), como ela pode ser utilizada para definir o comportamento de um personagem e mesmo os diferentes elementos que são necessários à sua implementação (o objeto da ação, a condição que ativa a regra e assim por diante); porém, ela teve dificuldades em se familiarizar com a maneira como este processo deveria ser realizado no editor.

Aplicações na Educação

Na época da entrevista, Laura estava estagiando em uma escola, dando aula para crianças e pré-adolescentes. Assim como a maioria dos outros entrevistados, ela acredita que o currículo escolar se beneficiaria da inclusão de disciplinas ou atividades relacionadas à criação de jogos, desde que feito de uma maneira bem planejada e em sintonia com as demais matérias. Segundo ela, uma maneira de garantir que este processo não prejudique ou interfira no equilíbrio do currículo preexistente seria adotar uma abordagem interdisciplinar, integrando a criação de jogos com as demais matérias sempre que possível.

Laura sugeriu dois exemplos interessantes de aplicações práticas deste princípio. O primeiro exemplo trata-se de um exercício no qual o ensino da escrita criativa e da criação de jogos acontece em paralelo, de forma que elementos e conceitos de uma atividade possam ser transferidos para a outra, incluindo as descobertas e “insights" dos alunos, em um processo de retroalimentação (ou "feedback").[119]

O outro exercício sugerido por Laura consiste no uso da criação de jogos para o ensino de um tópico mais específico na escrita criativa: os diferentes tipos de pessoa gramatical. Laura explica que, nas suas aulas, ela costuma usar videogames como exemplo de um narrador em segunda pessoa, uma vez que neste meio a ação e os eventos geralmente acontecem com o jogador. Segundo ela, embora nem todos os jogos adotem a linguagem verbal para comunicar ao jogador o estado do sistema, os demais

[119] Retomaremos este exemplo sugerido por Laura mais adiante na tese (p. 199).

aspectos da experiência sugerem uma narrativa em segunda pessoa (por exemplo, os gráficos, sons, comportamentos e a mecânica de interação). Nestes jogos, a fala do narrador é do tipo: "**Você** caminha pelo corredor", "**Você** ataca o inimigo", "**Você** abre a porta" e assim por diante.[120]

Pra Laura, a partir do momento em que os alunos criam jogos eles mesmos, programando a regras, comportamentos e propriedades, e lidando diretamente com o sistema e a lógica por trás desta experiência de "segunda pessoa", o conceito pode se tornar mais claro e intuitivo para eles. Assim, a atividade serve como um ponto de partida para discutir identidade, e o papel dos diferentes tipos de personagens e vozes narrativas em uma estória.

Por exemplo, neste exercício seria possível explorar como as demais vozes narrativas são empregadas nos jogos digitais. Vimos que o narrador em segunda pessoa faz mais sentido no caso de um jogo no qual o jogador controla um personagem. Por outro lado, nos jogos em que o narrador está em terceira pessoa, parece haver uma desconexão entre o que está sendo dito e a experiência do jogador. Isto acontece nos jogos The Stanley Parable (Davey Wreden, 2011) e Bastion (Supergiant Games, 2011), por exemplo - as falas do narrador neste caso são do tipo: "**Ele** alcança objetivo", "**Ele** é derrotado", "**Ele** está com medo" etc. Neste tipo de jogo, mesmo quando o narrador está se referindo ao avatar do jogador, a impressão é de que está falando sobre algum outro personagem.

O narrador em primeira pessoa também pode ser problemático, pois muitas vezes sugere uma desconexão entre o protagonista e o jogador. Este é o caso da maioria dos adventures, como o clássico The Secret of Monkey Island (LucasFilm Games, 1990), por exemplo - neles, a voz do protagonista se dá através de falas como: "**Eu** estou com a chave", "**Eu** não acho que isso vai funcionar", "**Eu** estou com fome" etc. Neste caso, o narrador está fazendo referência às intenções e pensamentos do personagem (isto é, de si mesmo), e não do jogador. É como se o personagem tivesse uma personalidade própria, ao invés de servir como avatar do jogador.[121]

A decisão por usar um narrador em primeira ou terceira pessoa pode ser expressiva, como no caso de Stanley Parable, em que a intenção é justamente de sugerir um distanciamento, de forma a causar estranhamento no jogador. De uma maneira geral, por conta da natureza única deste meio, que permite ao receptor controlar uma entidade que ao mesmo tempo se encontra submetida a um sistema ou simulação, a possibilidade de haver inconsistências é grande, como veremos mais adiante na tese.[122]

[120] De maneira simplificada, a voz em primeira pessoa é aquela de quem fala (Eu, Nós e assim por diante), segunda pessoa é aquela com quem se fala (Tu, Vós etc.) e terceira aquela de quem se fala (Ele, Eles etc.).

[121] O distanciamento entre o jogador e o seu personagem se torna ainda mais evidente quando este, utilizando voz em primeira pessoa, faz referência direta ao jogador - por exemplo "Você realmente quer que eu aperte a campainha?". Neste tipo de situação, relativamente comum em certos gêneros de jogos, principalmente naqueles com elementos de comédia, fica estabelecido claramente a distinção entre o jogador e o seu personagem.

[122] Também vimos um exemplo disso anteriormente, no jogo de Andrew Smith (p. 107).

3.2.6 A Recompensa (Niamh Hickey)

Figura 27 – A Recompensa (Niamh Hickey)

No jogo "A Recompensa" ("The Reward"), criado por Niamh Hickey, Hrothgar, o rei dos Danos, exige vingança pela morte de Æschere. O jogador (Beowulf) parte em sua aventura, encontra uma espada, a caveira de Æschere e uma arma, e chega até o covil da mãe de Grendel (fig. 27). Após matar o monstro, o jogador encontra restos mortais de suas vítimas, e então se vê de volta na presença do rei. Hrothgar congratula Beowulf pelo sucesso da missão e oferece a ele um baú cheio de ouro.

MissionMaker e Scratch

Em comparação com os demais participantes da oficina, a experiência anterior de Niamh com jogos digitais é bastante rica e variada. Além de The Sims, ela costumava jogar títulos do PlayStation, como o jogo de plataforma em 3D Crash Bandicoot (Naughty Dog, 1996), bem como jogos de computador mais antigos, baseados em texto ou com gráficos mais simples.

Niamh também teve experiência programando em HTML quando era criança (assim como Alice). Soraya, que também participou de parte desta conversa (ela era dupla com Alice), comentou que viu similaridades entre o MissionMaker e a linguagem de programação Scratch (Massachusetts Institute of Technology - MIT), com a qual ela já havia trabalhado antes. Entretanto, ela observa que Scratch tem uma lógica mais próxima do que seria, nas suas palavras, a "programação propriamente dita", isto é, lidando diretamente com o código.

Scratch é um ambiente de programação visual baseado em blocos de lógica, que correspondem às diferentes funções e comandos. Assim, ao invés de digitar o código diretamente, o designer pode compor o seu programa apenas combinando os blocos de lógica. No MissionMaker, por outro lado, a interface representa abstrações em um nível mais alto - isto é, mais distante das funções mais fundamentais encontradas em linguagens de programação de nível mais baixo.[123]

Por conta desta simplicidade, Soraya considerou o MissionMaker um sistema mais intuitivo para a criação de jogos, em comparação com Scratch. Por outro lado, isso também significa que, de uma maneira geral, o MissionMaker é mais limitado ou restrito, em termos da variabilidade nos projetos que podem ser criados. De fato, jogos criados no MissionMaker geralmente são relativamente parecidos entre si, enquanto que no Scratch existe uma abertura maior para customizar, incluir e modificar elementos, inclusive nos níveis mais fundamentais do código.

Comparando Linguagens (II)

Ao comparar o processo de criar jogos no MissionMaker com a prática da escrita criativa, Niamh considera que no meio digital os resultados são muito mais imediatos. Pra ela, a escrita criativa sempre começa "do zero", não apenas porque o processo começa literalmente com uma folha em branco (ou um arquivo vazio), mas também porque o escritor precisa, antes mesmo de criar os personagens e o enredo (quando este existe), formular todos os elementos fundamentais envolvidos - desde a lógica do universo ficcional, até a estrutura e estilo da narrativa. No MissionMaker, ela considera que a maior parte destes elementos mais fundamentais, nas suas palavras, "vem de graça", isto é, já existem no próprio engine.

Ela se refere ao fato de que o MissionMaker é baseado em um determinado engine, que por sua vez é estruturada em torno de um certo paradigma de representação do mundo, que neste caso é em três dimensões (3D), com um modelo específico de simulação de física e de renderização (a materialização da cena e dos objetos na forma de pixels na tela) aos quais todos os objetos, personagens e ambientes serão submetidos.

Assim, por exemplo, ao criar o seu jogo no MissionMaker, quando ela decidiu incluir o prêmio que o jogador recebe no final, bastou consultar a lista de objetos disponíveis na biblioteca até encontrar algo adequado (o baú com ouro), arrastar este elemento até o ambiente do jogo e adicionar uma regra (de forma que o baú apareça no momento correto). Isto é, ela não precisou modelar o objeto

[123] Retomaremos o tópico dos níveis de abstração mais adiante na tese (p. 156).

do baú em 3D e criar a sua textura de madeira, e tampouco precisou programar um engine que seria capaz de exibir este objeto na tela a partir de uma determinada perspectiva, e assim por diante.[124]

É interessante contrapor esta perspectiva de Niamh com a de Harriet (p. 121), uma vez que elas parecem opostas de início, mas na prática são similares. Harriet vê o processo criativo do designer de jogos como algo muito mais trabalhoso do que o do escritor, justamente por ser necessário programar - isto é, descrever e definir - todos os elementos e aspectos envolvidos. A rigor, entretanto, as duas estão descrevendo a mesma situação, apenas sob perspectivas diferentes, e considerando etapas distintas do processo criativo.

Se considerarmos, por exemplo, um designer que programa o seu jogo diretamente em uma linguagem de baixo nível, neste caso ele precisaria definir todas as funções e elementos fundamentais que no MissionMaker "vem de graça". Além do mais, é importante considerar que a princípio qualquer elemento de lógica representa uma oportunidade criativa ou expressiva. Assim, embora no MissionMaker o processo criativo seja facilitado pela presença de regras predefinidas, isso também representa uma perda de potencial expressivo para o designer, que não terá acesso aos seus mecanismos internos.

Atividades Criativas

Niamh também refletiu a respeito das atividades criativas no primeiro dia da oficina, e no seu papel fundamental como embasamento e referência para o segundo dia, na atividade de criação de jogos.

Na abertura da oficina, parte do poema original foi apresentado através de uma interpretação dramática. Depois, como vimos, foram apresentados filmes, animações, ilustrações e estórias em quadrinhos baseadas em Beowulf. Também foram promovidas discussões com base neste material, explorando as diferentes leituras e interpretações possíveis.

As atividades criativas permitiram uma aproximação ativa ao texto original, através de linguagens e meios tradicionais como teatro de improviso, desenho e filme. Niamh lembra que em um dado momento surgiu a discussão a respeito da aparência física da mãe de Grendel. No poema original não há uma descrição objetiva ou clara deste personagem. Por conta disso, cada participante teve que construir a sua própria interpretação sobre qual seria a manifestação física ou visual do monstro.[125]

[124] Convém notar que, seguindo esta mesma lógica, é possível argumentar que um escritor também trabalha com base em uma série de recursos preexistentes, porém de natureza conceitual, na forma das ideias, saberes e costumes compartilhados socialmente.

[125] O MissionMaker não oferecia muitas opções pelas quais os participantes podiam representar a sua interpretação dos personagens, principalmente para Grendel e sua mãe. Por outro lado, o aplicativo oferecia outras formas pelas quais era possível caracterizar os personagens, como os diálogos, sons e o seu comportamento.

Isso tudo contribuiu para que os participantes pudessem, no dia seguinte, abordar a atividade de criação de jogos com uma compreensão bastante rica e aprofundada do poema, não apenas do seu enredo e dos eventos que se passam na estória, mas também dos seus temas, a personalidade dos seus personagens e a própria lógica do seu universo ficcional.

Niamh também observou que, em muitas das atividades criativas do primeiro dia, os participantes eram colocados no papel de personagens ativos nas diferentes cenas e situações do poema. Isso os ajudou a compreender de uma maneira prática a perspectiva do jogador em um videogame, e sua função de centralizar e catalisar os eventos e elementos do jogo.

3.2.7 Comer, Beber, Dormir (Tracey Matthews)

Figura 28 – Comer, Beber, Dormir (Tracey Matthews)

"Comer, Beber, Dormir" ("Eat, Drink, Sleep"), de Tracey Matthews, se passa em Heorot, o grande salão de banquetes dos Danos, constante alvo de ataques de Grendel (fig. 28). Restos de comida e bebida sugerem que aconteceu uma grande festa. A missão do jogador, como Beowulf, é ajudar os guerreiros a proteger o salão. Beowulf pode conversar com alguns destes guerreiros - um deles, quando endereçado, faz referência a um trecho do poema: "Comer, Beber, Dormir...". Explorando o salão, o jogador pode encontrar uma arma, e um lugar para dormir. Se escolher dormir, mais tarde acorda surpreendido por uma batalha no salão, entre Grendel e os guerreiros. O jogador ganha o jogo se derrotar o monstro.

O jogo de Tracey apresenta dispositivos narrativos interessantes e sofisticados, e trata-se de uma das produções menos lineares entre as que analisamos. O jogador é encorajado a explorar o ambiente livremente, ao invés de ser guiado em direção a uma determinada sequência de acontecimentos.

Tracey tem um perfil diferente dos demais participantes da oficina. Ela é de uma geração anterior, tem mais experiência com ensino, porém tem dificuldades em acompanhar as novas tecnologias. Por outro lado, ela demonstrou conhecer bem os conceitos e a lógica por trás dos jogos digitais, e tinha ideias claras e bem elaboradas para o seu projeto. Ela acabou tendo a ajuda dos outros participantes da oficina, além dos monitores e organizadores, principalmente na implementação das regras e comportamentos dos personagens.[126]

[126] A dinâmica de trabalho na oficina era livre e flexível. Todos os participantes ajudavam de alguma maneira nos projetos uns dos outros.

O resultado foi que este projeto teve um fluxo de trabalho similar ao de um estúdio de desenvolvimento de jogos (embora evidentemente em escala muito menor), com Tracey atuando principalmente como diretora e designer. Ela planejou o projeto garantindo a coerência entre os diferentes elementos e partes do jogo, enquanto que outros participantes ajudavam pontualmente na implementação sempre que necessário.

A experiência de Tracey com jogo digitais veio principalmente de observar os seus próprios filhos jogando. Isso inclui títulos como o jogo de ação Grand Theft Auto (Rockstar Games) e a série de FPS de guerra Call of Duty (Activision), bem como o popular Minecraft (Mojang, 2009), no qual o ambiente é criado proceduralmente e o jogador pode explorar e fazer suas próprias criações. Entre os jogos que Tracey de fato jogou, ela cita o adventure Broken Sword (Revolution Software, 1996) e o primeiro Tomb Raider (Eidos Interactive, 1996), que ela prefere por terem um ritmo mais lento e "gentil".[127]

Processo Criativo

De acordo com Tracey, ela optou por uma abordagem "sistêmica" no design do seu jogo. Isto é, ao invés de definir ideias ou elementos específicos, ela dedicou-se a desenvolver um conceito claro a respeito da situação que queria representar. Uma vez tendo isso, ela identificou algumas diretrizes gerais com base nas quais começou a definir os diferentes elementos que iriam compor o jogo - localidades, objetos e personagens - bem como as situações, comportamentos e eventos mais específicos.

O ambiente principal do seu jogo representa Heorot, o salão de banquetes e festas, e foi criado com base nas descrições do local encontradas no poema original. Tracey também incluiu itens e objetos para ajudar na caracterização do ambiente, como frutas, pedaços de carne e garrafas de vinho - como se tivesse acabado de acontecer uma festa no salão. Outro elemento importante que ajudou na caracterização foi a inclusão de múltiplos personagens no ambiente, representando os outros guerreiros. O salão, além de ser um local para festas, para comer e beber, também é um lugar em que os guerreiros podem socializar e dormir.

Tracey foi uma das poucas participantes da oficina a explorarem o potencial de ter um número maior de personagens simultaneamente em um mesmo ambiente. A maioria dos outros jogos tinha apenas dois personagens - geralmente o rei, no início, e a mãe de Grendel no final -, ou no máximo três (o terceiro representando mais um desafio antes do conflito final com o monstro).

[127] Tomb Raider é um jogo de ação em 3D, mas que também tem um forte elemento de exploração.

No jogo de Tracey, além de existirem múltiplos personagens logo de início, estes não eram inimigos mas sim aliados do jogador. Além disso, cada um dos personagens apresenta um comportamento próprio, embora simples, produzindo uma dinâmica natural e orgânica no grupo.

Apesar de ter ficado satisfeita com o resultado final do seu jogo, Tracey queria ter mais tempo para trabalhar nele. Por exemplo, ela gostaria de ter explorado em maior profundidade os outros temas introduzidos no diálogo "Comer, beber, dormir", incluindo mais elementos e "affordances" (possíveis ações e interações) relacionadas a estas atividades. O jogador pode encontrar uma cama e "dormir" - esta ação leva ao aparecimento de Grendel. Porém, embora Tracey tenha incluído alguns objetos que fazem alusão aos outros temas - a garrafa de vinho para "beber" e os pedaços de carne para "comer" -, estes não tem uma função ou repercussão na lógica do sistema do jogo.

Experiência Não Linear

Vimos que a maioria dos jogos da oficina foram estruturados de forma que o jogador estivesse sempre avançando de maneira linear, de um desafio para o outro. Isto foi realizado principalmente através da topologia do espaço físico. Por exemplo, ao invés de ambientes amplos ou abertos, e contendo múltiplas conexões e passagens entre si, foram utilizados espaços menores, interligados por corredores.

Outro recurso empregado para controlar o fluxo nestes jogos foi a existência de eventos ou ações que precisam acontecer para ativar ou tornar disponíveis os próximos elementos ou espaços de acordo com a sequência predeterminada. Por exemplo, no jogo de Andrew Smith, de forma a garantir que o jogador irá conversar com o rei antes de enfrentar a mãe de Grendel, o portal que leva até o covil se encontra inicialmente desativado, e só se torna ativo após o diálogo acontecer.

O jogo de Tracey se distingue dos demais neste sentido. Não existe um percurso e tampouco um objetivo específico a ser cumprido pelo jogador. A única premissa estabelecida é que o jogador é Beowulf, e que ele se encontra no salão juntamente com outros guerreiros, disposto a ajudá-los a se protegerem de um possível ataque de Grendel. Embora exista uma condição de "vitória" - se Grendel for derrotado - esta não é apresentada como um objetivo propriamente dito. Em outras palavras, no jogo de Tracey, o jogador não é apresentado a uma "missão" mas sim a uma "situação".

Ao contrário da topologia predominantemente linear da maioria dos outros jogos, no projeto de Tracey o ambiente do salão convida o jogador à exploração, sem impor um trajeto específico. A área central do salão é bastante grande, com o pé direito alto. Há uma escada no fundo, que sobe para um corredor de onde o salão pode ser visto do alto - este corredor dá acesso aos cômodos do

andar de cima. No salão há um balcão de bar grande, e por trás dele há um corredor que leva a um quarto nos fundos. Finalmente, através de outro corredor, saindo do salão principal, é possível alcançar uma escada que desce até uma câmara subterrânea. Nesta câmara, há uma superfície de madeira que serve como uma cama, e que pode ser usada por Beowulf para dormir.

A maneira como o espaço do jogo foi populado também influencia fortemente o ímpeto para a exploração. Isso inclui desde a maneira como os objetos estão dispostos, até a forma como os outros personagens se deslocam pelo espaço e interagem entre si, de maneira fluída e orgânica.

Estes elementos fazem com que o jogador se sinta em um mundo coeso, no qual o fluxo do jogo é guiado pelas suas escolhas, ao invés de um roteiro predefinido pelo designer.

Lealdade Procedural

O jogador pode conversar com alguns dos outros guerreiros que estão circulando pelo salão. Tratam-se de mensagens breves e simples, através das quais cada personagem fornece uma certa informação ou comentário, dando mais contexto sobre a situação em que eles se encontram.

Um destes personagens, após conversar com Beowulf, passa a seguir o herói, de maneira sutil. Isto é, ele não o segue diretamente, mas sim de uma maneira livre, e com alguma variação. Na opinião de algumas das pessoas que testaram o jogo, o efeito é de que existe algum tipo de "intenção" que distingue este guerreiro dos demais - alguns se perguntavam "o que ele quer?", enquanto outros faziam algum comentário, como “esse cara é esquisito...".

Este comportamento - fazer um personagem seguir outro desta maneira - já vem predefinido no MissionMaker. O papel de Tracey foi na definição de uma regra que ativa este comportamento após a conclusão do diálogo entre o personagem e o jogador. Isso não significa, entretanto, que ela tem necessariamente menos mérito ou crédito sobre este elemento do jogo.

De uma maneira geral, toda prática criativa envolve ferramentas preexistentes, desde o pincel utilizado pelo pintor até o instrumento empregado por um músico. Além disso, existem as ferramentas conceituais e o conhecimento acumulado associado a cada forma expressiva (técnicas, estilos e assim por diante), e o próprio repertório de quem cria. Em se tratando das mídias digitais, por conta da complexidade das tecnologias envolvidas, o emprego de ferramentas e elementos preexistentes é ainda mais comum, e mesmo inevitável. A atuação do designer muitas vezes se dá justamente no ato de escolher os componentes e elementos que irão compor a sua produção, bem como a maneira como eles serão empregados e eventualmente combinados, e em que contexto.

No jogo de Tracey, por exemplo, embora o comportamento de "seguir" não seja de sua autoria, há um significado particular pretendido neste uso específico, e neste contexto particular. A sua intenção era de expressar a ideia da **lealdade**, representando o fato de que os guerreiros respeitam a figura de Beowulf, e confiam na sua liderança diante da ameaça iminente de Grendel.

Além disso, o comportamento reflete o próprio tema do diálogo que acontece antes da sua ativação. O resultado é uma ressonância entre, de um lado, um recurso referente à forma expressiva procedural (o comportamento), e do outro, um conteúdo referente a uma forma expressiva tradicional (da escrita verbal - o diálogo do personagem).

Tópicos Recorrentes

A seguir, apresentaremos três tópicos recorrentes identificados ao longo das entrevistas realizadas com os participantes da oficina, relacionados ao processo criativo dos jogos no MissionMaker.

Examinaremos (1) a perspectiva que alguns dos usuários tiveram sobre o design no MissionMaker acontecer "ao contrário"; (2) como a abordagem didática no fluxo de trabalho deste aplicativo se reflete no processo de criação dos jogos; e (3) algumas das estratégias adotadas pelos participantes da oficina para representar nos jogos um aspecto específico encontrado no poema - a possível ambiguidade entre a representação do bem e do mal.

Design "ao Contrário"

Durante as entrevistas conduzidas com os participantes da oficina, em mais de uma ocasião o processo de design do jogo foi descrito como "ao contrário". Este comentário faz referencia especificamente ao processo necessário para a criação de uma regra no MissionMaker, que estaria de alguma forma "invertido", por conta de exigir que o designer considerasse primeiro os resultados ou consequências de uma regra, para só depois lidar com a causa.

De forma a ilustrar esta questão, apresentaremos a seguir os passos para a criação de uma regra simples no MissionMaker, que serve para abrir uma porta se o jogador clicar em uma alavanca.[128]

1. Selecionar a porta (fig. 29, abaixo);

Figura 29 – Passo 1: seleção da porta

[128]Para este exemplo nos concentraremos em alguns elementos específicos da interface do aplicativo. Apresentamos uma perspectiva mais geral da interface no início deste capítulo (p. 98).

2. Acessar o menu "ações" (referente à porta; fig. 30, abaixo);

3. Neste menu, selecionar a ação "abrir" (Open);

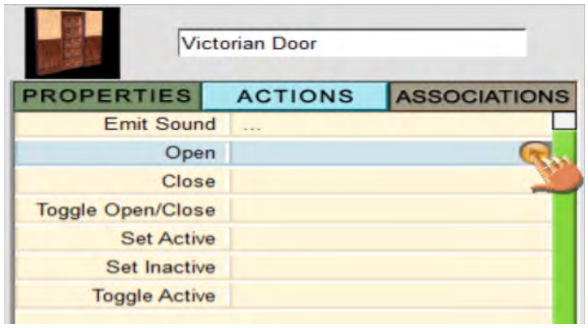


Figura 30 – Passos 2/3: função "abrir"

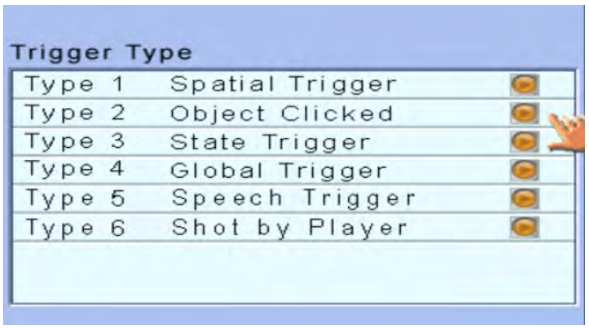


Figura 31 – Passo 4: tipo de ativador

4. Selecionar o tipo de ativador (trigger) para a regra: "Object Clicked" (fig. 31, acima);

5. Selecionar o objeto que deve ser clicado: isto é, a alavanca.

As figuras a seguir mostram a regra completa (fig. 32) e o seu funcionamento (fig. 33).

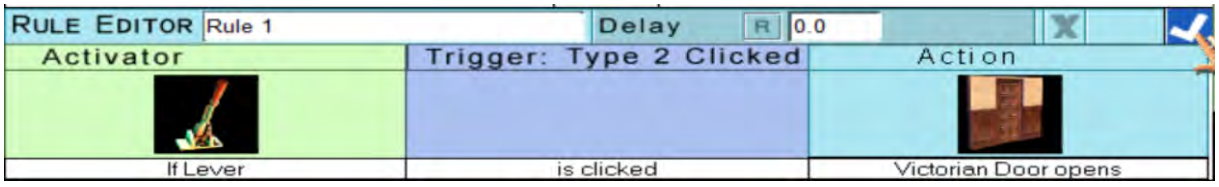


Figura 32 – Regra completa

Figura 33 – Rega em funcionamento no jogo

Para alguns dos usuários este processo está "ao contrário" porque contradiz o que imaginam ser uma sequência "natural" na qual um designer criaria esta regra. Entretanto, a rigor não existe uma única maneira de realizar esta operação. Entre os próprios entrevistados, pudemos identificar diferentes opiniões quanto a qual seria uma implementação "correta" para este processo.

Por exemplo, para alguns, o primeiro passo estava correto (selecionar a porta), uma vez que este é o objeto que a regra irá afetar (isto é, abrir). Outros consideram que este não deveria ser o primeiro passo, uma vez que a porta é apenas um dos parâmetros da regra (além da alavanca e do clique do mouse). Para estes, a entidade principal desta regra seria a ação "abrir" em si. Nesta perspectiva, o primeiro passo para a criação da regra deveria ser a seleção da ação "abrir", e só depois os demais elementos e parâmetros poderiam ser escolhidos.

Outro elemento que alguns dos entrevistados consideraram pouco intuitivo é o fato de que o ativador da ação é definido no final do processo, após a escolha da ação. Para estes usuários, faz mais sentido se as operações realizadas pelo designer para criar a regra seguissem uma lógica similar aos gestos e ações do jogador ao interagir com ela. Isto é, sob esta perspectiva, faria mais sentido primeiro selecionar a alavanca (que é o objeto com o qual o jogador irá interagir), depois escolher que tipo de interação deve ser considerada (neste caso, um clique), qual ação ela deve ativar ("abrir") e finalmente o objeto desta ação (a porta).

Abordagem Didática

Uma das principais dificuldades encontradas por alguém que está começando a aprender programação é o caráter fundamentalmente abstrato da lógica computacional. Ao contrário de atividades como desenho e música, nas quais os gestos físicos se refletem diretamente na materialidade do que está sendo criado, a programação de computadores envolve lidar com abstrações, que não se relacionam de maneira intuitiva ao produto ou comportamento resultante do algoritmo. Por conta disso, uma abordagem comum no ensino da programação, principalmente para crianças, é evitar operações ou funções que não levam a resultados imediatos na execução do programa.[129]

De uma maneira geral, o fluxo de trabalho no MissionMaker é relativamente intuitivo. O editor permite ao usuário trabalhar diretamente com base na perspectiva em 3D do jogo, deslocando-se pelo espaço, selecionando, modificando e criando objetos e elementos. Por exemplo, para incluir uma porta em uma determinada passagem, basta caminhar até ela e, quando a área em questão

[129] Neste contexto, é compreensível que alguns usuários descrevam o processo de criação de regras no MissionMaker como sendo "ao contrário". Embora não exista uma maneira "correta" ou "melhor" para implementar um determinado processo, certamente existem formas mais **didáticas** para fazê-lo.

estiver visível, identificar o ícone correspondente no menu e arrastar ele para o local. O mesmo para posicionar e modificar objetos, personagens e assim por diante.[130]

Este modelo não é exclusivo ao MissionMaker. Outros aplicativos e plataformas de criação de jogos utilizam abordagens similares, embora implementados de maneiras diferentes. Na prática, é como se o designer pudesse interagir com o seu próprio jogo, no papel de um jogador com "superpoderes".

Além disso, os passos que o usuário deve executar de forma a criar ou modificar algum elemento ou regra no jogo geralmente também acontecem na sequência que se esperaria no mundo físico. Por exemplo, não é possível criar um personagem sem antes criar um espaço físico no qual ele será posicionado. O mesmo se aplica aos objetos e outros elementos, e nas relações entre eles. Assim, digamos que o designer queira que um objeto esteja dentro de outro, como uma garrafa dentro de uma geladeira, por exemplo. É necessário primeiro criar e posicionar a geladeira, **abrir** a sua porta, e então criar e posicionar o objeto da garrafa no seu interior.

Esta é uma limitação arbitrária, baseada na suposta fisicalidade destes elementos. Um designer mais experiente pode considerar este processo conturbado e ineficiente. Afinal, um dos benefícios da criação em meio digital é justamente a flexibilidade que vem do fato de que este meio não se encontra submetido às limitações do mundo físico, e dos processos criativos tradicionais. Entretanto, para alguém aprendendo os fundamentos, este fluxo de trabalho mais didático pode facilitar o processo.

Mesmo no caso do processo para a criação de uma regra, tratado anteriormente, o objetivo por trás desta implementação particular no MissionMaker foi justamente de definir a sequência de passos que refletiria isso da maneira mais intuitiva possível, considerando o público alvo pretendido.

Representando a Ambiguidade

Uma característica comum à maioria dos jogos criados na oficina é o fato de que o inimigo de Beowulf - em alguns casos a mãe de Grendel, e em outros o próprio monstro - não ataca o jogador a não ser que seja atacado, desafiado ou provocado (por exemplo, se o jogador invadir o seu covil). Alguns dos entrevistados declararam que decidiram implementar este comportamento como uma maneira de refletir uma ambiguidade que existe no poema original.

De acordo com algumas interpretações de Beowulf, o poema não deixa claro quem é o herói e quem é o monstro na estória. O comportamento agressivo de Grendel poderia ser justificado pelo seu instinto de preservação, enquanto que sua mãe poderia estar apenas reagindo ao assassinato do

[130] O editor também oferece uma perspectiva em planta (do topo), que geralmente é mais adequado para planejar a estrutura do espaço do jogo.

filho. Da mesma forma, as estórias sobre o guerreiro Beowulf, seus feitos, sua força e heroísmo podem ser interpretadas como a descrição de um monstro impiedoso. O escritor americano John Gardner explora esta perspectiva no romance Grendel (1971), no qual o monstro é apresentado como um anti-herói.

Este aspecto do poema foi implementado no contexto dos jogos de diferentes maneiras pelos participantes da oficina. Por exemplo, alguns interpretaram a ideia da ambiguidade como uma ausência de dicotomia na estória (isto é, considerando que não há heróis, e tampouco monstros), outros mantiveram uma perspectiva mais tradicional do poema original (Beowulf como herói) e alguns simplesmente inverteram os papéis (Beowulf como o monstro). No momento de representar a ideia da ambiguidade através da mecânica de jogo, alguns apenas desativaram a capacidade de ataque do (suposto) inimigo, outros criaram uma regra para ativá-la apenas se o jogador invadir uma certa área, e houve um participante que programou um comportamento dinâmico, que fazia a mãe de Grendel atacar apenas nos momentos logo após ter sido agredida.

Alguns dos entrevistados relataram que a ideia de sugerir esta ambiguidade na mecânica do jogo teve uma ajuda do acaso. Inicialmente, de forma a aprender as funcionalidades do aplicativo, muitos deles criaram um personagem representando um "inimigo" (que posteriormente viria a ser Grendel, ou sua mãe, por exemplo) e atribuíram a ele um comportamento simples de "matar o jogador" (que já vem predefinido no MissionMaker). Porém, ao testar o jogo, notaram que o inimigo estava abandonando a sua posição original, à procura do jogador (de forma a cumprir sua "missão"). Isso impedia os designers de controlar de maneira adequada o fluxo do jogo - por exemplo, a maioria deles queria que o monstro esperasse pelo jogador no covil.

Eventualmente eles descobriam que era possível desativar este comportamento no início do jogo, e ativá-lo apenas quando necessário. Alguns dos designers programaram a regra de forma que ela simplesmente se ativasse com a presença do jogador. Entretanto, muitos viram uma oportunidade de explorar este comportamento de uma maneira menos tradicional e mais sofisticada, refletindo a questão da ambiguidade sobre o comportamento e os relacionamentos dos personagens no poema.

Por exemplo, outra proposta que surgiu para representar a ambiguidade de Beowulf na forma de uma mecânica de jogo foi inverter o papel sendo representado pelo jogador. Ao invés de controlar Beowulf, o jogador passaria a controlar Grendel (ou a mãe dele). Esta possibilidade foi discutida por alguns dos participantes durante os estágios iniciais de planejamento e desenvolvimento dos jogos na oficina. Alguns pensavam em estabelecer esta identidade do jogador deste o início (por exemplo, através de um texto de introdução, ou na forma de um diálogo ou narração), enquanto que outros

tiveram a ideia de explorar esta revelação de uma maneira mais expressiva ou dramática, mais adiante no jogo. Isso seria possível por conta do fato de que os jogos são em primeira pessoa, e portanto a princípio o jogador nunca vê a si mesmo.

Uma das ideias envolvia colocar o jogador no próprio castelo dos Danos, sendo atacado por seus guerreiros, porém sugerindo que trata-se de uma estória alternativa, na qual Beowulf estaria em conflito com eles. No final do jogo, após matar todos os guerreiros, seria revelado que o jogador estava representando Grendel desde o início.

Outra variação mais complexa discutida para esta ideia foi de trocar o personagem sendo controlado pelo jogador em algum momento durante o próprio jogo. Por exemplo, o jogador poderia começar o jogo como Beowulf, em uma missão para encontrar e matar Grendel. Porém, após vencer uma série de desafios e finalmente ficar frente a frente com o inimigo em seu covil, os papéis se inverteriam - o jogador passaria a controlar Grendel (com a nova missão de se defender de Beowulf).

A intenção seria fazer o jogador se esforçar para chegar até o covil, e assim valorizar esta conquista (sua e do seu avatar, Beowulf). Desta forma, quando os papéis se invertem, e o jogador passa a controlar Grendel, o personagem de Beowulf não seria apenas um inimigo qualquer. Matar Beowulf seria um ato significativo, por conta da conexão que tinha sido feita antes entre o jogador e este personagem. A ambiguidade se realizaria no momento desta percepção, combinada com o fato de que o jogador agora controla Grendel.

3.3 Análise do Aplicativo: MissionMaker

Nesta seção apresentaremos o estudo prático sobre o MissionMaker, isto é, o aplicativo empregado para a criação dos jogos analisados anteriormente. Nossa análise terá como ênfase a maneira como os mecanismos e funções relativas à autoria procedural são implementadas e disponibilizadas ao designer no aplicativo.[131]

Primeiro apresentaremos uma metodologia (ou área de estudos) chamada **Análise Multimodal**, da qual emprestamos parte das ferramentas teóricas que empregamos neste estudo (p. 151). Esta metodologia nos ajudará a identificar e examinar quais são os **recursos** e mecanismos disponíveis ao designer no MissionMaker (p. 155), e como eles são organizados dentro da gramática particular deste aplicativo a partir do ato da **composição** (p. 185). Também mostramos como pode ser feita uma **comparação** entre os jogos digitais e outros meios e linguagens tradicionais, a partir dos recursos compartilhados entre elas (p. 189), de forma a investigar as estratégias expressivas únicas oferecidas pela autoria procedural.

Como vimos, a oficina Playing Beowulf envolveu dois dias de atividades criativas. O primeiro dia se concentrou em linguagens e meios tradicionais, como Artes Dramáticas, Desenho e Cinema, enquanto que o segundo foi dedicado exclusivamente ao exercício de criação de jogos com o MissionMaker.

De acordo com o relato de muitos dos participantes da oficina, as atividades do primeiro dia foram fundamentais para que eles pudessem construir a sua própria interpretação da estória, incluindo a personalidade dos personagens, os relacionamentos entre eles, e os significados e conexões entre os diferentes eventos e situações retratadas. Assim, no segundo dia eles puderam se concentrar no aprendizado e familiarização com as funcionalidades e particularidades do MissionMaker.

Como veremos mais adiante, esta relação entre o primeiro e segundo dia da oficina tem particular relevância no contexto deste estudo sob o ponto de vista do processo criativo. Ao considerar os recursos e estratégias compartilhadas entre a criação de jogos e as outras atividades criativas tradicionais, é possível compreender melhor as particularidades da autoria procedural de uma maneira mais ampla, e como esta prática se distingue das demais linguagens e formas expressivas.

[131] Lembrando que há uma introdução para a interface e fluxo de trabalho do aplicativo no início deste capítulo (p. 97).

3.3.1 Análise Multimodal (Metodologia)

Os estudos práticos apresentados neste capítulo foram realizados a partir de duas abordagens - de um lado, uma metodologia própria, desenvolvida na etapa anterior desta pesquisa, em nível de mestrado; do outro, os conceitos e teorias de uma área relativamente nova de investigação chamada Análise Multimodal. Estas abordagens são complementares, e permitem perspectivas únicas e valiosas sobre a autoria e a expressividade procedural.

Nossa própria metodologia, apresentada anteriormente (p. 79), fornece mecanismos teóricos para a análise das mídias digitais (e das produções realizadas neste meio) de uma maneira mais fundamental, a partir dos níveis mais "baixos" de abstração da lógica computacional.

A Análise Multimodal, por sua vez, fornece a base para analisar o aspecto procedural do meio digital como uma linguagem em si, lado a lado às demais formas e linguagens tradicionais. Nesta perspectiva, a proceduralidade é considerada principalmente a partir das suas estruturas e funções em um nível mais "alto" de abstração, tais como aquelas presentes na camada do software, e outras implementações e usos mais específicos, como veremos.

A seguir apresentaremos em maiores detalhes a metodologia da Análise Multimodal, e a maneira como ela é empregada neste estudo.

Modos

Um dos principais desafios enfrentados ao longo desta pesquisa foi o próprio processo de definir a natureza do aspecto procedural do meio digital. Desde a etapa anterior deste estudo, temos investigado a autoria procedural, identificando as suas propriedades e estratégias. Esta análise segue na presente pesquisa, com um foco específico na narratividade e jogos digitais. Neste percurso, pudemos verificar que existem múltiplas perspectivas através das quais a proceduralidade pode ser considerada e investigada, em termos do seu papel no processo criativo em meio digital.[132]

Durante a etapa sanduíche deste estudo, tivemos contato com a área da Análise Multimodal (ou *Multimodal Analysis*). Trata-se de uma metodologia relativamente recente para o estudo da linguagem, com foco na construção de **significado**, e nas diferentes formas pelos quais ele pode se manifestar através dos diferentes **Modos** comunicativos, bem como entre eles (JEWITT, 2014). No contexto da Análise Multimodal, o termo "Modo" representa uma perspectiva mais ampla para o conceito de linguagem (a grafia com a primeira letra em maiúsculo é empregada nesta pesquisa de forma a diferenciar este do uso mais comum da palavra "modo").

[132] Tratamos sobre este percurso anteriormente na tese (p. 42).

O termo "linguagem" tende a ser empregado para determinadas manifestações comunicacionais específicas, principalmente a linguagem verbal (em muitos casos, considerada como sinônimo de "linguagem"), além de outras formas, como as linguagens visual, sonora, musical e assim por diante.

A ideia de Modo, por outro lado, se aplica a qualquer gesto ou manifestação humana capaz de produzir significado, para propósitos práticos ou expressivos. Por exemplo, o "olhar" pode ser considerado como um Modo, uma vez que permite ao indivíduo se comunicar - indicando um objeto de interesse apenas olhando para ele, por exemplo - e se expressar - um olhar distante, sem um alvo específico, sugere contemplação, por exemplo.

Assim, o conceito do Modo permite uma grande flexibilidade e fornece uma base ampla e completa para analisar os atos de comunicação, representação e expressão, a partir daquilo que eles de fato significam, ou realizam. Esta abordagem fornece uma perspectiva alternativa, e que não se prende aos modelos e conceitos de linguagens já estabelecidas. De uma maneira geral, a natureza dinâmica e flexível da Análise Multimodal é particularmente útil para a investigação das mídias digitais, um meio relativamente novo, principalmente se comparado às formas e linguagens tradicionais mais consolidadas.

Como veremos mais adiante, a definição de cada Modo específico é fundamentalmente arbitrária, e se dá a partir de uma estrutura hierárquica (ou aninhada). Por exemplo, o Modo do "olhar" pode ser considerado como parte do Modo maior (ou mais amplo) da linguagem corporal. Mas o olhar também pode ser decomposto em Modos menores - na sua direção, foco ou rapidez, por exemplo.

No contexto da Análise Multimodal, estes diferentes elementos que compõem um Modo são chamados de recursos semióticos, um conceito que se refere a qualquer elemento psicológico ou tecnológico que pode ser usado para comunicar, representar ou expressar uma ideia, mensagem ou significado (LEEUWEN, 2005). Sob esta perspectiva, o conceito de Modo pode ser definido como qualquer conjunto de recursos empregados para a construção de significado.

A metodologia da Análise Multimodal tem como origem e se baseia no campo mais amplo da Semiótica Social. Na presente pesquisa, nossa intenção não é de aprofundar nesta área, mas sim unicamente de emprestar alguns dos conceitos e teorias da Análise Multimodal, tomados de maneira mais simplificada, e recontextualizados para o nosso tema e recorte.

Assim, em nosso estudo o conceito do recurso semiótico é considerado principalmente a partir da sua analogia com a função da palavra dentro da linguagem verbal - isto é, como uma unidade de composição da frase. De forma a refletir este emprego mais simplificado deste conceito, nos

referimos aos recursos semióticos como “recursos modais", ou apenas "recursos", de acordo com o contexto (como veremos, os recursos modais em meio digital podem ser tanto procedurais quanto estáticos).[133]

Outra ideia útil que vem da aplicação da abordagem Multimodal no estudo de manifestações comunicacionais é a noção de que a **escolha de Modo** trata-se de um ato significativo em si mesmo. Isso será particularmente relevante mais adiante, ao discutirmos como os diferentes Modos se relacionam entre si, identificando tanto os recursos modais compartilhados entre eles quanto aqueles únicos a cada um. Veremos também como este conceito se relaciona à análise das intenções por trás de atos criativos particulares, permitindo compreender em maior profundidade como se dá o emprego da autoria procedural por parte dos artistas e criativos atuando em meio digital.

Convém reiterar que, ao analisar este uso específico do meio digital - para a criação de jogos utilizando o MissionMaker -, nossa intenção principal é de acessar e investigar aspectos e elementos da autoria procedural de uma maneira mais ampla.

Proceduralidade como Modo

Vimos anteriormente que, de acordo com Janet Murray, o emprego expressivo de um meio se dá a partir do uso de mecanismos que lhe são próprios e únicos, enquanto que o emprego aditivo se refere ao uso de estratégias expressivas de outros meios (p. 25).

Este emprego aditivo pode ser descrito na terminologia da Análise Multimodal a partir da ideia da "combinação de Modos", ou do conceito de um "Modo orquestral" (ou "orchestrating MODE"; BURN, 2014). Isto é, a situação na qual o computador opera unicamente na organização e apresentação de conteúdo de diferentes Modos em um mesmo contexto - espacial, temporal e assim por diante -, coexistindo e operando cada qual através do seu próprio mecanismo e lógica.

Esta perspectiva não nos interessa nesta pesquisa. Como vimos, nosso foco se encontra no emprego das mídias digitais como um **meio expressivo**, a partir da autoria procedural. Assim, no contexto da Análise Multimodal, consideramos a proceduralidade como um Modo em si mesmo (isto é, independente), a partir do seu próprio potencial expressivo.

[133] Outro conceito do campo da Semiótica Social que faz parte da Análise Multimodal é o "potencial de significação", definido como a representação de uma ideia que tem o potencial de ser interpretada por um determinado público em um certo contexto. Como o nosso foco é na perspectiva do processo criativo, consideramos o significado principalmente a partir da intenção do criador (reconhecemos a ideia da abertura de interpretação, entretanto ela não é o nosso foco). O mesmo se aplica à noção de um "potencial expressivo", no caso da construção de significado a partir de uma intenção expressiva.

3.3.2 Recursos Modais Procedurais

> Eu trabalho com declarações, variáveis, *loops*, condicionais, funções, objetos e *arrays*.
>
> (REAS, 2015)

A proceduralidade pode ser definida a partir da sua relação com os dados estáticos - enquanto que dados estáticos representam informações e conteúdo diretamente (através de elementos como palavras, números, tabelas, imagens e som), a proceduralidade representa regras e comportamentos.

De um lado, o computador pode ser empregado para emular meios e linguagens tradicionais, criando e manipulando conteúdo estático - é o caso do desenhista que lida com pixels ou curvas vetoriais, o escritor que lida com palavras ou o músico que lida com ondas sonoras ou partituras. Nestes casos, embora a ferramenta empregada seja o meio digital, o processo criativo se dá a partir da expressividade de **formas tradicionais** - Artes Visuais, Literatura, Música e assim por diante.

Do outro lado, as mídias digitais podem ser empregadas a partir das estratégias procedurais de criação, através das quais pode se realizar o potencial expressivo **próprio** deste meio. Como descreve o artista Casey Reas, um dos criadores da linguagem de programação Processing, isso se dá através de elementos como declarações, funções e estruturas de dados - conteúdo de natureza fundamentalmente procedural.[134]

No MissionMaker, um exemplo comum de autoria procedural é a criação de uma regra para abrir uma porta de acordo com uma condição, como vimos anteriormente (p. 143). Neste caso, enquanto que a regra é um comportamento dinâmico, a porta é um elemento estático. Como veremos mais adiante, esta distinção nem sempre é tão clara, especialmente da perspectiva dos recursos modais.

Nas páginas a seguir, examinaremos os recursos modais procedurais no contexto do MissionMaker, bem como em relação à lógica computacional de uma maneira mais ampla. Elaboraremos sobre os diferentes **níveis de abstração** no meio digital (p. 156), veremos como os recursos procedurais são realizados ou materializados no contexto da sua **execução** (p. 158) e examinaremos as implicações e significados da existência de **múltiplas contribuições** no processo criativo em mídias digitais (p.

[134] Note a diferença entre um determinado conjunto de números (conteúdo estático), por exemplo, e os modelos e processos pelos quais eles são formatados e processados (conteúdo procedural). Mais adiante aprofundaremos sobre esta distinção. Epígrafe: tradução nossa - texto original: "I work with statements, variables, loops, conditionals, functions, objects, and arrays." (sem numeração de página)

162). Introduziremos a **perspectiva de análise** que adotamos sobre os mecanismos do MissionMaker (p. 164), e apresentaremos os dois recursos modais principais disponíveis ao usuário neste aplicativo - **as variáveis e as condicionais** (p. 166). Finalmente, apresentaremos os recursos presentes no **engine** do aplicativo, desde os níveis mais altos de abstração, até os mais baixos, e examinaremos as possíveis implicações e significados associados a eles (p. 170).

Níveis de Abstração

Os sistemas dos jogos digitais, assim como de qualquer outro aplicativo (ou programa), são compilados a partir de algoritmos. Estes, por sua vez, consistem em séries de instruções que representam abstrações particulares de determinados processos. Existem diferentes maneiras pelas quais é possível modelar e implementar um mesmo processo.

É possível fazer uma analogia com um mapa, que representa uma abstração particular de um lugar. Uma mesma região pode ter múltiplos mapas, cada um representando uma visão ou perspectiva particular sobre ela. Por exemplo, um mapa pode representar os diferentes tipos de terreno em um local, outro pode mostrar apenas as fronteiras entre diferentes regiões e outro pode fornecer uma interpretação artística mais subjetiva do mesmo espaço.

Algoritmos são similares, no sentido em que uma mesma ideia ou processo pode ser representado (ou implementado) em uma variedade de formas, de acordo com as perspectivas e propósitos particulares do designer (ou programador). Entretanto, diferentemente de mapas, um programa de computador permite a representação de sistemas, regras e comportamentos, de maneira procedural.

Nesta seção, elaboraremos sobre o conceito de níveis (ou graus) de abstração que temos empregado ao longo desta tese. Este conceito será particularmente útil nas próximas seções, quando investigaremos os diferentes recursos modais disponíveis no editor do MissionMaker, bem com no seu engine.

Existem diferentes níveis de abstração nos quais um determinado objeto ou conceito pode ser representado ou modelado. Na analogia do mapa, mencionada acima, considerando que o objeto original é a região sendo representada, uma abstração de baixo nível seria uma representação realista, precisa e completa do local (incluindo a topologia em 3D, e todos os demais aspectos e elementos relacionados, bem como suas propriedades). Uma abstração de nível médio poderia ser um mapa tradicional, em 2D, mostrando as divisórias entre regiões, estradas e assim por diante. Uma possível abstração de nível mais alto seria um gráfico representando as proporções entre os diferentes tipos de terreno, por exemplo.

Note como que, à medida em que passamos para os níveis mais alto de abstração, a maioria dos elementos e aspectos do objeto original são removidos ou diminuídos, enquanto que alguns deles são mantidos e enfatizados. No último "mapa" descrito acima, por exemplo, não há informação sobre o formato de cada região, mas em compensação a informação específica quanto à proporção da área de cada tipo de terreno está presente de maneira clara, algo que não existe no mapa tradicional, nos níveis mais baixos de abstração (principalmente se o formato das regiões for irregular).

No meio digital, o nível mais baixo de abstração geralmente é considerado como sendo a linguagem de máquina, que dá acesso à lógica do hardware (embora, a rigor, pode-se dizer que o hardware em si estaria em um nível mais abaixo). A partir daí, existem as linguagens de programação e elementos de lógica de nível mais alto, que oferecem abstrações adicionais (tanto sobre a linguagem de máquina, quanto sobre linguagens e elementos de níveis mais baixos). Em um nível mais acima, existem os aplicativos, que oferecem uma abstração ainda maior para lidar com a lógica computacional (e os dados estáticos), geralmente através de interfaces visuais mais elaboradas.

Por um lado, cada novo nível de abstração permite uma perspectiva, acesso ou controle particular sobre os mecanismos da máquina, que não seriam possíveis (ou viáveis) em um nível mais baixo. Por exemplo, um aplicativo gráfico fornece uma interface visual com abstrações que simulam canetas, pincéis e assim por diante, permitindo a um desenhista produzir um tipo de imagem que nunca poderia ser criada de outra forma neste meio.

Por outro lado, isso é feito ao custo de uma restrição seletiva de acesso aos níveis inferiores. Isto é, aquele mesmo desenhista se encontra limitado às ferramentas e funcionalidades disponíveis naquele aplicativo específico - ele não tem acesso ao código que define os seus mecanismos.[135]

Não existe uma definição única ou universal para os diferentes níveis de abstração em meio digital, e tampouco limites ou fronteiras claras entre o que pode ser considerado um nível baixo ou alto de abstração, bem como quaisquer níveis intermediários. Entretanto, é relativamente claro estabelecer uma comparação quanto ao nível de abstração entre dois ou mais objetos de análise. Por exemplo, uma linguagem como C++ claramente atua em um nível abaixo da Scratch (baseada em uma interface visual), e acima da linguagem de máquina.

[135] Exceto, é claro se o artista for capaz de programar a funcionalidade desejada ele mesmo. Outra opção seria utilizar uma linguagem de scripting específica, como as que existem em alguns dos aplicativos no pacote da Adobe (como Illustrator, Photoshop e assim por diante), ou a linguagem MAXScript do 3D Studio Max, por exemplo.

Executando Recursos Procedurais

Recursos modais procedurais podem ser representados internamente de diferentes maneiras, de acordo com a sua implementação em uma dada linguagem de programação ou aplicativo de autoria. Também existem diferentes maneiras pelas quais estes recursos pode ser acessados e manipulados pelo usuário. Assim, no caso de uma interface visual, isso acontece a partir de elementos gráficos, mas para um programador eles só existem na forma de abstrações que podem ser empregadas no código.

De qualquer maneira, a princípio estas implementações particulares destes recursos, no nível mais baixo de abstração, são independentes do seu efeito ou função prática no contexto mais amplo do sistema. O significado representado ou expressado por um artista ou criativo através de um determinado recurso procedural não reside na sua implementação individual, mas sim na sua função dentro da lógica do sistema, em um nível de abstração mais alto.

Isto se torna aparente se considerarmos que um mesmo recurso procedural (como por exemplo uma regra ou comportamento) pode ser implementado de diferentes maneiras, utilizando uma variedade de linguagens de programação, abordagens ou técnicas.

Um determinado recurso procedural só pode produzir significado no contexto da execução do seu código. Isso pode acontecer diretamente, a partir da materialização de algum tipo de produto ou saída (ou output) que pode ser percebido pelo usuário ou público (como veremos mais adiante), ou indiretamente, influenciando outras funções ou recursos contidos no sistema (considerando que estes, por sua vez, tenham algum tipo de produto que possa ser percebido externamente por um receptor, em tempo real ou não).

Esta particularidade reflete uma propriedade que distingue os recursos procedurais daqueles de natureza estática, que é a sua integração com a lógica do sistema. Um determinado recurso modal só pode ser considerado procedural se ele próprio for procedural, ou se ele afetar a lógica do sistema (isto é, se afetar ao menos algum outro recurso procedural) de uma maneira significativa.

Além disso, note que, enquanto que os recursos estáticos podem se materializar diretamente para o receptor, os recursos procedurais só podem se manifestar indiretamente, por intermédio dos elementos estáticos.

Isso pode ser exemplificado a partir do emprego de arquivos de som no MissionMaker. No editor, é possível importar conteúdo de áudio para utilizar dentro de um projeto, como música, efeitos sonoros para determinados eventos, falas para personagens e assim por diante.

Do ponto de vista desta análise, este tipo de recurso é considerado como estático, pelo fato de que este conteúdo não tem influência sobre a lógica do sistema. Isto é, qualquer áudio de um projeto poderia ser trocado por outro, e a execução do programa seria idêntica em termos da sua lógica ou comportamento.

Isso não significa que estes recursos estáticos não tem um potencial expressivo. No caso da música, por exemplo, se uma composição suave for trocada por uma mais agressiva, isso pode alterar completamente a experiência, mesmo que não afete a lógica do sistema em si. Na maioria dos jogos criados na oficina, elementos importantes da experiência eram apresentadas na forma de falas e efeitos sonoros, como falas de personagem e vocalizações expressando dor ou raiva.

Entretanto, independentemente do conteúdo escolhido pelo designer, o recurso sonoro no MissionMaker é sempre de natureza estática. Como tal, só pode ser utilizado para realizar formas expressivas tradicionais (como Música e Drama, por exemplo).

Suponha, entretanto, que o MissionMaker oferecesse uma função que permitisse analisar o áudio sendo utilizado em um projeto. Digamos que seja uma função simples, capaz de detectar a amplitude (ou volume) de um arquivo de som em relação aos demais. Esta nova função permitiria, por exemplo, que um designer criasse uma regra que faria um personagem reagir caso o volume de som em um ambiente fosse mais alto que um certo valor. Por exemplo, pode haver uma situação na qual o jogador precisa pegar um objeto que se encontra do lado de um personagem que está dormindo, sem fazer muito barulho para não acordá-lo (uma possível solução seria remover os sapatos, e caminhar lentamente de forma a fazer o mínimo de barulho).

Note que, nesta situação hipotética, na qual arquivos de áudio passariam a ser considerados como recursos procedurais, participando da lógica do sistema, a natureza do recurso em si (o som) não muda - este conteúdo continua sendo inerentemente estático. O que muda neste caso é o contexto no qual este recurso está inserido - um sistema capaz de analisar um determinado aspecto do som. Note também que, embora o arquivo de som esteja contribuindo para a construção de significado procedural, isto ainda acontece em um grau menor em comparação com recursos procedurais tais como regras e comportamentos (como, por exemplo, a própria função capaz de analisá-lo).[136]

[136] A ideia de mensurar graus de proceduralidade se relaciona ao conceito de intensidade procedural, mencionado anteriormente (p. 56).

Isto é, o que determinada se um dado recurso terá uma função estática ou procedural depende do design do sistema e da maneira como ele foi implementado. Assim como a função hipotética apresentada nos parágrafos anteriores, uma infinidade de outras poderiam ser implementadas, envolvendo qualquer tipo de conteúdo, como as texturas dos objetos no jogo, o texto nos diálogos dos personagens e assim por diante.

Convém lembrar que, se por um lado o significado procedural só pode ser realizado a partir da execução do código no qual ele é representado, por outro, a interatividade com o receptor nem sempre é necessária para que isso aconteça.

É verdade que, no caso particular dos jogos, a interação é necessária para a realização dos significados representados especificamente nos elementos de jogabilidade (ou gameplay). Por exemplo, o clássico jogo de computador The Oregon Trail (MECC, 1971) representa na sua lógica uma série de aspectos da vida dos colonizadores Americanos no século XIX (fig. 34, abaixo). Porém é necessário interagir com o jogo de forma a realizar e perceber este significado.

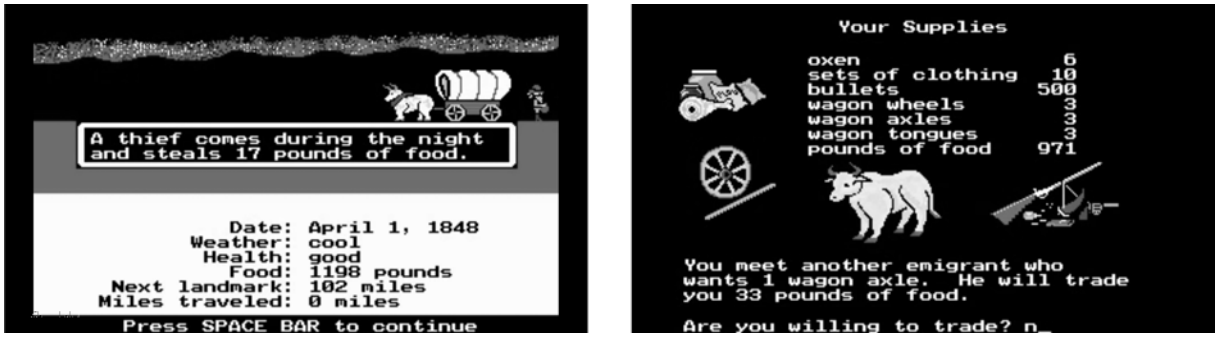


Figura 34 – The Oregon Trail (MECC, 1971)

Entretanto, há jogos que apresentam elementos procedurais expressivos, mas que não fazem parte da sua jogabilidade ou mecânica fundamental. Por exemplo, jogos baseados em simulações ou sistemas tendem a revelar os seus mecanismos, aspectos e características mesmo quando não há interação do jogador. Grande parte das regras e comportamentos que determinam como os personagens e outros elementos da simulação irão reagir ao jogador também se aplicam a estes próprios elementos do sistema. Embora o jogador seja a figura central na experiência, sob esta perspectiva, ele nada mais é do que mais um personagem entre todos os outros.

Em jogos como The Witcher 3: Wild Hunt (CD Projekt RED, 2015), uma aventura de fantasia (fig. 35, abaixo), ou Grand Theft Auto 5 (Rockstar; fig. 36), o universo do jogo está sempre "vivo", e muitas vezes os seus habitantes tem vidas próprias, realizando tarefas, interagindo entre si e assim por diante. A mera observação do comportamento destas simulações já revela grande parte do tipo de experiência que foi idealizada e projetada pelos seus designers e criadores.

Figura 35 – Witcher 3 (CD Projekt RED, 2015)

Figura 36 – Grand Theft Auto 5 (Rockstar)

Por exemplo, fora de situações de ação ou conflito, o mundo de Witcher 3 tende a ser plácido e pacífico, enquanto que as ruas de GTA frequentemente são palco de brigas espontâneas e acidentes espetaculares. Esta diferença pode ser percebida por alguém apenas observando o sistema destes dois jogos, sem interagir de maneira significativamente, de maneira análoga ao público em uma performance de teatro de improviso, por exemplo.[137]

Alguns artistas e criativos exploram este potencial expressivo sistêmico através de produções não interativas (ou com interatividade diminuída), como no caso de Mountain (David OReilly, 2014), uma obra experimental na qual o "jogador" limita-se a observar (ou "ser") uma montanha que flutua no espaço. Enquanto que alguns criticam Mountain, sugerindo que trata-se de um simples "protetor de tela", outros elogiam a natureza introspectiva e poética do sistema.

Outro exemplo do emprego expressivo de estratégias procedurais sem a inclusão de elementos interativos é a arte de instalação Balance From Within (Jacob Tonski, 2012), cuja composição envolve um sofá antigo equilibrado em apenas um dos pés. Na obra, exposta no festival FILE 2013, em São Paulo, a única interação que acontece é entre o sistema computacional e as forças da gravidade. O artista descreve a obra como uma "metáfora cinética para o risco inerente nas relações sociais" (TONSKI, 2014).[138]

[137] Fonte da imagem para Witcher 3: Gamers Pack: The Witcher 3: Wild Hunt gameplay video and screens show a world worth exploring (2015/01/27). URL: `www.gamerspack.com/2015/01/27/the-witcher-3-wild-hunt-gameplay-video-and-screens-show-a-world-worth-exploring/`

[138] Citação: tradução nossa - texto original: "a kinetic metaphor for the inherent risk in social relations" (sem numeração de página)

Da mesma forma, em grande parte da Arte Generativa, na qual o produto final é um texto, imagem ou som de natureza estática (em suporte digital ou físico), não há interatividade com o receptor.[139]

Para contextualizar de maneira mais completa os recursos procedurais que serão apresentados nesta seção, também é útil compreender a relação deles com o código que os representa, do ponto de vista prático, a partir do momento da sua execução.

De uma maneira simplificada, um algoritmo ou programa geralmente é organizado em duas partes (ou seções) principais: o setup (ou montagem) e o loop (ciclo ou repetição) - ambas tratam-se de código sendo representado na memória do computador, porém elas diferem em termos do seu propósito no contexto da sua execução. O setup é executado apenas uma vez, antes do loop, e geralmente contém a porção do código relacionada à preparação ou montagem do ambiente para a execução do restante do programa. O código dentro do loop, por sua vez, repete-se indefinidamente.[140]

Na prática, todas as funções, atribuições e processos contidos no loop estão sempre sendo executados e atualizados - isto é, estão "vivos". Do ponto de vista dos recursos procedurais, isso significa que mesmo regras ou funções que não envolvem repetições ou ciclos internos (como uma função que verifica uma colisão entre dois objetos na simulação, por exemplo) serão executadas indefinidamente de acordo com a maneira como forem implementadas na estrutura do algoritmo.

Mídias Digitais: Múltiplas Contribuições

Vimos que as práticas criativas em meio digital envolvem e podem ser afetadas por uma série de diferentes camadas e contribuições (p. 79). Além do artista ou criativo responsável pela produção digital, há inúmeros outros profissionais envolvidos, desde os programadores dos aplicativos e ferramentas utilizadas, bem como do sistema operacional sobre o qual estes programas foram implementados, até os criadores do hardware no qual eles residem. Se por um lado a identidade de uma determinada produção é geralmente definida pelas ideias e gestos do seu criador, existe uma série de outros elementos influenciando e determinando inúmeros aspectos da sua composição. Isso torna a determinação da autoria deste tipo de produção uma questão particularmente complexa (p. 84).

[139] De qualquer maneira, mesmo nos casos em que a interação é necessária para a realização de um determinado significado procedural, no contexto desta pesquisa, o nosso foco não é na interação em si, mas sim nos sistemas e algoritmos que a viabilizam.

[140] Esta é uma simplificação sobre a estrutura e execução de um programa de computador, apresentada aqui com propósitos didáticos. As duas partes (ou seções) mencionadas aqui nem sempre estão presentes em um programa, e diferentes linguagens e ambientes de programação e autoria procedural podem utilizar estruturas e nomenclaturas diferentes.

Isso também acontece nas práticas tradicionais, nas suas ferramentas e processos, que também podem envolver uma variedade de diferentes camadas e contribuições. O pincel utilizado por um pintor tem uma composição particular que influencia o seu trabalho, por exemplo. Entretanto, nas mídias digitais os sistemas são potencialmente muito mais complexos, principalmente por conta do aspecto procedural.

Esta complexidade pode causar um distanciamento significativo entre o gesto de quem cria e os seus resultados ou efeitos na produção final. Por exemplo, enquanto que o pincel pode ser percebido pelo pintor como uma entidade física, cuja interação com a tinta e a tela é relativamente direta e evidente, a programação de computadores trata-se de uma atividade predominantemente abstrata, a princípio desprovida de uma conexão clara com a fisicalidade do mundo e dos gestos realizados por quem cria (principalmente em comparação com as formas e linguagens tradicionais).

No MissionMaker, embora o aplicativo ofereça uma autoria relativamente granularizada, e diversas funcionalidades pelas quais os designers podem criar e editar os seus jogos, estas se limitam predominantemente a elementos e propriedades estáticas. Como veremos mais adiante, a autoria procedural neste aplicativo se limita principalmente à atribuição de variáveis e a criação de regras - ou declarações condicionais -, bem como na escolha de composição entre estes recursos. Praticamente todos os demais recursos procedurais se encontram predefinidos e fixados no engine do aplicativo.

Isso significa que a maior parte dos recursos procedurais que compõem uma produção realizada no MissionMaker são de autoria dos criadores da ferramenta, e não dos designers dos jogos propriamente ditos. Estes recursos incluem desde funções gráficas, de entrada e saída, até muitas das regras e comportamentos que compõem a lógica dos jogos. Por outro lado, esta predominância é principalmente quantitativa, e não implica necessariamente em uma diminuição no potencial expressivo desta ferramenta.

Dito isso, nas análises que realizamos sobre os jogos produzidos no contexto da oficina Playing Beowulf, constatamos que a autora exercida pelos participantes foi predominantemente a partir de recursos estáticos, tais como textos, imagens, sons e música. A autoria (e a expressividade) procedural foi relativamente menos significativa, embora esteve presente de alguma maneira na maior parte das produções.

Escolha da Perspectiva de Análise

Os recursos procedurais identificados neste estudo foram selecionados por serem os mais relevantes especificamente para o designer utilizando o MissionMaker. Desta forma, não se trata de uma análise definitiva deste tipo de recurso em todas as ferramentas similares, e tampouco nos meios digitais de uma maneira geral.

Muitos dos recursos procedurais no MissionMaker não são acessíveis diretamente ao usuário, sendo unicamente parte do seu engine. Mais adiante apresentaremos alguns destes recursos, e veremos como eles podem afetar e contribuir para a execução e exibição dos jogos desenvolvidos nesta ferramenta.

Os principais recursos no MissionMaker que estão à disposição do usuário ou designer são as **variáveis** e as **regras** (ou declarações condicionais). Como veremos, uma variável, além de poder ser um recurso modal em si, também pode fazer parte de uma declaração condicional.

Este tipo de hierarquia aninhada é análoga a aquela encontrada entre os diferentes Modos comunicacionais, de acordo com a Análise Multimodal. Por exemplo, o ato de caminhar pode ser considerado como um recurso modal, por conta de tratar-se de um gesto potencialmente comunicativo e expressivo (LEEUWEN, 2005). O fato de que uma pessoa pode caminhar de diferentes formas significa que existe um potencial para a construção de significado neste gesto. Um estilo de caminhada energético e amplo pode exprimir jovialidade e alegria, enquanto que um estilo lento, preciso e metódico pode expressar frieza e controle, como na escolha do ator Anthony Hopkins para a construção do seu icônico personagem Hannibal Lecter, o assassino em série canibal no filme Silêncio dos Inocentes (Jonathan Demme, 1991).[141]

O ato de caminhar, por sua vez, é formado por uma série de recursos modais mais específicos, tais como "movimento de perna", "ritmo" e "direcionamento do olhar". Estes recursos, além de fazerem parte do conjunto mais amplo da "caminhada", também podem ser considerados individualmente, e mesmo como parte de outros recursos modais. Por exemplo, uma pessoa que está sentada pode utilizar apenas o "movimento de perna" para comunicar ou expressar uma série de ideias ou significados - cruzá-las para expressar formalidade, abri-las para sugerir intimidade, balançar elas para revelar nervosismo e assim por diante. Parte do potencial de significação disponível neste subconjunto do "movimento de perna" pode interseccionar com o recurso modal mais amplo do "caminhar", e outra parte pode ser exclusiva a ele.

[141] Hopkins depois interpretaria este mesmo personagem em outros dois filmes, mantendo - e de certa forma enfatizando - esta mesma escolha de interpretação. Título do filme original: The Silence of the Lambs.

Cada uma destas escolhas de perspectiva - ou recorte - permitem abordar o mesmo objeto de análise (neste caso, o ato de caminhar) a partir de diferentes contextos e níveis de detalhamento (ou de hierarquia), e para propósitos específicos.

De maneira similar, os recursos modais disponíveis no MissionMaker, apresentados a seguir, também existem em diferentes níveis de abstração, dentro do contexto mais amplo do aplicativo. Nos níveis mais baixos estariam elementos como as variáveis e estruturas de dados, nos níveis acima residiriam as funções e declarações condicionais, e as abstrações de nível mais alto seriam os modelos conceituais, por exemplo (esta categorização será examinada em maior profundidade mais adiante, ao tratarmos sobre os recursos presentes no engine do MissionMaker).[142]

No MissionMaker a autoria procedural pode acontecer em certos elementos ou níveis de abstração, mas não em outros. Por exemplo, o aplicativo não permite ao usuário criar ou modificar os movimentos dos personagens. Isso se aplica tanto a aqueles movimentos diretamente relacionados aos comandos (ou interação) do jogador - como a animação da caminhada ou de luta - quanto a aqueles associados a eventos automatizados do comportamento dos personagens e outros elementos da simulação.

Este é o caso, por exemplo, do comportamento predefinido que faz um personagem seguir o outro. Trata-se de um um recurso que não pode ser acessado diretamente pelo designer, porém que pode ser empregado como parte de um sistema mais amplo. Como vimos na análise do jogo criado por Tracey na oficina, ela criou uma regra que determinava quando este comportamento seria ativado, com isso expressando a ideia da lealdade (p. 141). Isto é, embora ela não tivesse acesso para atuar criativamente no nível de abstração do comportamento em si, ela tinha algum controle no nível acima, ao definir as condições nas quais ele iria acontecer, através da definição da regra.

[142] Como o foco deste estudo é em um aplicativo específico - isto é, no MissionMaker -, não trataremos de todos os possíveis níveis de abstração (por exemplo, os níveis mais baixos da linguagem de máquina, ou da lógica do hardware). Além disso, nossa intenção não é de esgotar todos os recursos disponíveis em cada nível de abstração.

MissionMaker: Variáveis e Condicionais

A seguir, analisaremos como os recursos modais procedurais se encontram implementados no MissionMaker. Isso será feito a partir da perspectiva do designer utilizando as funções e ferramentas deste aplicativo para a criação de uma narrativa ou jogo digital

Variáveis

A variável é um dos elementos mais fundamentais da lógica computacional, presente de alguma forma em praticamente todas as linguagens de programação, e consiste em um recurso importante no contexto do MissionMaker. Trata-se de uma abstração que pode conter, ser associada ou apontar para diferentes tipos de dados e conteúdo, incluindo outras variáveis e expressões, ou funções. A definição de uma variável pode ser desde uma simples atribuição (como: $x = 1$) até uma equação ou estrutura mais complexa e sofisticada. Assim, uma variável pode conter tanto conteúdo estático quanto procedural - em ambos os casos, como veremos, a variável tem o potencial de atuar como um recurso procedural.

Geralmente a variável é nomeada de maneira intuitiva pelo programador, fazendo referência direta à sua função ou o que ela representa - por exemplo, uma variável contendo o tamanho de um objeto poderia ser nomeada "tamanhoObjeto". Entretanto, a princípio o nome de uma variável não afeta a lógica do programa em si, e desta forma não contribui na construção de significado da perspectiva do produto final. O potencial expressivo ou de significação de uma variável reside na sua influência e implicações dentro do sistema, e nas intenções do designer na sua implementação.

Variáveis podem ser consideradas como recursos procedurais mesmo quando elas contém ou apontam para dados ou conteúdo estático (como texto, números, imagens, sons e assim por diante), desde que estes afetem a lógica do sistema.

Por exemplo, vimos que na biblioteca de objetos do MissionMaker existe uma garrafa de vinho, que pode ser coletada, manipulada e utilizada pelo jogador. Este objeto também tem determinadas propriedades e variáveis que podem ser associadas a ele. Uma destas variáveis permite definir se o uso deste objeto por um personagem afeta a sua saúde, e em que medida. Embora esta seja uma informação estática (trata-se de um valor numérico), ela pode afetar o comportamento da simulação - por exemplo, o designer pode definir que se o personagem consumir o vinho a sua saúde melhora.

No contexto do MissionMaker, a atribuição de variáveis como um recurso procedural tem um potencial expressivo e de significação relativamente limitado. Há dois motivos principais para isso: primeiro, porque o usuário não pode criar novas variáveis, pelo menos não diretamente (é possível criar novas entidades, que por sua vez podem conter variáveis associadas a eles - mas não é possível adicionar ou remover variáveis destas entidades); segundo, porque o designer tem acesso a um conjunto limitado de variáveis, seja para edição, ou para serem utilizadas como parâmetros na definição de regras e comportamentos.

A limitação no acesso do usuário às variáveis no MissionMaker também tem uma importante implicação do ponto de vista da autoria. Além do conjunto de recursos procedurais existentes ser fixo e predeterminado, cada um destes recursos representa um potencial de significado que não pode ser modificado pelo usuário. Por exemplo, considere o fato de que a maior parte das variáveis existentes no MissionMaker referem-se aos aspectos físicos e materiais da simulação - a variável "resistência" ("resistance") define como a saúde do personagem é afetada por um ataque, "escala" ("scale") define o tamanho de um objeto dentro do espaço do jogo, e assim por diante. O designer pode escolher valores diferentes para estas variáveis, desta forma exercendo um certo grau de autoria, mas ainda assim de maneira limitada.

Esta limitação é ainda mais significativa no caso de variáveis de jogo que não são modificáveis pelo usuário, ou aquelas que não podem ser referenciadas pelas regras e comportamentos - é o caso dos recursos presentes no engine do aplicativo, como veremos mais adiante (p. 170).[143]

A maior parte destas limitações decorrem do fato de que, como vimos, o MissionMaker é um aplicativo intencionalmente mais simplificado, de forma a tornar o seu uso mais acessível e didático. De qualquer forma, estas limitações não são necessariamente um obstáculo, principalmente considerando que muitas vezes a criatividade prospera justamente a partir da diminuição da liberdade. Por exemplo, em uma das análises de jogo apresentadas anteriormente, no projeto de Andrew Smith (p. 107), vimos que ele incluiu a garrafa de vinho, porém definiu a variável de saúde ("health") com um valor negativo - sendo que normalmente uma propriedade associada a esta função tem um valor positivo. O resultado desta implementação é que, ao invés do jogador ter um aumento na saúde ao consumir o vinho, o oposto acontece - isto é, o vinho prejudica o personagem. Segundo Andrew, a sua intenção ao implementar esta mecânica era representar a ideia de que Beowulf, o protagonista do jogo e herói da aventura, não deveria perder tempo enebriando-se.[144]

[143] Mesmo no caso das variáveis que podem ser modificadas pelo usuário, os seus valores padrão (ou "default") refletem uma perspectiva particular que os criadores do aplicativo tiveram em relação a cada um destes recursos.

[144] Andrew também reforçou esta ideia com um elemento adicional de natureza estática, na forma de uma mensagem de texto repreendendo o jogador se o vinho for consumido.

Declarações Condicionais (ou Regras)

As **declarações condicionais** são funções que, juntamente às variáveis e expressões, controlam o fluxo de execução de um programa, de acordo com determinadas condições. Este é um dos principais elementos pelos quais a proceduralidade pode ser explorada, uma vez que, ao contrário da variável, a declaração condicional é um recurso inerentemente procedural. No MissionMaker este recurso existe na forma das "regras" (rules), que são os principais elementos pelos quais o usuário pode realizar autoria procedural neste aplicativo.

Em comparação com as variáveis, uma vantagem prática e direta no uso de declarações condicionais é que elas permitem e facilitam ao designer criar e organizar desde regras simples até sistemas complexos. Entretanto, um dos aspectos mais importantes deste recurso do ponto de vista do processo criativo é o fato de que ele fornece uma camada adicional de abstração no fluxo lógico do algoritmo, que pode ser explorada pelo designer de maneira expressiva, ou para a construção de significado.

O usuário tem um grau relativamente amplo de liberdade para criar e modificar regras e comportamentos, que podem ser relacionadas ou associadas a personagens, objetos e outros elementos da simulação e da lógica do sistema. Entretanto, existem limitações sobre o quanto um designer é capaz de expressar a partir do emprego deste recurso no MissionMaker.

Por exemplo, embora existam múltiplas opções disponíveis para ativadores ("triggers"), apenas um deles pode ser utilizado por vez em cada regra. Por conta disso, não é possível criar regras mais sofisticadas, envolvendo múltiplas condições . Além disso, cada tipo de elemento, seja ele um objeto, personagem ou localidade, tem um conjunto limitado de ações que podem ser ativadas por uma regra.

O potencial de significação que pode ser representado a partir das condicionais é determinado (ou limitado) pelo tipo de controle que o aplicativo fornece ao designer sobre a sua criação e modificação. Isso pode depender de quais elementos estão envolvidos em um determinado recurso, ou do nível (ou camada) na qual eles se localizam.

Isto é, por um lado, existem aspectos que são fixados (ou "hardcoded") no aplicativo, tais como o número específico de ativadores por regra, mencionados anteriormente, que limitam as possibilidades criativas do designer. Por outro lado, existem alguns elementos que são mais flexíveis, como a escolha do "alvo" (ou "target") dentro da definição de uma regra, e que permitem ao designer uma liberdade criativa e expressiva maior.

Além disso, o potencial expressivo das condicionais também depende de outros fatores, comuns a qualquer outro meio ou linguagem, incluindo a subjetividade, intenção e repertório de quem cria.

Assim, enquanto que um designer em uma determinada situação pode sentir falta de alguma funcionalidade específica no MissionMaker, e portanto considerar a ferramenta incompleta, outro criativo em um contexto diferente pode considerar esta mesma funcionalidade irrelevante, e não ter o seu processo criativo afetado por este aspecto.

Como mencionamos anteriormente, na maior parte dos jogos analisados neste estudo as regras foram empregadas pelos criadores de uma maneira superficial, para a realização de tarefas simples e práticas.

O tipo de regra mais comum que encontramos nos jogos produzidos na oficina é aquele cuja função é a ativação de um determinado elemento (como um som, música ou uma "mensagem pop-up") em decorrência de algum evento no jogo. Neste caso, apesar de tratar-se de uma regra - um recurso procedural por excelência -, esta não afeta o sistema de maneira alguma, pois sua única função é de produzir uma saída que pode ser percebida pelo receptor.

Outro uso comum das regras que pudemos identificar nos jogos, e que explora esta funcionalidade de maneira um pouco mais elaborada, é na implementação de mecanismos para abrir e fechar portas ou passagens, em decorrência de alguma outra ação (pegar um objeto, mover uma alavanca e assim por diante). Neste caso, a regra tem uma certa influência na simulação, embora de maneira bastante simples.

O Engine: Significados Ocultos

Os dois recursos procedurais apresentados anteriormente - as variáveis e as regras (ou declarações condicionais) - são os principais elementos identificados neste estudo pelos quais o designer pode exercer autoria procedural a partir do editor do MissionMaker. Entretanto, há uma quantidade (e variedade) muito maior de recursos presentes no seu engine. Estes recursos, embora não sejam acessíveis ao designer, também fazem parte dos jogos produzidos no MissionMaker, ou os influenciam (direta ou indiretamente).

Esta seção apresenta exemplos de alguns destes recursos, bem como outros elementos que influenciam o engine, vindos de níveis de abstração mais baixos. Analisaremos como cada um destes recursos pode afetar ou contribuir na execução ou apresentação dos jogos implementados através desta ferramenta, tanto de um ponto de vista mais prático, quanto em termos do seu potencial expressivo e para construção de significado.[145]

Vimos anteriormente que os níveis de abstração no meio digital representam as camadas que se encontram acima da lógica mais fundamental da máquina (p. 156). No caso dos recursos procedurais disponíveis para o designer no MissionMaker, apresentados anteriormente, as variáveis se encontram em um nível de abstração mais baixo, comparado às declarações condicionais (ou regras).

No contexto dos exemplos apresentados a seguir, abstrações de **alto nível** se referem às funções que são específicas ao engine do MissionMaker, o **nível médio** se refere à linguagem de programação empregada na implementação do aplicativo em si e **baixo nível** se relaciona à lógica computacional do meio digital de uma maneira mais fundamental (fig. 37; p. 171). A análise que faremos sobre o nível mais alto de abstração é mais aprofundada e extensa em comparação com os níveis mais baixos, uma vez que aquela se refere mais diretamente ao aplicativo sendo analisado, enquanto que os níveis mais baixos de abstração tendem a ser mais genéricos em termos de função.

Esta divisão entre os diferentes níveis de abstração não deve ser considerada como absoluta, e tampouco rígida. Assim, por exemplo, por um lado existem determinados elementos e aspectos que são compartilhados por qualquer produção ou entidade representada em meio digital, referentes às características e funcionalidades mais fundamentais da lógica computacional. Por outro lado, diferentes produções, aplicativos e linguagens podem empregar diferentes abordagens para a implementação de cada um destes sistemas e estruturas. Além disso, a classificação apresentada aqui se limita a

[145] Como vimos, embora nesta pesquisa fazemos referências a determinados aspectos e elementos técnicos do meio digital, estes são tratados sempre a partir da perspectiva do artista ou criativo exercendo autoria procedural. Tratamos sobre o conceito do engine anteriormente na tese (p. 67).

três categorias - nível alto, médio e baixo de abstração - principalmente por motivos didáticos. Na prática, é possível considerar subdivisões adicionais dentro de cada uma destas categorias.

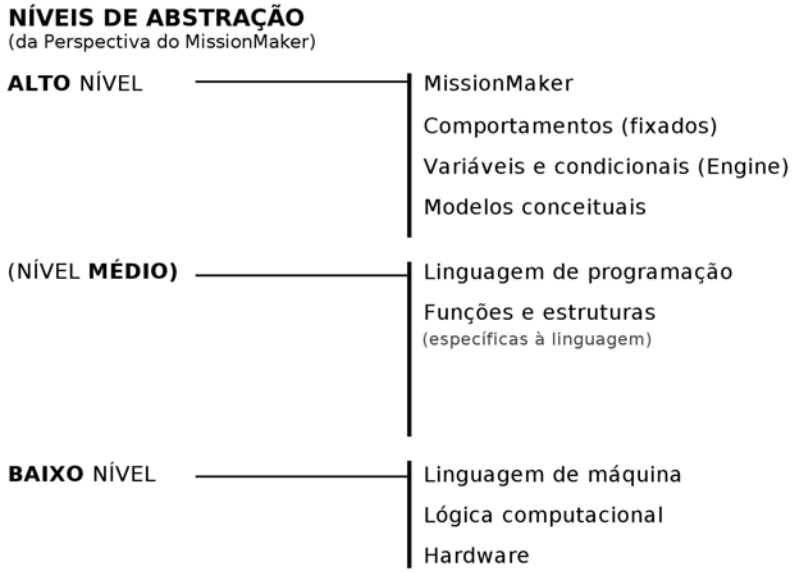


Figura 37 – Diagrama: níveis de abstração (MissionMaker)

Abstrações de Alto Nível

As abstrações de alto nível mais evidentes, presentes no engine do MissionMaker, são as regras e variáveis análogas a aquelas que podem ser editadas (e eventualmente criadas) pelo usuário no aplicativo, e que ocupam um nível similar de abstração (e de funcionalidade) que elas.

Um exemplo da presença de regras no engine são as abstrações que representam determinados comportamentos dos personagens. Por exemplo, considere o comportamento padrão para quando um personagem sofre um ataque ou dano (como um tiro, soco ou os efeitos da explosão de uma bomba). Há uma animação específica correspondente a este evento - os braços se levantam ligeiramente, como se o personagem estivesse tentando se proteger, o torso curva-se um pouco para frente e as pernas se dobram.

Como este comportamento se encontra predefinido e fixado no engine do MissionMaker, isso significa que o designer não pode modificar a maneira como os personagens reagem a uma agressão física. Por exemplo, não é possível alterar a amplitude dos movimentos, remover ou incluir novos gestos e eventos.

De maneira análoga, também existem inúmeras variáveis internas ao engine, relativas a propriedades e informações que se encontram fixadas na simulação. Por exemplo, a varável que representa a velocidade na qual os objetos podem ser arremessados pelo jogador na simulação.

Entretanto, para os exemplos a seguir vamos nos concentrar em um tipo de abstração em um nível ainda mais alto, que são os **modelos conceituais.** Todo aplicativo ou produção digital emprega ou envolve uma série de modelos conceituais, que são um tipo de abstração de alto nível utilizada para descrever a estrutura e a lógica de cada um dos elementos e comportamentos presentes no sistema. Por exemplo, em um engine de jogo como MissionMaker, os modelos conceituais servem para definir desde a implementação da representação interna de elementos físicos (como a topologia 3D dos personagens, objetos e localidades) e a estrutura do sistema de regras, até a forma como o espaço virtual será apresentado ao receptor (a perspectiva de visão do jogador, a maneira como os elementos serão apresentados visualmente e assim por diante).[146]

Considerados a partir da sua qualidade como recursos procedurais, os modelos conceituais contidos (ou empregados) no engine do MissionMaker são construções fundamentalmente arbitrárias, e que determinam ou influenciam de maneira significativa a maior parte dos aspectos dos diversos elementos e comportamentos que compõem os jogos produzidos com este aplicativo.

Por exemplo, considere o modelo conceitual de **representação,** utilizado para descrever o sistema ou universo do jogo internamente em um projeto no MissionMaker. Como vimos, neste aplicativo o engine é predominantemente baseado em uma metáfora para o mundo físico, em três dimensões (3D). Este é um modelo comum entre os jogos digitais modernos, tais como os FPS (jogos de tiro), os jogos de plataforma 3D e mesmo produções artísticas experimentais. O modelo de representação empregado no MissionMaker para descrever internamente os ambientes, personagens e outros elementos do jogo consiste em uma série de diretrizes, como por exemplo a definição de que toda entidade física presente na simulação possui uma topologia particular, uma posição e rotação no espaço tridimensional, pode estar submetida a uma metáfora que simula o efeito das leis da física sobre elas, e assim por diante.

De forma a ilustrar o caráter fundamentalmente arbitrário de um modelo conceitual, comparemos este modelo de representação do MissionMaker com o de outras produções digitais. Por exemplo,

[146] A definição de "modelo conceitual" na computação não é rígida, e pode variar entre diferentes paradigmas de programação, programadores ou autores. De uma maneira geral, refere-se às descrições de alto nível que determinam as estruturas e lógica de um dado sistema.

considere o modelo de representação empregado no clássico Pac-Man (Namco, 1980). Embora este jogo também seja baseado em uma metáfora do espaço físico, isso é feito de uma maneira menos realista do que no MissionMaker, abstraída para uma representação bidimensional. Considere agora um aplicativo de planilhas, como o OpenOffice Calc, cujo modelo interno de representação é composto de entidades como números e tabelas. Neste caso, não há referência ou relação alguma entre o conteúdo deste aplicativo e o mundo físico - exceto talvez pela camada de interface de usuário, pela qual esta informação pode ser acessada (fig. 38, abaixo).[147]

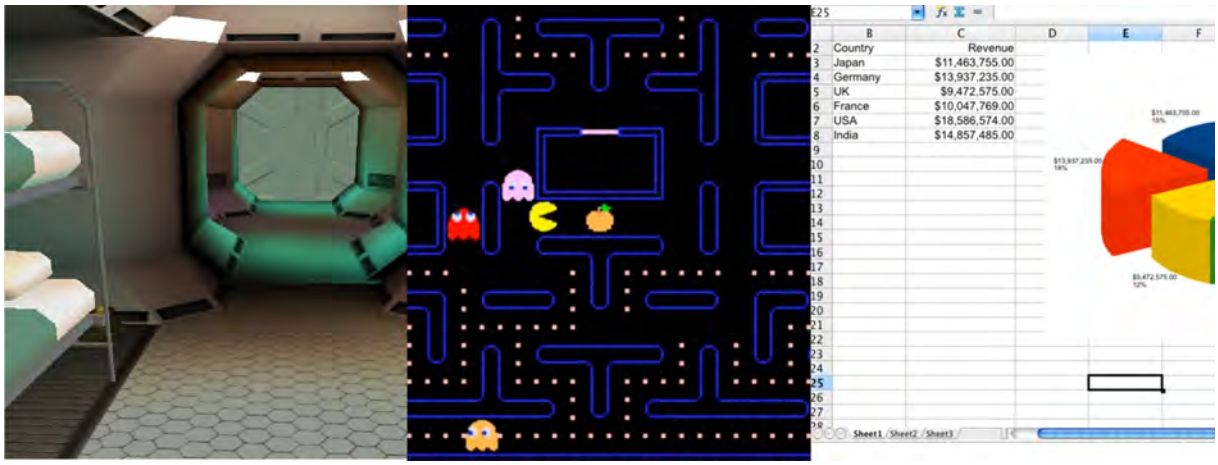


Figura 38 – Modelos de representação: MissionMaker (esq.), Pac-Man e Calc.

O conteúdo em si, que é informado por estes modelos conceituais, também pode ser considerado como um recurso modal, mesmo se este for de natureza estática. Como vimos anteriormente ao tratar sobre as variáveis, qualquer informação estática pode ser contextualizada pelo designer de maneira a afetar ou contribuir na lógica do sistema - na ocasião, isto foi ilustrado com o exemplo da garrafa de vinho, cuja propriedade numérica "saúde" pode participar da mecânica do jogo.

No contexto do modelo de representação do MissionMaker, por exemplo, a topologia do espaço físico do jogo trata-se de um conteúdo estático, que consiste em uma descrição tridimensional, incluindo o posicionamento de vértices, faces e assim por diante. Porém, este conteúdo estático pode afetar diretamente o comportamento dos personagens e a maneira como eles se relacionam entre si.

[147] Embora os modelos conceituais de representação mencionados neste parágrafo estejam sendo ilustrados nesta página a partir de imagens, neste caso a apresentação visual não é relevante, uma vez que o foco se encontra nas estruturas internas destes sistemas. Como veremos adiante, a apresentação visual trata-se de outro modelo conceitual, que pode ou não refletir o modelo de representação interno de um determinado sistema. Fonte das imagens: Pac-Man (youtube/Old Classic Retro Gaming) URL: `youtu.be/dScq4P5gn4A`; MissionMaker: Learning with Games - URL: `learningwithgames.pbworks.com/f/world_war.jpg`; OpenOffice Calc: Site oficial: OpenOffice.org 3.0 New Features - URL: `www.openoffice.org/dev_docs/features/3.0/`

Contraponha isto com a topologia dos personagens, por exemplo, que trata-se de um conteúdo que não afeta o sistema. Embora cada personagem tenha uma aparência distinta, da perspectiva da simulação todos são representados internamente como volumes simples (de forma a simplificar determinadas operações, como detectar colisões entre os objetos, por exemplo).

Outro tipo de modelo conceitual é aquele referente à **apresentação** de conteúdo ao receptor - seja ele visual, sonoro, cinético e assim por diante. Considerados como recursos procedurais, estas funções também podem ser exploradas por um designer para expressar uma determinada estética, ou mesmo para transmitir ideias e significados particulares.

Este é o caso das funções que servem para renderizar um ambiente virtual 3D, mapeando ele para uma representação 2D na tela, como no caso do MissionMaker. Neste aplicativo, da mesma forma que o modelo de representação interna é baseado no mundo físico, o modelo de apresentação visual também busca emular a maneira como enxergamos na realidade (embora evidentemente de maneira mais simplificada). A simulação do espaço físico do jogo é apresentada visualmente de forma a emular uma perspectiva em primeira pessoa.

Isto é feito a partir de uma simulação das propriedades físicas da luz, e como ela se comporta em relação à topologia, as texturas e outras propriedades do ambiente virtual, e da sua percepção no olho humano, em uma aproximação do que uma pessoa veria em um ambiente similar na realidade. Entretanto, este é um modelo arbitrário - trata-se de apenas uma entre as inúmeras maneiras de representar visualmente um espaço tridimensional.

Figura 39 – Modelos de apresentação: Psychonauts (esq.), StarCraft e FM2015.

De forma a ilustrar isto, consideremos um dos principais aspectos envolvidos na apresentação visual do sistema de um jogo: a metáfora da câmera. A câmera, quando presente, é utilizada para definir um recorte no espaço do jogo. No caso do engine do MissionMaker, a câmera representa uma perspectiva em primeira pessoa, mas há outras alternativas possíveis.

Por exemplo, na maioria dos jogos bidimensionais a perspectiva que o jogador tem do universo virtual não se dá através do olhar do avatar, mas sim de um ponto de vista externo a ele. Isso permite ao jogador ver a sua própria representação no contexto do espaço físico. É o caso dos jogos de "plataforma" - no estilo de Super Mario Bros. (Nintendo, 1985) -, nos quais a "câmera" é lateral, ou em Pac-Man, cuja "câmera" mostra a ação do topo.

Esta perspectiva externa também é utilizada em jogos 3D, principalmente de ação, corrida ou plataforma, como em Psychonauts (Double Fine, 2005; fig. 39, página anterior). Neles, a câmera encontra-se geralmente deslocada para trás do avatar, de forma que o jogador tenha uma percepção melhor quanto ao seu posicionamento no ambiente 3D.

Em determinados gêneros, o jogador não controla um avatar no sentido tradicional, considerado como uma figura humana individual, mas sim um grupo ou entidade maior. É o caso dos jogos de estratégia, como na série StarCraft (Blizzard Entertainment, 1998-; fig. 39), nos quais o jogador pode controlar múltiplos personagens simultaneamente, em uma abstração da função de um general. Neste caso, a câmera pode mostrar conjuntos de personagens, e muitas vezes o jogador pode deslocar a sua perspectiva de maneira independente à ação no jogo (de certa forma tornando-se uma entidade onipresente, capaz de flutuar pelo espaço físico).

Uma abstração de nível ainda mais alto para o modelo de apresentação pode ser encontrada em jogos nos quais nem ao menos existe uma representação ou simulação interna da fisicalidade dos personagens como entidades individuais. É o caso dos jogos de esporte nos quais o jogador assume o papel do técnico ou gerente de um time, como em Football Manager 2015 (Sports Interactive, 2014; fig. 39). Nestes jogos, as partidas não são representadas visualmente - apenas o resultado delas é calculado, de acordo com estatísticas e outras informações sobre os times envolvidos, o campo e assim por diante (note que, embora existam representações internas para cada jogador individual na simulação, esta se dá na forma de números e estatísticas - não há representação física no mundo virtual). Neste caso, a metáfora da câmera é abandonada completamente, pois nem ao menos existe uma representação do espaço físico.[148]

[148] Note que, neste caso, de certa forma o modelo de apresentação reflete a natureza abstrata do próprio modelo de representação. Fonte das imagens: Psychonauts - URL: `firsthour.net/first-hour-review/psychonauts`; StarCraft - URL: `www.sc2blog.com/2007/11/26/three-new-starcraft-2-screenshots-zerg-attack/`; FM2015 - URL: `www.sigames.com/content/football-manager-2015`

Assim como os outros aspectos e elementos que compõem o modelo de apresentação, a metáfora da câmera trata-se de um recurso procedural que pode ser empregado de maneira expressiva - por exemplo, a partir dos diferentes ângulos de câmera, aberturas de lente e estilos de composição, ou mesmo na escolha por não utilizar uma câmera virtual. Mas este é apenas um entre os inúmeros elementos que podem compor um modelo de apresentação visual.

Outro elemento que faz parte do modelo de apresentação e pode ter um papel fundamental na estética de uma produção é a definição do estilo da renderização. Vimos que no MissionMaker o estilo escolhido para a renderização do ambiente virtual é predominantemente realista. Isto é, a metáfora da câmera busca emular não apenas a perspectiva de como um ser humano veria uma cena similar no mundo real, mas também a maneira como as fontes de luz dispostas no ambiente virtual reagem à sua topologia e às diferentes texturas presentes, e como esta informação chega aos "olhos" do avatar.

Entretanto, esta renderização poderia ser diferente, levando a outras estéticas. Por exemplo, diminuindo os parâmetros de contraste e saturação de cor, é possível obter uma aparência envelhecida, ou contribuir para sugerir uma estética noir; aumentando eles é possível transmitir intensidade ou expressar agressividade, e assim por diante.

No jogo No Man's Sky (Hello Games), mencionado anteriormente, a intenção dos seus criadores é de reproduzir uma estética similar a aquela encontrada nas ilustrações das capas de livros de ficção científica dos anos 70 e 80 - veja a comparação na figura apresentada na página ao lado (fig. 40; p. 177).[149]

É verdade que a similaridade neste caso se deve em grande parte aos próprios ambientes e elementos presentes nas composições - as naves, a topologia dos planetas, sua fauna e flora. Porém, a estética escolhida para a renderização destes ambientes também foi cuidadosamente escolhida, priorizando uma certa combinação de tons e contraste entre as cores, de forma a obter uma aparência similar às ilustrações de referência (KHATCHADOURIAN, 2015).

[149]Na figura, as capas à direita são de Dune (Frank Herbert, 1979; topo), com arte de Chris White; e The Caves of Steel (Isaac Asimov, 1974), arte de Chris Foss.

Figura 40 – No Man's Sky (esq.) e capas de livros de ficção científica

O jogo A Slower Speed of Light (MIT Game Lab, 2012) também explora um modelo de apresentação alternativo, só que neste caso a diferença se encontra no mapeamento da representação tridimensional do ambiente virtual para a tela, que se dá de maneira distorcida, de acordo com variações na velocidade da luz (fig. 41; próxima página).

Trata-se de um jogo experimental que utiliza uma biblioteca de recursos para simulação de física desenvolvida especialmente para o projeto, chamada OpenRelativity, baseada no engine Unity. Note que não há alterações na topologia do ambiente 3D em si (isto é, na representação interna da simulação), apenas na sua apresentação ao receptor.

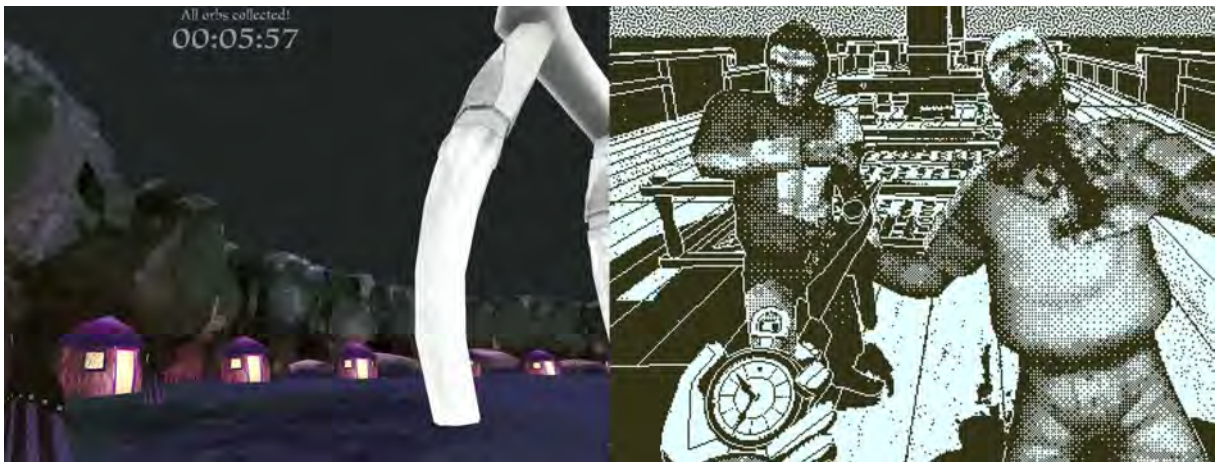


Figura 41 – A Slower Speed of Light (esq.) e Return of the Obra Din

Outra renderização alternativa que é relativamente comum em jogos 3D é a estética do "cartoon". Neste modelo, os elementos do jogo e o espaço tridimensional tem a aparência de desenho animado, geralmente com contornos pretos grossos e bem definidos, paleta de cor reduzida e uma simulação de luz e sombra menos realista. No jogo Return of the Obra Dinn (Lucas Pope), o modelo de apresentação é ainda mais simplificado, limitando a renderização a apenas 1 bit (fig. 41, acima) - isto é, no jogo, o ambiente 3D é apresentado utilizando apenas uma cor, sugerindo variações de tonalidade, luminosidade e transparência a partir de efeitos de pontilhado. Esta escolha expressiva de Pope resulta em uma estética similar a aquela dos jogos de computador antigos.[150]

Além da aparência dos jogos, também podem existir modelos conceituais de apresentação para outros aspectos da experiência, como os elementos sonoros ou mesmo cinéticos. Por exemplo, nos videogames modernos o designer do jogo pode fazer o controle vibrar em certas situações. Geralmente este recurso é empregado para refletir algum evento físico na simulação, como um carro batendo em um obstáculo, ou se o jogador é atingido. Entretanto, a princípio a informação cinética também pode ser empregada para outros fins mais subjetivos ou expressivos.

Note que, embora estas funções que lidam com a saída de um sistema sejam recursos procedurais - isto é, conjuntos de instruções executadas pela máquina - o seu produto final é sempre de natureza estática. No caso das funções de apresentação visual, o produto são imagens exibidas na tela; no caso das funções responsáveis pela saída de áudio, o produto é um fluxo de informação contendo uma determinada configuração de ondas sonoras, e assim por diante.[151]

[150] Fonte das imagens: A Slower Speed of Light - URL: `gamelab.mit.edu/games/a-slower-speed-of-light/`; Return of the Obra Dinn (em desenvolvimento) - URL: `killscreen.com/articles/return-obra-dinn-update-details-challenges-1-bit-rendering/`

[151] Isso também significa que estes são os únicos recursos modais que são diretamente acessados pelo receptor, isto é, pelo jogador ou público. Qualquer outro recurso, seja ele estático ou procedural, depende das funções de saída (ou apresentação) para serem visualizadas ou percebidas externamente.

Outro tipo de modelo conceitual particularmente relevante para esta análise são aqueles que se referem à **jogabilidade**, uma vez que, por definição, jogos só podem se realizar de maneira plena a partir da interação do jogador.

O MissionMaker emprega uma metáfora de interação relativamente tradicional, similar a aquela utilizada na maioria dos jogos em primeira pessoa (p. 100). Neste caso, por conta da natureza realista da simulação, muitos destes controles apresentam uma correspondência relativamente direta com o seu efeito no sistema - por exemplo, a tecla direcional para esquerda move o personagem para a esquerda, e o movimento do mouse é mapeado de maneira relativamente intuitiva na perspectiva do jogador. Entretanto, embora este mapeamento possa parecer evidente e intuitivo, trata-se de um modelo arbitrário (assim como todos os outros modelos conceituais). Como tal, ele tem um potencial expressivo ou de significação.

Por exemplo, a própria escolha pela inclusão deste conjunto específico de controles, e a omissão de outros, já reflete um determinado recorte para este modelo. Nos jogos criados no MissionMaker, o jogador tem grande liberdade para se deslocar no espaço físico (e modificar a sua perspectiva em relação a ele), porém o seu controle sobre os movimentos do corpo do avatar em si, e dos seus membros individuais (braços, pernas etc.), é praticamente nulo. A rigor, neste caso o corpo do avatar nem ao menos existe - ele não é representado dentro da simulação (isso pode ser verificado olhando-se para baixo - vê-se apenas o chão, e não o torso e pernas do avatar, como esperado).

Os controles também são definidos de maneira predominantemente arbitrária. Por exemplo, ao pressionar as teclas direcionais para a esquerda ou direita o avatar move-se lateralmente. Este movimento, conhecido como "strafe", é pouco natural, e não corresponde à maneira como as pessoas geralmente se movem na realidade (tendemos a preferir o deslocamento para frente, nos virando sempre que necessário). Por conta disso, alguns designers optam por mapear as setas para a rotação do avatar, e incluem um comando modificador que pode ser usado quando o jogador deseja mover-se lateralmente.

De qualquer forma, é inevitável que o modelo de controle de um personagem em um ambiente virtual seja arbitrário, uma vez que não é possível estabelecer uma correspondência direta entre os comandos disponíveis em um computador (ou qualquer outro dispositivo) e os gestos em si sendo representados (mesmo considerando as tecnologias atuais de realidade virtual). Estas abstrações, assim como todas as demais, também são recursos procedurais, e como tal inevitavelmente carregam significados, ou algum tipo de intencionalidade dos seus criadores.

Por exemplo, o fato do engine do MissionMaker enfatizar o deslocamento espacial do avatar, em oposição aos movimentos e a fisicalidade do seu corpo em um nível maior de granularidade, significa que os jogos criados neste aplicativo inevitavelmente irão refletir esta tendência. O designer não pode, por exemplo, criar uma situação na qual o jogador precisa chamar a atenção de outro personagem balançando o braço, uma vez que este aspecto da simulação não foi priorizado na implementação do engine. De fato, a maioria dos jogos e narrativas produzidas nesta ferramenta tem o seu design fortemente influenciado pelo posicionamento espacial dos personagens no universo virtual, exigindo do jogador deslocar-se entre localidades de forma a realizar a experiência.

O próprio fato de que o engine do MissionMaker padroniza o controle de movimentação do personagem no modo "strafe", mencionado anteriormente, sugere uma ênfase no combate e na ação. Como resultado disso, há uma presença maior deste tipo de jogabilidade nos projetos criados no MissionMaker, uma vez que o designer constata que é mais fácil pra o jogador controlar o avatar em uma situação de ação (por exemplo, correndo rapidamente de um ambiente a outro, e através de corredores e passagens estreitas) do que em uma interação local, em um contexto mais específico (por exemplo, mover-se atrás de um armário, ou examinar os objetos em uma estante).

Abstrações de Nível Médio

Neste estudo, consideramos o nível médio de abstração como sendo a camada localizada abaixo daquela do MissionMaker, no qual reside a linguagem de programação empregada na implementação do aplicativo. A análise apresentada a seguir explora este nível de abstração principalmente na relação lógica e conceitual entre os recursos e funções presentes nesta camada, e aqueles encontrados nas demais camadas, acima e abaixo desta. Em outras palavras, ao invés de focar nas especificidades de implementação de uma linguagem de programação específica, esta análise trata da natureza e das funcionalidades oferecidas pelos recursos presentes neste nível de abstração, e o seu papel no contexto mais amplo de uma produção digital.[152]

Neste nível residem elementos de lógica cujo escopo é mais geral ou amplo, tais como funções fundamentais para a produção de saídas ("output"; por exemplo, gerenciando o envio de informações de cor para os pixels no "buffer" visual, a ser transmitido para uma tela para exibição), o recebimento de alguma entrada ("input"; comandos no teclado, por exemplo), gerenciamento de recursos e dados, e assim por diante.

Outras funções geralmente encontradas neste nível incluem procedimentos para ordenação de dados,

[152] O Missionmaker foi desenvolvido na linguagem C++, e inclui um engine proprietário (isto é, desenvolvido especialmente para este aplicativo). Como veremos, a nova versão da ferramenta está sendo desenvolvida com base no engine Unity.

geradores de números pseudoaleatórios (a função "random") e descrições de estruturas de dados mais fundamentais, entre outras. Cada linguagem de programação tende a oferecer a sua própria implementação para estas funções, eventualmente levando a variações ou particularidades que podem se refletir nas produções realizadas através delas.

As funções presentes neste nível de abstração servem como uma camada intermediária que permite ao programador empregar os recursos presentes no nível inferior, relativos ao funcionamento do hardware (tratadas mais adiante), para a construção de funções mais complexas nos níveis mais altos (como vistas anteriormente).

Por exemplo, considere o modelo responsável pela apresentação visual (ou renderização) do ambiente virtual no MissionMaker, presente no nível de abstração superior, visto anteriormente. Este modelo faz uso de uma série de funções mais genéricas deste nível médio de abstração. Por exemplo, a função responsável por obter as informações referentes à representação interna de algum elemento 3D do jogo - isto é, a posição relativa dos vértices que compõem este elemento, a textura que deve ser usada para cada superfície e assim por diante.

Outro conjunto de funções pode servir para a realização de cálculos específicos, empregados no processo de determinar como esta representação interna do objeto 3D (obtida através da função anterior) deve ser mapeada para uma determinada perspectiva - isso geralmente envolve, por exemplo, funções que realizam cálculos trigonométricos, de forma a determinar o posicionamento da geometria de acordo com a câmera virtual, ou para calcular a rotação e redimensionamento da topologia tridimensional. Finalmente, a apresentação da imagem resultante na tela é feita através de outras funções, responsáveis pela organização das informações (obtidas nos cálculos da função mencionada anteriormente) de forma que possam ser enviadas para um dispositivo de saída, como uma tela ou monitor.[153]

Os recursos presentes nos diferentes níveis de abstração se organizam em uma estrutura hierárquica "aninhada". Assim, aqueles recursos e funções localizados nos níveis inferiores de abstração tendem a ser compartilhadas entre diferentes funções nos níveis mais acima. Por exemplo, a função fundamental responsável por lidar com a criação e manipulação de conteúdo textual pode ser chamada (ou utilizada) tanto pela função que exibe "mensagens pop-up" na tela para o jogador, quanto por aquela que exibe o diálogo entre personagens, ambas presentes no nível mais alto do aplicativo.

[153]Esta última etapa, por sua vez, pode empregar funções do nível mais baixo de abstração, para o envio das informações ao hardware de saída.

Alguns recursos de baixo nível tendem a ser empregados com maior frequência que outros, e compartilhados por uma variedade maior de funções. É o caso da função random, que gera números pseudoaleatórios, presente em praticamente qualquer linguagem de programação, e empregada em uma infinidade de diferentes contextos, tanto para fins práticos quanto expressivos.

Este nível médio de abstração, que se localiza abaixo da camada do aplicativo, mas acima dos mecanismos mais fundamentais da máquina digital, tende a ter uma contribuição mais sutil, e geralmente menos direta, no produto final.

Por exemplo, no MissionMaker a função random é utilizada no comportamento que faz um personagem seguir o outro (presente no nível de abstração mais alto), mencionado anteriormente. O uso desta função permite produzir um comportamento relativamente orgânico - como vimos, o personagem não segue o outro diretamente, mas sim aparenta mover-se de maneira espontânea. Isto é, este comportamento, que na nossa classificação se localiza no nível mais alto de abstração (no engine do MissionMaker) é influenciado pelos mecanismos de uma função presente neste nível médio (na linguagem de programação empregada para a implementação do MissionMaker).

Note que, neste caso, trata-se de uma influência relativamente indireta. Caso esta função tivesse sido implementada de outra maneira na linguagem de programação empregada pelos criadores do MissionMaker (ou se os programadores do aplicativo tivessem modificado a função de alguma maneira), os movimentos realizados pelo personagem provavelmente seriam diferentes. Porém, se esta diferença seria perceptível ou não, esta é outra questão, que pode ser considerada pelo designer ao decidir sobre o emprego deste recurso modal específico no seu processo criativo.

Abstrações de Baixo Nível

Este nível mais baixo de abstração refere-se à camada que se encontra mais próxima dos mecanismos fundamentais da lógica computacional, do funcionamento da máquina digital e do hardware, bem como a sua comunicação com os demais dispositivos que compõem ou se encontram associados a ele. Em comparação com o nível médio, visto anteriormente, as funções e elementos de lógica encontrados aqui são ainda mais genéricos em termos de função, e portanto menos específicos a alguma implementação ou emprego particular, sendo compartilhados por praticamente qualquer tipo de produção digital.

Isto é, enquanto que a implementação particular de uma função no MissionMaker só tem influência no contexto daquele aplicativo em específico (no nível mais alto de abstração), elementos presentes

na linguagem de máquina tem o potencial de influenciar a implementação ou o funcionamento de qualquer produção realizada neste meio (incluindo aplicativos, linguagens de programação de alto nível e o próprio sistema operacional).

A influência que os recursos e funcionalidades deste nível mais baixo de abstração tem sobre os jogos produzidos no MissionMaker é de natureza fundamentalmente distinta, em comparação com aquela das camadas mais altas. Embora estes recursos mais fundamentais não influenciem de forma tão direta as estruturas e elementos definidos pelo designer no editor, ainda assim eles carregam uma série de implicações e significados particulares.

Embora o artista ou criativo não tenha acesso a estas camadas mais baixas de abstração, o conhecimento dos seus mecanismos e características é importante para garantir que a escolha pelo emprego deste meio se dá de maneira informada e consciente. Como vimos, todo meio ou linguagem apresenta as suas próprias diretrizes e restrições, seja na forma da sua estrutura ou gramática particular, ou mesmo por conta das características físicas das ferramentas e elementos envolvidos no processo criativo.

Existem determinadas práticas criativas que se baseiam justamente no questionamento e subversão destas estruturas e sistemas fundamentais. Um exemplo disso em um nível médio de abstração são os experimentos do artista Adrian Ward com programas "kamikazes", que destruíam endereços de memória em que eles próprios estavam localizados (WARD, 2002). Em uma das camadas mais baixas de abstração, no nível do hardware, existe a prática do Circuit-Bending, na qual os artistas modificam os próprios componentes físicos das máquinas, de forma a alterar os seus comportamentos e propriedades fundamentais.[154]

Estes são apenas alguns entre os inúmeros recursos procedurais que fazem parte do engine do MissionMaker, ou que influenciam o seu funcionamento a partir dos níveis mais baixos de abstração. Tratam-se de recursos que não estão ao alcance direto do designer, mas que podem carregar significados, sugerir uma determinada abordagem criativa ou mesmo representar uma ideologia em particular. Por outro lado, embora a influência destes mecanismos e funções "ocultas" seja praticamente inevitável, o seu efeito nas camadas mais altas de abstração tende a ser mais sutil e indireto.

[154] A influência deste nível mais fundamental do meio digital pode ser traçado aos primórdios da computação, na origem da máquina digital como a manifestação de uma postura sistemática para lidar com a informação (além de tratar-se de uma invenção diretamente influenciada por interesses militares). Embora os significados e implicações relacionadas a estas conexões não exerçam necessariamente uma influência direta ou evidente no uso dos computadores modernos, e nas produções realizadas através deles, elas certamente estão presentes de alguma maneira, e contribuem de alguma forma para a identidade do meio digital.

Finalmente, lembramos que os recursos modais **procedurais** diferem dos recursos **estáticos** de maneira fundamental, principalmente pelo fato de que aqueles existem unicamente a partir da sua contribuição ou efeito para a lógica do sistema.

Por exemplo, considere o modelo conceitual para a construção de regras no MissionMaker, que descreve os componentes necessários para a autoria deste elemento de lógica, bem como a estrutura na qual eles devem ser combinados. Este modelo só é relevante a partir do seu efeito no sistema, no momento da execução das funções e operações associadas a uma determinada regra, e no contexto de um determinado jogo.

3.3.3 Compondo Modos

Vimos que o designer empregando o MissionMaker tem acesso a dois tipos de recursos modais: aqueles procedurais, foco deste estudo, que consiste nas variáveis e regras; e os estáticos, que podem ser de natureza textual, visual, sonora e assim por diante. Nesta seção veremos como estes recursos podem ser integrados e relacionados entre si na forma de um sistema coeso (um jogo finalizado, ou algum elemento ou parte dele), de acordo com um determinado plano ou design, a partir do ato da **composição**.

O diagrama abaixo representa como os recursos modais são organizados no contexto do MissionMaker, incluindo aqueles que estão presentes apenas no engine do jogo (fig. 42). Recursos estáticos também estão presentes no diagrama, uma vez que estes são fundamentais para a criação de jogos nesta plataforma.[155]

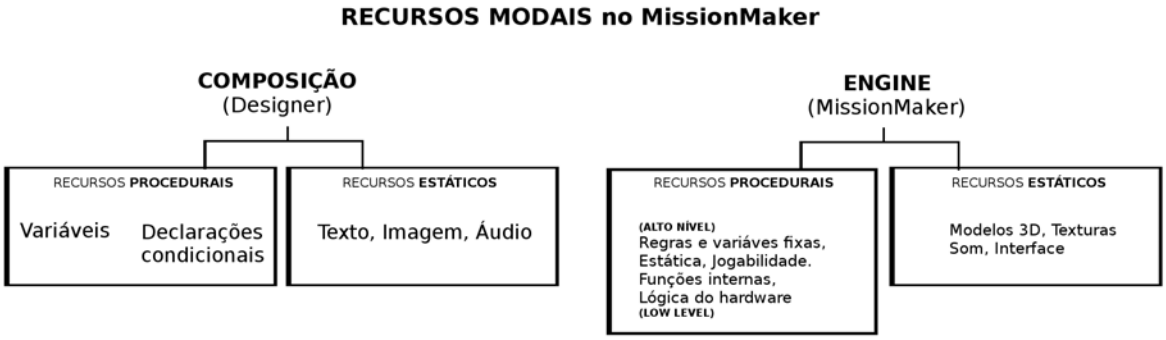


Figura 42 – Diagrama: recursos modais no MissionMaker

Vimos que no MissionMaker o engine enfatiza os aspectos físicos e espaciais do universo de jogo, e que isso se reflete no fluxo de trabalho do próprio editor. Em decorrência disso, neste aplicativo o ato da composição geralmente tem como ponto de partida este mesmo contexto, isto é, na seleção e posicionamento físico dos ambientes, personagens e outros elementos do jogo. Após esta etapa, o designer pode realizar a definição das eventuais variáveis, regras e comportamentos.

[155] Como vimos, no MissionMaker os recursos estáticos são aqueles que não participam da lógica do sistema do jogo - como tal, eles não contribuem na construção de significado procedural, embora possam ser empregados para expressar significados de natureza estática (p. 159).

Gramática

O conceito de composição no contexto de um jogo digital é similar a aquele em outras linguagens tradicionais, tais como nas Artes Visuais, Música ou Cinema. Isto é, envolve a organização de elementos menores e mais específicos com a intenção de formar estruturas maiores e mais complexas, que geralmente expressam ou representam significados que não poderiam ser realizados de outra forma - seja pelos elementos individualmente, ou através de uma composição diferente entre eles.

Neste estudo, a composição é considerada como sendo um ato semiótico, no sentido em que envolve a seleção, combinação e organização de recursos de forma a construir um significado particular. Este processo é análogo ao ato de combinar palavras para formar frases ou declarações. Aplicando esta analogia ao processo criativo no MissionMaker, a "gramática" na qual estas frases podem ser escritas seriam as estruturas e relações particulares através das quais o designer pode compor os recursos modais à sua disposição, e o "texto" seria o jogo finalizado (ou uma parte dele).

Esta gramática interna do MissionMaker molda e serve de ponto de partida para o processo criativo do designer, e se reflete nas produções realizadas no aplicativo. Ela fornece as diretrizes, na forma de regras e estruturas, através das quais o ato da composição pode ser realizado. Além das diretrizes que definem o MissionMaker como uma ferramenta para criação de jogos - em oposição a um processador de textos ou uma linguagem de programação de baixo nível, por exemplo - vimos que este aplicativo também apresenta uma série de restrições ou limitações planejadas para tornar o fluxo de trabalho mais acessível e didático.

O termo "composição" deve ser interpretado no contexto deste estudo como o ato em si mesmo, realizado pelo designer, em oposição a considerar os elementos individuais empregados e os processos internos envolvidos na sua realização, ou ao seu resultado. Por exemplo, no MissionMaker, quando o designer posiciona um determinado personagem em um certo local no espaço do jogo, este é um ato de composição. Este mesmo ato poderia ser descrito a partir dos processos internos correspondentes dentro do sistema - neste caso, isso envolveria, por exemplo, a operação de modificar a propriedade relativa à posição deste elemento no espaço 3D. De qualquer forma, na nossa análise, interessa principalmente o ato da composição da perspectiva do designer.

A diferença fundamental entre o ato da composição em meio digital e em outras linguagens tradicionais é o fato de que ele pode produzir (ou conter) recursos modais de natureza **procedural**. Além disso, no meio digital o próprio ato da composição pode ser se dar de maneira procedural. Isso se dá na forma de um algoritmo ou função que descreve a estrutura ou lógica pela qual a composição deve acontecer.

Com isso, uma determinada composição pode ser executada de maneira dinâmica, de forma repetida ou contínua, aplicada individualmente ou a um conjunto de elementos (desde que estes cumpram com as especificações da função). Este é o caso das regras (Rules) no MissionMaker, por exemplo - a maneira como este recurso é estruturado representa um tipo específico de composição, envolvendo um determinado conjunto de elementos.

Composição Como Recurso Modal

O ato da composição também pode ser considerado como um recurso modal, uma vez que tem o potencial para representar ou expressar significado. Na prática, qualquer recurso modal minimamente complexo pode ser interpretado como sendo uma composição de recursos menores, levando a uma hierarquia similar a aquela tratada anteriormente no caso dos diferentes níveis de abstração. O tipo de significado sendo representado ou expressado por um determinado ato de composição - se será de natureza estática ou procedural - é diretamente relacionado aos elementos envolvidos na sua construção, como eles são organizados e se relacionam entre si.

Por exemplo, consideremos o ato de posicionar um objeto no espaço virtual do jogo. Se esta escolha particular de composição não afeta outros elementos do sistema - sejam eles objetos, personagens ou regras e comportamentos -, e se o objeto em si for de natureza estática, este gesto não pode produzir um significado de natureza procedural. Nos jogos digitais, este é o caso dos objetos de decoração, elementos comuns em diversos gêneros, com os quais o jogador ou outros personagens não podem interagir, e que não se encontram associados de maneira alguma às regras ou comportamentos presentes na simulação (exceto por funções especificamente voltadas para a apresentação).

Um ato de composição como este - o posicionamento de um objeto - pode ser considerado como um recurso modal em si se esta escolha expressa ou produz algum significado adicional, além daquele que eventualmente já esteja presente nos elementos individuais envolvidos - neste caso, o objeto sendo posicionado (e, talvez, o ambiente no qual ele está sendo posicionado).

Por exemplo, considere um jogo de investigação, no qual o jogador faz o papel de um detetive. O designer pode escolher incluir no quarto de um dos suspeitos um prego preso à parede, de forma a sugerir que havia um quadro pendurado ali, mas que foi removido (servindo como uma pista para que o jogador investigue sobre este quadro). Cada um destes elementos individuais - o prego e a parede - não expressam nada por si só, mas a composição específica entre eles produz um significado.

Note que, neste caso, o significado produzido por esta composição é de natureza **estática**. Entretanto, esta mesma situação pode ser modificada de forma a se configurar como um recurso de natureza **procedural**.

Por exemplo, suponha que exista um NPC (sigla para "non-player character", empregada em referência aos personagens ativos do jogo que não são controlados pelo jogador) que serve como um ajudante para o jogador, capaz de reconhecer inconsistências no ambiente, como a ausência de um objeto. Uma forma de implementar isso seria a partir de uma biblioteca de combinações comuns entre elementos, com base na qual este NPC analisaria os objetos ao seu redor. Esta biblioteca teria pares (ou combinações maiores) de elementos considerados comuns, como no exemplo abaixo.

```
[quadro, prego],
[pires, xícara],
[cama, travesseiro, cobertor],
[pia, escova de dentes, torneira],
[...]
```

Cada vez que este NPC se depara com um novo objeto, ele consulta esta biblioteca. Caso encontre o objeto em um dos grupos, ele então verifica se ele se encontra associado aos demais objetos no mesmo grupo. Assim, por exemplo, ao ver um travesseiro, o NPC irá verificar se ele se encontra próximo de uma cama e de um cobertor. Caso contrário, ele pode reagir comunicando ao jogador que há algo estranho naquela situação.[156]

Voltemos agora à situação original, do prego na parede, desta vez considerando a existência deste novo comportamento do NPC, descrito acima. Nesta nova situação, aquela mesma escolha de composição por parte do designer passaria a atuar como um recurso de natureza procedural, uma vez que desta vez ela teria uma influência na simulação. Isto é, a presença do prego sozinho, sem um quadro, levaria o NPC a reagir de uma determinada maneira.

Este é apenas um exemplo de como um ato de composição pode se constituir como um recurso procedural. Há inúmeras outras possibilidades pelas quais isso poderia ocorrer, incluindo a partir de processos envolvendo elementos diferentes, e de diferentes naturezas, e em outros níveis de abstração. Por exemplo, como vimos, a própria estrutura da regra, presente no editor do MissionMaker (isto é, em um nível de abstração alto), trata-se de um modelo de composição.

[156] Este exemplo não se refere a uma linguagem de programação específica (embora seja possível implementar um comportamento similar no MissionMaker, a partir de determinadas regras e funções presentes neste aplicativo). Além disso, o comportamento sugerido aqui é intencionalmente simples, de forma a servir ao seu propósito didático. Há inúmeras regras e propriedades que poderiam ser adicionadas de forma a torná-lo mais interessante e sofisticado. Por exemplo, utilizando como base esta mesma biblioteca, seria possível implementar um processo inverso, que permite ao NPC identificar também quando há combinações inesperadas entre objetos.

3.3.4 Comparando Modos (Recursos Compartilhados)

Vimos como o designer de uma narrativa ou jogo digital pode criar, selecionar e compor recursos modais no MissionMaker de forma a representar ideias e construir significados. A seguir, de forma a elaborar e aprofundar esta investigação, propomos uma análise comparativa entre esta prática e aquelas em outras linguagens e formas expressivas - ou Modos - tradicionais.

Examinaremos como podem ser identificados recursos modais compartilhados entre o processo de criação de um jogo no MissionMaker e as outras atividades criativas realizadas na oficina Playing Beowulf. Também será feita uma comparação entre o uso destes recursos pelo designer em cada caso, com base nas entrevistas conduzidas com os participantes da oficina. Como veremos, algumas destas intersecções são mais aparentes, enquanto que outras acontecem entre recursos cujo efeito ou influência é menos direta nas produções realizadas nas respectivas linguagens.

Lembrando que, no contexto destes estudos práticos, o aspecto procedural do meio digital está sendo considerado como um Modo, ou linguagem, e a escolha por empregar este Modo é considerada como significativa.

Modos Como Grupos de Recursos

De forma a realizar esta análise, consideramos que as diferentes linguagens e formas expressivas podem ser descritas a partir de grupos de recursos modais à disposição do artista ou criativo, bem como da gramática específica para a composição destes recursos em cada uma delas - isto é, os Modos.

Vimos que no MissionMaker os recursos modais existem em uma variedade de níveis de abstração. O processo de identificar os recursos no contexto de um determinado Modo envolve desde a compreensão a repeito da gramática e das características particulares de cada linguagem, passando pela abordagem que diferentes autores podem ter sobre o seu uso, até a perspectiva a partir da qual estas investigações podem ser realizadas.

O diagrama na página seguinte ilustra este processo, utilizando como exemplo as linguagens exploradas nas atividades da oficina Playing Beowulf, incluindo o design e criação de jogos no MissionMaker (fig. 43; p. 190). Na figura, os pontos representam os diferentes recursos modais individuais, e cada agrupamento indica uma determinada linguagem, como Desenho, Cinema ou Artes Dramáticas. As intersecções entre estes agrupamentos simbolizam os recursos compartilhados entre eles. Os pontos no diagrama também podem representar recursos estáticos, porém no contexto desta análise o foco será naqueles de natureza procedural.

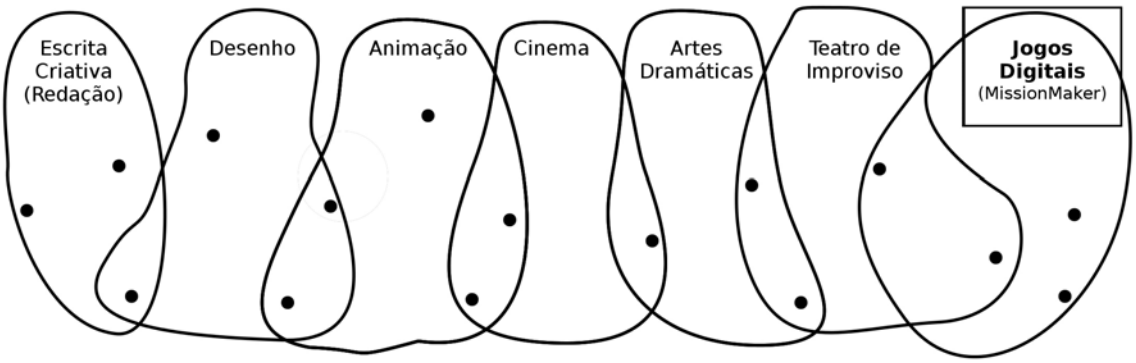


Figura 43 – Diagrama: recursos modais agrupados por linguagem

Note que este diagrama trata-se de uma representação consideravelmente simplificada destes elementos e relacionamentos. Por exemplo, como mencionado, os pontos no diagrama não diferenciam entre recursos procedurais e estáticos. Além disso, eles não representam os níveis de abstração de cada recurso representado. A quantidade de pontos também não pretende representar o número de recursos presentes em cada Modo, ou nas intersecções entre eles, e tampouco a proporção entre os diferentes grupos.

Os agrupamentos mostrados no diagrama não devem ser considerados como rígidos ou fixos. Embora existam certos conjuntos de recursos que geralmente se encontram associados a determinadas linguagens, a delimitação de cada uma delas é dinâmica e flexível.

Além disso, os grupos apresentados aqui são essencialmente arbitrários. Outras configurações poderiam ser definidas de forma a representar perspectivas alternativas sobre cada uma destas linguagens (ou mesmo propondo novas linguagens) e os recursos que as compõem, bem como os usos que diferentes artistas e criativos podem fazer delas.

As próprias intersecções entre os grupos mostrados no diagrama também são representados de maneira simplificada, uma vez que geralmente cada Modo compartilha recursos com uma infinidade de outros Modos (ao invés de apenas um ou dois, como mostrado na figura).

Por exemplo, enquanto que "cor" trata-se de um recurso compartilhado por praticamente todas as linguagens que envolvem um componente visual (desde pintura e colagem, passando pela escultura, até as mídias digitais), "textura" é um recurso relativamente menos presente. Indo além, recursos específicos ao meio digital, como as funções em uma linguagem de programação, por exemplo, são compartilhados por ainda menos Modos.[157]

[157] Note que o fato de um recurso estar presente em um determinado Modo não significa necessariamente que ele será empregado pelo artista ou criativo. Trata-se de um potencial, que pode ou não ser explorado.

Quanto aos níveis de abstração, por um lado, recursos presentes em camadas mais baixas tendem a ser compartilhados por mais Modos, uma vez que o seu uso tende a ser menos específico em termos práticos e conceituais - é o caso de propriedades fundamentais como cor e textura, por exemplo. Por outro lado, certas abstrações de alto nível também podem ser compartilhadas por muitos Modos, quando não se tratam de recursos específicos a um meio ou suporte específico.

Recursos Modais Compartilhados

Entre as atividades criativas realizadas na oficina Playing Beowulf, algumas envolviam múltiplas linguagens e formas expressivas, empregadas de diferentes maneiras e em combinações diversas, inclusive simultaneamente.

Este foi o caso da atividade de criação de kennings, por exemplo. Os kennings são uma forma de linguagem da literatura medieval Escandinava, bastante empregada em Beowulf, e que consiste na substituição de um termo ou nome por uma versão composta - por exemplo, "morte" torna-se "sono da espada" e "vento" é substituído por "quebrador de árvores".[158]

Nesta atividade, cada participante deveria criar um kenning, e depois apresentá-lo aos demais de uma maneira performática, tendo uma espada de madeira em punho, que servia para simbolizar a voz do narrador, e incitar o vigor e intensidade característicos da linguagem do poema original. Assim, no caso desta atividade, foram combinados elementos da Literatura (os kennings) e das Artes Dramáticas (na apresentação).[159]

A oficina foi organizada de forma que cada uma destas atividades deveria ser realizada em um período relativamente curto de tempo - primeiro, era feita uma rápida apresentação da atividade, justificando o seu propósito central e o resultado esperado (seja uma performance ou produção), em seguida grupos eram formados (exceto no caso das atividades individuais), e era estipulado um tempo para a realização do exercício, após o qual ele era apresentado para o grupo.

[158] Tradução nossa - texto original: "death" = "sleep of the sword", "blood" = "battle-sweat" ou "life-liquid", "wind" = "breaker of trees"

[159] O kenning trata-se de um tipo específico de epíteto, que por sua vez consiste em um elemento linguístico que se associa a um nome de forma a qualificá-lo. Segundo o tradutor Ary Gonzalez Galvão, o poema possui muitos epítetos, sendo alguns deles kennings, que ele define como uma espécie de metáfora primitiva (GALVÃO, 1992). Também podem haver múltiplos kennings para um mesmo termo - por exemplo, "sangue" pode ser "suor da batalha", mas também pode ser substituído por "líquido da vida".

A duração da atividade dependia da sua natureza - em alguns casos o processo inteiro poderia levar quinze minutos (como no exercício de leitura dramática de trechos do poema), em outros o período excedia uma hora (no caso de atividades mais complexas, como na produção de filmes, na qual os grupos tiveram que planejar, produzir, filmar e editar o material).[160]

Apesar do período curto de tempo para cada atividade, na maioria dos casos as diretrizes estipuladas pelos organizadores garantiam um ritmo de trabalho confortável. Para cada atividade era definido um objetivo principal, de forma a permitir que os participantes se concentrassem nos elementos mais importantes.

Por exemplo, na atividade de desenho, o foco era em representar visualmente determinados elementos do poema (fig. 44, página ao lado) - a estética e acabamento da produção final não eram tão relevantes quanto a compreensão sobre as diferentes estratégias de figuração.

Desta forma, ao término da oficina todos os participantes tinham realizado todos os exercícios, e ainda tiveram tempo para a discussão e reflexão acerca de cada linguagem e processo criativo. Foi uma experiência produtiva e estimulante, tanto para os participantes quanto para os organizadores e pesquisadores envolvidos, uma vez que permitiu perspectivas únicas sobre estas diferentes abordagens, e às noções mais amplas de linguagem, criatividade e expressividade.

Nas entrevistas conduzidas com os participantes, um dos tópicos de discussão foi justamente sobre esta diversidade de linguagens. A maioria dos entrevistados havia refletido sobre este tema, e tinha opiniões relativamente bem formadas sobre ele, principalmente a respeito das particularidades e similaridades entre as diferentes linguagens com as quais elas tiveram contato na oficina.

Algumas destas discussões e reflexões serão apresentadas a seguir, como uma maneira de ilustrar, explorar e compreender em maior profundidade o papel dos recursos modais dentro de uma linguagem, ou Modo. Isso inclui como estes recursos se relacionam entre si, e como eles podem ser compartilhados entre os diferentes Modos.

[160] No caso do exercício de criação de jogos no MissionMaker foi necessário um tempo muito maior. Por conta disso, o segundo dia da oficina foi dedicado exclusivamente para esta atividade.

Figura 44 – Oficina: atividade de desenho

Na época da oficina, a maioria dos participantes eram alunos do curso de treinamento de professores de Inglês e Drama. Naquela ocasião, quase todos estavam cumprindo a etapa de estágio do curso, dando aulas de redação criativa e outros tópicos relacionados, predominantemente para turmas de crianças e adolescentes.

Assim, estes participantes tinham maior familiaridade e interesse nas áreas da Literatura (bem como na linguagem verbal de uma maneira geral), bem como nas Artes Dramáticas. Consequentemente, também tinham um melhor desempenho e maior facilidade para as atividades da oficina mais diretamente relacionadas a estas áreas (este foi o caso do exercício com os kennings, mencionado anteriormente, por exemplo).

Esta familiaridade não se limitava às atividades envolvendo escrita ou drama. Ela também se refletia de alguma forma em praticamente todos os demais exercícios.

Por exemplo, havia diálogos e textos presentes em muitas das produções, e que revelavam claramente a facilidade dos participantes na compreensão do estilo e das estruturas do poema original, e a sua capacidade para produzir construções literárias interessantes e sofisticadas. A facilidade dos participantes com as Artes Dramáticas também era refletida nas outras atividades e produções, como nos filmes e nos jogos produzidos na oficina, uma vez que muitos deles usavam narração ou diálogos gravados pelos participantes.

Linguagem Verbal

A seguir relataremos uma conversa com Niamh Hickey, sobre uma comparação que ela sugeriu entre o processo criativo no MissionMaker e a prática da escrita, no exercício de redação criativa (fig. 45, abaixo). Segundo ela, a criação de um jogo trazia resultados muito mais diretos e imediatos do que escrever uma história.

Niamh considera a escrita um processo mais trabalhoso, pois é preciso sempre partir "do zero", tanto no sentido prático da folha em branco (ou do documento vazio em um editor de texto), quanto no fato de que o autor precisa definir todos os elementos e aspectos envolvidos na obra, desde a sua estrutura, temática, ideias e eventual narrativa, até o estilo e forma da escrita. No caso da criação de um jogo no MissionMaker, por outro lado, Niamh observa que a maior parte destes elementos já vem predefinidos na lógica do aplicativo, e do engine.

Por um lado, estes comentários revelam que Niamh fez uma reflexão relativamente avançada sobre o tema, e os seus argumentos são em grande parte válidos. Entretanto, eles também revelam algumas confusões a respeito de determinados aspectos fundamentais destas duas linguagens.

Figura 45 – Oficina: escrita criativa (apresentação)

A busca de correspondências na estrutura e nos mecanismos entre linguagens de naturezas essencialmente distintas apresenta uma série de dificuldades. A comparação proposta por Niamh é particularmente problemática pelo fato de que, relativamente falando, o MissionMaker opera em um nível

de abstração distinto - mais especificamente, acima - daquele da linguagem verbal. O aplicativo se caracteriza por oferecer abstrações de alto nível, pelas quais os diferentes elementos e mecanismos da lógica computacional se tornam acessíveis ao usuário. A prática da escrita, por outro lado, envolve diretamente os elementos mais fundamentais desta linguagem.

Um paralelo mais adequado para o MissionMaker no contexto da linguagem verbal seria um jogo para crianças composto de blocos contendo palavras que podem ser combinadas para formar frases - isto é, um sistema limitado, porém com um certo grau de flexibilidade criativa. Inversamente, um paralelo mais adequado para a linguagem verbal no contexto do meio digital seria uma linguagem de programação de baixo nível.

Além do mais, o ato da escrita nunca acontece de fato "do zero". Em qualquer forma artística, o processo criativo envolve uma série de etapas, e grande parte do trabalho acontece antes mesmo de qualquer tipo de externalização ou gesto de materialização por parte do criativo Assim, embora na prática a escrita de um texto geralmente comece como uma "folha em branco", o escritor geralmente já passou pelas etapas anteriores de idealização e planejamento. Mesmo quando um artista decide improvisar, ele ainda está trabalhando com base em um repertório, estilo e experiência pessoal, bem como o conhecimento e senso comum compartilhados socialmente em uma determinada época.

No MissionMaker, grande parte dos elementos que podem ser empregados para compor a produção final já estão presentes no aplicativo em si, ou no engine. Isso inclui desde o estilo de apresentação visual e as estruturas lógicas disponíveis ao designer, até os próprios personagens, objetos e localidades que podem ser utilizadas no jogo. Todos estes elementos predefinidos representam recursos que não podem ser explorados expressivamente pelo designer, pelo menos não de maneira direta.

Por exemplo, enquanto que um escritor tem a liberdade para criar uma estória na qual um personagem pode voar, no MissionMaker isso não é possível, pois a ferramenta não possui esta funcionalidade. Isto é, embora certas partes do processo criativo sejam mais imediatas no MissionMaker, isso acontece com o custo de uma perda no potencial expressivo para o designer.

Dito isso, como mencionamos algumas vezes ao longo desta tese, qualquer atividade criativa envolve algum tipo de predeterminação ou limitação, seja prática ou conceitual. Por exemplo, uma pintura é a expressão de um pintor, mas também carrega características do pincel empregado na sua produção. Na literatura, os gêneros literários, ao mesmo tempo que fornecem ao escritor uma estrutura para trabalhar, também podem servir para restringir a sua criatividade. Além disso, de uma maneira mais ampla, todo artista ou criativo é inevitavelmente influenciado pela sua cultura, experiência e repertório particulares.

Teatro de Improviso

Nesta segunda análise, apresentaremos uma conversa com Harriet Piercy sobre as similaridades que ela identificou entre o exercício de criação de jogos no MissionMaker e a atividade baseada no teatro de improviso (fig. 46, abaixo).

Nesta atividade, os participantes foram apresentados a diferentes situações relacionadas ao universo de Beowulf, bem como a cenas do próprio poema, e deveriam interpretar personagens, individualmente ou em grupo, de acordo com determinadas instruções de direção fornecidas por um dos tutores da oficina.

Figura 46 – Oficina: Artes Dramáticas e improviso

Ao contrário do teatro tradicional, no qual os atores geralmente recebem todas as falas e ações que devem realizar na cena, no teatro de improviso são os atores que decidem como irão agir, tendo como base apenas descrições e instruções mais genéricas.

Estas instruções podem incluir, por exemplo, informações sobre um determinado personagem - como aspectos da sua personalidade, sua história e motivação -, uma direção mais ampla relacionada ao

contexto da cena - por exemplo, uma descrição do local, da sua aparência e dos sons presentes - ou determinados eventos aos quais deve-se reagir - o diretor pode, por exemplo, explicar que os atores estão sendo atacados por um exército.

Harriet, assim como os demais participantes da oficina, já tinha alguma familiaridade com exercícios de teatro de improviso. Entretanto, ela nunca havia refletido a respeito da natureza desta distinção, entre o teatro de improviso e a prática tradicional (na qual os eventos tendem a ser predeterminados, e acontecem em uma ordem linear).

Ela explica que isso se tornou claro para ela após o segundo dia da oficina, quando foi realizada a atividade de criação de jogos no MissionMaker. À medida em que aprendia as funcionalidades do aplicativo, e se familiarizava com o conceito do algoritmo, e da natureza procedural do meio digital, Harriet passou a perceber as suas similaridades com a abordagem no teatro de improviso. Ela notou que, assim como o designer de jogos não determina situações específicas que devem acontecer, mas sim um sistema de regras, a direção no teatro de improviso consiste de certa forma no ato de "programar" os atores com determinados comportamentos.[161]

Entretanto, Harriet observa que esta comparação não deve ser considerada de maneira direta. Ela observa que, em comparação com o meio digital, o teatro de improviso tem um potencial maior para produzir experiências imprevisíveis, criativas e subjetivas. Isso se deve ao fato de que a performance é predominantemente baseada na sensibilidade e experiência de um ser humano (o ator), e nas suas capacidades expressivas. Enquanto isso, o personagem em um jogo se limita a seguir as instruções definidas no seu algoritmo, com base nas informações que lhe são fornecidas.

De qualquer maneira, existe uma infinidade de recursos modais que podem ser empregados para compor um personagem em um jogo, e que são compartilhados entre estas duas práticas, desde as suas características físicas, passando pela maneira como ele se move, até a forma como deve reagir a eventos específicos, e assim por diante.[162]

Por exemplo, na nova versão do MissionMaker, atualmente em desenvolvimento, a biblioteca já contém um personagem que pode ser empregado para representar o monstro Grendel. Este personagem

[161] De certa forma, mesmo o roteiro de uma peça de teatro tradicional pode ser interpretada como uma espécie de algoritmo, uma vez que trata-se de uma série de instruções a serem seguidas (esta analogia não considera a diferença entre a objetividade na compilação e execução de um código e a subjetividade na interpretação que um ator tem do roteiro). De fato, é possível criar um código contendo apenas instruções diretas, sem procedimentos, regras ou condicionais - este seria o caso de um algoritmo com baixo grau de intensidade procedural.

[162] Note que estes recursos residem nos níveis mais altos de abstração, uma vez que, como vimos, nos níveis mais baixos as duas linguagens divergem de maneira significativa.

se caracteriza por apresentar uma postura mais agressiva e soturna que os demais. Trata-se de uma caracterização similar a aquela adotada pelos participantes da oficina para representar Grendel nos exercícios dramáticos da oficina. Entretanto, embora nos dois casos os personagens compartilhem desta mesma postura, no jogo isso é implementado a partir de variáveis e regras, enquanto que no exercício de interpretação o ator emprega processos mentais.

Niamh, a outra participante mencionada anteriormente, também discutiu sobre as similaridades entre estas duas atividades, e chamou atenção para uma correspondência específica particularmente relevante entre elas - o fato de que nos exercícios de improviso o participante geralmente é colocado em uma posição ativa. Isto é, a similaridade entre o teatro de improviso e a autoria procedural também sugere um paralelo entre o ator e a máquina digital, ambos servindo para interpretar e "executar" as instruções recebidas.

Além disso, no caso dos exercícios de improviso no qual o ator interpreta um personagem, também é possível fazer um paralelo entre a função do ator e aquela do jogador, tomando decisões e agindo de maneira criativa dentro do contexto da cena, ou "simulação".

Por conta disso, Niamh acredita que esta atividade ajudou os participantes a "pensarem com a cabeça de um designer de jogos". Além disso, ela afirma que o exercício de teatro de improviso também contribuiu para facilitar o processo de aprendizado do MissionMaker, e para a compreensão a respeito da lógica por trás da autoria procedural de uma maneira mais ampla.

De fato, alguns dos outros participantes entrevistados em nosso estudo fizeram comentários similares, quanto aos benefícios que as atividades realizadas no primeiro dia da oficina tiveram sobre o exercício de criação de jogos. Entretanto, nenhum deles tenha identificado uma correspondência específica como esta. Também pudemos verificar uma influência similar vinda de outras atividades do primeiro dia, facilitando e contribuindo para a compreensão do processo criativo no design dos jogos. Na maior parte dos casos isso aconteceu de maneira indireta, ou mesmo inconsciente.

Uma comparação similar foi feita anteriormente nesta pesquisa, entre as estratégias procedurais de criação e formas artísticas tradicionais anteriores, como a Arte Conceitual e o teatro de improviso. É o caso do Teatro do Oprimido, método teatral desenvolvido por Augusto Boal nos anos 60, no qual temas políticos e sociais são examinados através de cenas improvisadas, com a participação das próprias pessoas envolvidas nas problemáticas.

Aplicações na Educação

As entrevistas que conduzimos com os participantes da oficina tinham como foco discutir a sua experiência do ponto de vista do processo criativo. Isto é, como designer utilizando o MissionMaker para a criação de jogos, e com ênfase na autoria procedural (e no emprego de recursos procedurais). Porém, também foram discutidos tópicos relacionados aos aspectos educacionais do aplicativo, uma vez que o estudo estava sendo realizado no contexto do projeto Playing Beowulf, cujo foco se dá nas aplicações didáticas do MissionMaker em sala de aula.

Ao longo das entrevistas, alguns dos participantes também apresentaram as suas próprias ideias sobre como a criação de jogos poderia ser empregada em um contexto educacional, de maneira integrada a outras linguagens e disciplinas.

Apresentaremos a seguir uma discussão que fez parte da entrevista realizada com Laura Scott, na qual ela descreveu uma possível aplicação do MissionMaker no ensino da redação criativa para crianças, explorando justamente as intersecções - ou recursos modais compartilhados - entre a prática da escrita narrativa e a linguagem dos jogos digitais. Ela exemplificou com o fato de que tanto o escritor quanto o designer de jogos se dedicam à construção de mundos ficcionais, dos seus personagens, e dos relacionamentos entre eles.

A ideia de Laura consiste em realizar estas duas atividades como parte de um mesmo processo criativo, e atuando sobre um mesmo projeto, porém de maneira alternada, para que cada nova iteração possa informar e talvez trazer novas perspectivas sobre a próxima. Isso poderia ser feito individualmente, com o mesmo aluno realizando os dois tipos de atividade, ou entre dois ou mais alunos.

Por exemplo, ela explica que o processo pode começar com um aluno escrevendo uma narrativa simples. Em seguida, outro aluno leria este texto, e criaria o protótipo de um jogo baseado nele. O primeiro aluno iria então testar este jogo e, se desejar, revisar o texto original que havia escrito, com base nas suas impressões. A partir daí, o processo pode se repetir quantas vezes forem necessárias, com cada nova iteração em uma das linguagens sendo informada pela iteração anterior da outra.

O processo também poderia começar com a criação do jogo, e o produto final pode ser tanto o texto quanto o jogo, ou mesmo uma combinação de ambos. Adicionalmente, Laura sugere que o primeiro aluno, além de produzir o seu próprio texto, também poderia escrever um relato secundário, no qual descreve a sua experiência com o jogo. Esta produção secundária seria similar a um "passo a passo" do jogo (*ou "walkthrough"*), que de certa forma pode ser considerado como um gênero de escrita em si mesmo (BURN, 2006). O "passo a passo" poderia ser usado tanto pelo próprio aluno, no processo

de revisão do seu texto, quanto pelo criador do jogo, que por sua vez poderia realizar alterações na sua produção.

Segundo Laura, a intenção deste exercício seria de criar uma correspondência ativa entre estas duas linguagens na mente dos alunos, de forma a incentivar a compreensão dos mecanismos e características de cada uma delas. Ela acredita que este processo poderia levar a soluções e estratégias expressivas que nunca seriam alcançadas de outra maneira - tanto individualmente, em cada linguagem, quanto de maneira combinada.

Estas são apenas algumas das maneiras pelas quais é possível realizar comparações entre os diferentes Modos, a partir dos recursos compartilhados entre eles. Assim como estas, há inúmeras outras perspectivas que poderiam ser adotadas. Além disso, poderiam ser consideradas comparações com outras linguagens e meios, que não foram abordadas aqui. Por exemplo, como mencionamos, as práticas da Arte Conceitual e Arte Cinética também podem apresentar paralelos e similaridades significativas com o design de jogos e com a autoria procedural de uma maneira mais ampla.

Traduções Modais

Vimos que o meio digital pode ser considerado como uma metamídia, pois é capaz de emular ou conter em si virtualmente qualquer outro meio ou linguagem. Isso é possibilitado, ou viabilizado, principalmente pelo seu aspecto procedural que, como argumentamos nesta tese, trata-se de uma linguagem, ou Modo, em si mesmo. Os jogos digitais refletem de maneira evidente este caráter metamidiático, por ser uma linguagem composta de recursos modais vindos de uma infinidade de outras formas e tipos de conteúdo, como gráficos, som, texto, comportamentos, regras e assim por diante.

Em alguns casos, estas linguagens estão presentes na forma de recursos procedurais. Por exemplo, é o caso de personagens cujos movimentos são definidos em tempo real, ao invés de baseados em animações predefinidas; ou então do uso de música generativa, que se modifica de acordo com a situação do jogo - a música pode mudar para um tom menor (mais soturno) se o jogador estiver próximo da derrota.

No caso do MissionMaker, a presença destas outras linguagens e formas artísticas se dá de maneira relativamente limitada, e predominantemente na forma de recursos estáticos. Por exemplo, as Artes

Dramáticas se encontram de certa forma representadas no engine do MissionMaker, através das animações e comportamentos dos personagens. Porém, isso se dá de maneira limitada, tanto em termos da simulação física - isto é, as animações em si são predefinidas - quanto no comportamento - os personagens só podem reagir a situações e eventos para os quais foram programados.

Esta limitação influencia diretamente a atuação do designer que emprega esta ferramenta. Por exemplo, isso significa que não é possível modificar a amplitude de uma animação do gesto de um personagem, e tampouco adicionar novos gestos.

No MissionMaker há uma flexibilidade muito maior em relação ao uso de conteúdo estático nos jogos. Por exemplo, há poucas limitações quanto aos arquivos de som ou música que podem ser importados no aplicativo. Além disso, este conteúdo pode ser reproduzido em praticamente qualquer contexto da simulação, e ser associado à maioria dos eventos e elementos do jogo. É possível iniciar uma música quando um certo personagem adentra um ambiente, produzir um efeito sonoro na mudança de estado de um determinado objeto e assim por diante.

Por conta disso, o som e a música tendem a ser explorados pelos usuários do MissionMaker como uma maneira de compensar por outras limitações, nos demais aspectos da ferramenta. Como vimos, uma das maiores dificuldades encontradas pelos participantes da oficina na criação dos jogos foi a caracterização dos personagens, uma vez que a versão do MissionMaker empregada naquela atividade não oferecia os modelos 3D necessários para representá-los. Assim, alguns dos participantes gravaram a si mesmos interpretando falas, usando um tipo de voz e entonação que permitiriam expressar e comunicar a identidade e personalidade dos personagens para o jogador.

O uso de recursos textuais e visuais estáticos também é relativamente flexível no MissionMaker, embora de maneira mais limitada do que no caso do áudio, tanto em termos do conteúdo em si, quanto do contexto de sua apresentação. Texto e imagens externas podem ser incluídos no jogo principalmente na forma de "mensagens pop-up", ou diretamente no ambiente do jogo, de uma maneira limitada.

3.4 Algumas Considerações

Ao longo dos estudos práticos apresentados neste capítulo, investigamos os recursos e mecanismos de autoria disponíveis no MissionMaker, através dos quais os usuários podem criar narrativas e jogos digitais nesta ferramenta, com ênfase no uso das estratégias procedurais de criação.

No jogo criado por Andrew Smith, a mãe do monstro Grendel é caracterizada como um personagem forte e ao mesmo tempo pacífico; na produção de Tracey Matthews é estabelecido um laço de lealdade entre os guerreiros e Beowulf - em ambos os casos isto foi implementado de acordo com a gramática ou lógica interna deste aplicativo, a partir da definição de regras, condicionais e outros recursos procedurais.

Estes exemplos são simples, porém servem para ilustrar o emprego expressivo do meio digital, e como ele se dá a partir de uma abordagem fundamentalmente distinta daquelas de outras linguagens e formas tradicionais. Ao acompanhar o processo criativo dos jogos analisados neste estudo, foi possível constatar que a expressividade procedural exige novas formas de pensar sobre linguagem, representação e expressividade (além de novas estratégias de fruição e interpretação por parte do receptor).

Metamídia

Uma reflexão importante que resultou destes estudos práticos foi a realização de que, apesar da natureza sistemática, objetiva e discreta da lógica computacional, não existem modelos ou notações preestabelecidas pelas quais as mídias digitais podem ser empregadas de maneira expressiva por artistas e criativos, e tampouco sobre como ela pode se manifestar ou materializar para o público.

É verdade que todos os algoritmos e funções por trás de cada uma das operações e ações realizadas por um computador precisam seguir estritamente as regras e estruturas determinadas por alguma linguagem de programação particular, pela lógica do sistema operacional ou do hardware. Entretanto, não existe uma gramática ou notação única predefinida para os sistemas e comportamentos que podem ser construídos através delas, ou que resultam ou emergem da sua execução. Isso se relaciona ao fato de que, como vimos antes, o meio digital se configura como um metamídia, capaz de representar as qualidades e mecanismos de outras linguagens, incluindo formas completamente originais (p. 27).

Por um lado, o processo criativo de quem se expressa através de algoritmos compartilha muitos aspectos e elementos com formas artísticas estruturadas e de notação forte, tais como a Literatura

e a Música. Por outro lado, o momento da execução de um programa (ou o seu produto) pode ter mais em comum com formas de notação fraca, cuja gramática é menos estruturada, tais como as Artes Visuais, a Dança, a Arte Conceitual e as Artes Dramáticas e do improviso.

Jogo Digitais na Educação

Embora o foco destes estudos práticos sobre o MissionMaker tenha sido na investigação dos aspectos criativos e expressivos da autoria procedural, esta pesquisa também se beneficiou do caráter educacional do projeto Playing Beowulf, principalmente no que se refere ao tema da literacia procedural.[163]

Na Inglaterra, estudos mostram que iniciativas voltadas para o ensino da programação de computadores tendem a se concentrar nas aplicações mais práticas desta atividade, em oposição ao potencial criativo e expressivo do meio digital. O pesquisador Bruno de Paula relata uma pesquisa feita sobre o assunto na qual foi constatado que "[e]nquanto em casa os alunos usavam as tecnologias digitais de uma maneira criativa e comunicativa, nas escolas, o uso [...] era destacado como 'limitado' e 'frustrante'" (de PAULA; VALENTE; BURN, 2014:47).[164]

Comparando o ensino da programação com o da linguagem verbal, seria como se os alunos tivessem aula de gramática, na qual aprendem a escrever e ler, porém não tivessem aula de redação, na qual aprenderiam a aplicar esta linguagem na prática, comunicando e expressando ideias. Além disso, nesta analogia, seria como se a linguagem verbal nem ao menos fosse empregada nas demais disciplinas (Geografia, História, Biologia e assim por diante).

O projeto Playing Beowulf visa suprir esta deficiência, propondo uma abordagem interdisciplinar, possibilitando integrar a programação de jogos à própria ementa das escolas, bem como na curadoria de museus e outros espaços expositivos.

Existem cada vez mais iniciativas promovendo a literacia procedural, como é o caso do projeto Hour of Code (Code.org), voltado para ensinar crianças a programar através de uma plataforma dinâmica e interativa para a realização de exercícios práticos. O projeto foi criado nos Estados Unidos, porém hoje já está presente em diversos países, incluindo o Brasil. Outra iniciativa na América Latina foi a criação em 2011 de uma representação para a organização Games for Change, com direção do pesquisador Gilson Schwartz, focando no uso de jogos digitais para o desenvolvimento social.[165]

[163] Este período de pesquisa sanduíche foi desenvolvido na área da educação, no Institute of Education (Instituto da Educação) da UCL.

[164] Na citação, de Paula descreve um estudo do pesquisador David Buckingham.

[165] Schwartz também publicou o livro Brinco, Logo Aprendo: Educação, Videogames e Moralidades Pós-Modernas (SCHWARTZ, 2014), no qual trata sobre games, educação e comunicação digital. Games for Change (América Latina). URL: `www.gamesforchange.org.br`

No final de 2015, no Reino Unido a programação de computadores passou a ser parte do currículo nacional para crianças a partir dos cinco anos de idade (GOV.UK, 2013). Nos Estados Unidos uma nova legislação passou a reconhecer a ciência da computação como uma disciplina tão importante quanto a Matemática e o Inglês (KOH, 2015).

Esta popularização na prática da programação de computadores também tem se refletido na própria temática de alguns jogos digitais. Nestes títulos, a interpretação e manipulação de código ou algoritmos tem um papel fundamental na mecânica de jogo. Exemplos incluem o jogo de ação Hack 'n' Slash (Double Fine, 2014), o adventure/RPG Else Heart.Break() (Erik Svedäng, 2015), a produção experimental Pony Island (Daniel Mullins, 2016) e The Magic Circle (Question, 2015), no qual o jogador controla o protagonista de um FPS inacabado e repleto de falhas.

Para mais informações relativas ao aspecto educacional de Playing Beowulf, bem como sobre outros projetos sendo desenvolvidos no DARE (LKL), refira ao site do grupo: `darecollaborative.net`

3.5 Futuros Desenvolvimentos

Nesta última seção deste capítulo, apresentaremos a segunda versão do MissionMaker, denominada aqui como MissionMaker2, atualmente em desenvolvimento (em fase alpha). O foco será na nova estrutura e fluxo de trabalho para a criação e edição de regras, que permite múltiplas condicionais (e ações), e operações booleanas. Estes novos recursos resultaram em parte do trabalho de pesquisa e desenvolvimento do qual eu participei durante minha pesquisa sanduíche junto ao projeto.

Também apresentaremos algumas das ideias planejadas para as próximas iterações do MissionMaker2, investigando como determinadas estruturas linguísticas do poema podem ser representadas na própria lógica de funcionamento das regras e comportamentos do engine, bem como no seu processo de criação e manipulação no contexto do editor.[166]

No contexto desta pesquisa, os estudos de caso apresentados neste capítulo nos interessam principalmente a partir das reflexões e elaborações teóricas que derivaram deles, em oposição aos objetos de análise propriamente ditos. Isto é, as análises do aplicativo em si, e dos jogos nele produzidos, servem principalmente como um meio através do qual realizamos a nossa investigação central, a respeito do emprego expressivo das estratégias procedurais de criação.

É a partir desta mesma abordagem e perspectiva que apresentaremos a nova versão do aplicativo nesta seção. Da mesma forma, nos momentos em que comparamos o MissionMaker2 com a versão anterior do aplicativo, isso é feito principalmente com o propósito de evidenciar determinados aspectos e elementos da ferramenta que consideramos relevantes para esta pesquisa.

3.5.1 MissionMaker2

Embora o MissionMaker2 seja diretamente baseado na estrutura, fluxo de trabalho e lógica da versão anterior do aplicativo, a implementação desta nova versão é original (exceto pelo engine em si - como vimos, nesta nova versão está sendo empregado o Unity). Assim, por um lado, a maior parte dos conceitos e ideias do primeiro MissionMaker se aplicam aqui - mesmo as regras e condicionais, que nesta nova versão apresentam diferenças significativas, tanto em termos funcionais quanto de interface, elas ainda seguem uma lógica análoga à da versão anterior. Por outro lado, como veremos, há diferenças fundamentais na forma como muitas das funcionalidades e aspectos do aplicativo estão sendo implementadas, e que podem influenciar o processo criativo dos designers utilizando a ferramenta.

[166] O termo "versão" é empregado para distinguir entre o primeiro MissionMaker e o MissionMaker2. O termo "iteração" é empregado em referência a novos desenvolvimentos feitos nesta nova versão do aplicativo.

No momento da finalização desta tese, o MissionMaker2 ainda estava em fase alpha de desenvolvimento, com algumas funcionalidades ainda não implementadas, e apresentando alguns bugs. Os estudos práticos desta pesquisa, apresentados ao longo deste capítulo, foram realizados utilizando a primeira versão. Como o MissionMaker2 ainda não estava sendo empregado nas oficinas durante este período, a apresentação a seguir foi feita principalmente a partir da minha própria experiência testando o aplicativo, criando jogos e experimentos. Também tomei como base a pesquisa do brasileiro Bruno de Paula, que fazia parte da equipe do Playing Beowulf antes da minha entrada no projeto (de PAULA, 2015), bem como os guias (ou manuais) para os aplicativos, desenvolvidos pela equipe de desenvolvimento.[167]

Editor

O fluxo de trabalho no editor do MissionMaker2 se baseia em grande parte na mesma metáfora da fisicalidade do espaço virtual utilizada no jogo em si (e na versão anterior do aplicativo). Partes da simulação se encontram sempre ativas, mesmo dentro do editor, como o modelo de iluminação, certos comportamentos dos personagens e alguns aspectos da simulação de física. Isso significa que muitos dos recursos procedurais do engine e do projeto se encontram visíveis durante todo o processo de design e criação do jogo, enfatizando este aspecto da simulação para o usuário.

Assim como no primeiro Missionmaker, nesta segunda versão o design de um novo jogo começa com a criação de localidades. Neste modo de edição, a tela se divide em duas seções principais (fig. 47; próxima página): à esquerda, há uma área maior contendo uma grade (ou quadriculado) na qual as localidades podem ser inseridas, e à direita há um menu contendo as opções disponíveis, na forma de ícones.

A criação e edição dos demais elementos do jogo, como objetos (incluindo os personagens) e regras, ou comportamentos, se dá em outro modo ("Edit Mode"). Por exemplo, selecionando-se a opção referente a objetos, surge uma nova janela no topo, ao centro, contendo dois menus (fig. 48), para criar ("Create") ou editar ("Edit") objetos, respectivamente.[168]

[167] No caso do MissionMaker2, foi empregado como referência um manual relativo à iteração alpha do aplicativo (DARE, 2015).

[168] As imagens mostradas aqui representam um recorte da interface, evidenciando os elementos mencionados no texto.

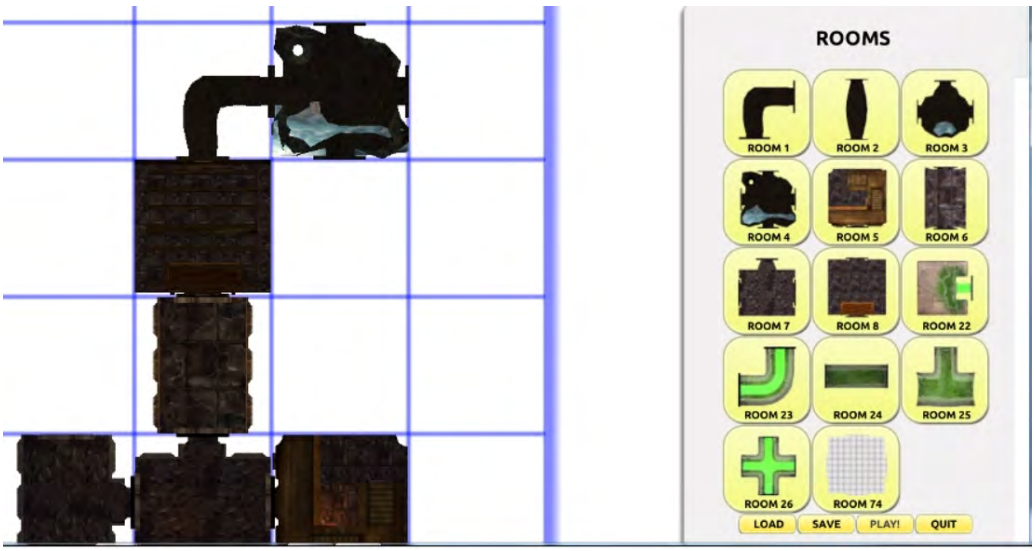


Figura 47 – MissionMaker2: modo de edição de localidades

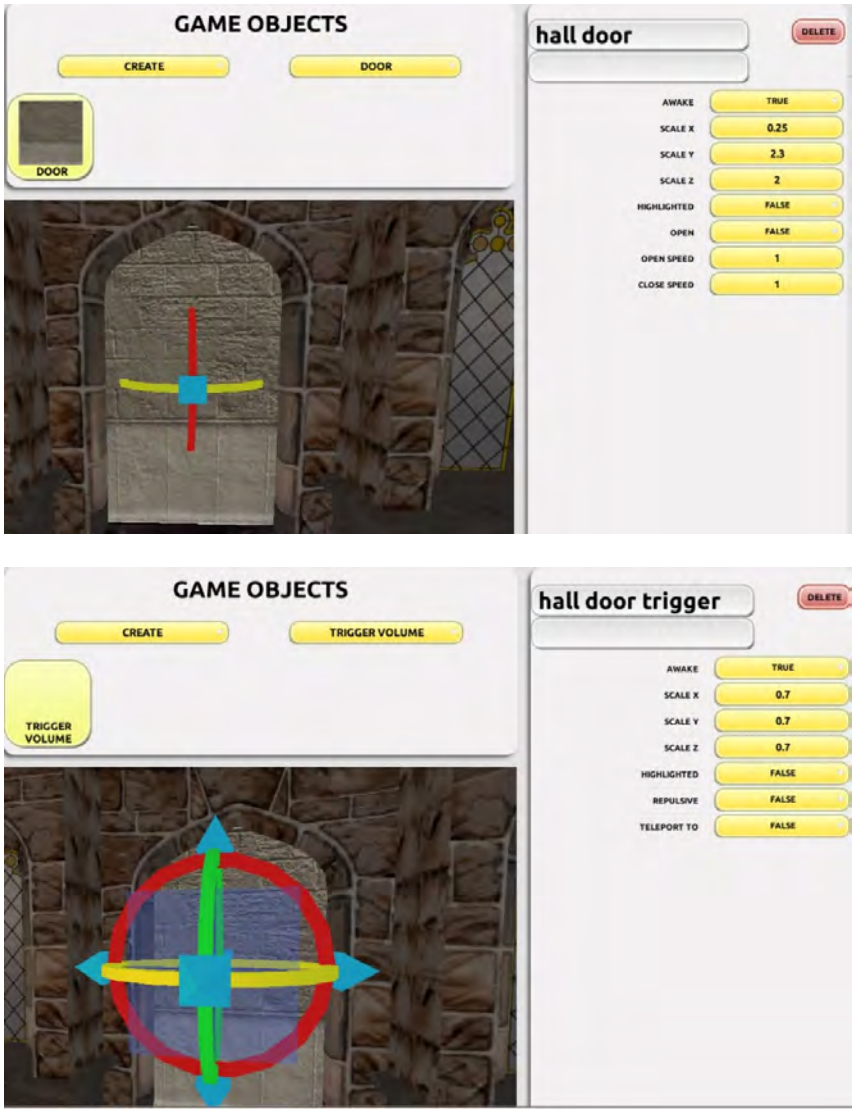


Figura 48 – Criando objetos: porta (topo) e volume de ativação.

As categorias disponíveis são similares a aquelas no primeiro MissionMaker, com algumas poucas diferenças. Porém os objetos em si são diferentes, e mais customizados para a temática do Beowulf. Há uma grande variedade de personagens do poema, com modelos para o rei, uma dama nobre, súditas, diversos tipos de bárbaros e de criaturas (fig. 49, abaixo).

Figura 49 – MissionMaker2: personagens de Beowulf

O sistema de regras no MissionMaker2 segue uma lógica similar à da versão anterior do aplicativo, porém acrescenta funcionalidades e elementos que tornam esta nova implementação significativamente mais poderosa e flexível: a possibilidade de criar múltiplas condicionais para uma mesma regra.

A interface contendo a lógica da regra se divide em duas partes, como pode ser visto na figura a seguir, na região abaixo do título "Hall Door Rule" (fig. 50; p. 211). As condições (ou "conditions"), no topo; e as ações ("actions"), embaixo. Como veremos a seguir, cada condição ou regra exige uma certa quantidade e tipos de parâmetros, de acordo com a lógica e os elementos envolvidos (no topo da janela referente à regra o nome dela pode ser editado; também existem opções para apagar e confirmar a regra).

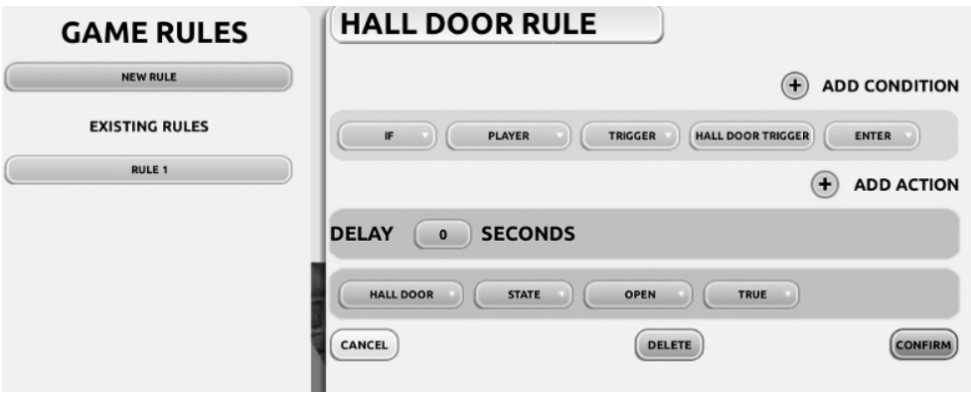


Figura 50 – Criação de uma regra para abrir a porta

O exemplo mostrado na imagem representa uma regra simples para a abertura de uma porta. Ela é composta de apenas uma condição e uma ação, e emprega os objetos mencionados anteriormente - uma porta e um volume de ativação. Neste exemplo, a condição para a regra acontecer é que o jogador entre no volume de ativação criado anteriormente. Seguindo os parâmetros da regra na imagem, a regra será ativada se (If) o jogador (Player) ativar (Trigger) o "Hall Door Trigger" (nome dado ao volume de ativação) entrando nele (Enter).[169]

Se esta condição for atendida, a ação resultante (descrita na parte de baixo da regra), é que a porta em questão se abre. Na lógica do editor, a ação determina que a "Hall Door" (nome dado para a porta) tenha o seu estado (State) de aberta (Open) definido como verdadeiro (True).[170]

Há diferentes estruturas possíveis para uma regra no MissionMaker2, de acordo com os tipos de elemento envolvidos, bem como os aspectos e parâmetros sendo considerados. Por exemplo, na primeira condição, o primeiro elemento de lógica pode ser "If" ou "If not" ("Se" ou "Se não"), e nas demais condições este primeiro elemento podem ser as operações booleanas "And", "And not", "Or" ou "Or not" ("E", "E não", "Ou" ou "Ou não"). Isso permite, por exemplo, criar uma regra que se ativa Se o jogador pegar um determinado item Ou se ele entrar em um volume de ativação - a condição que for atendida antes ativa a regra.

O elemento central (ou sujeito) da condição pode ser um objeto (incluídos aí os personagens), o personagem do jogador ou o jogo em si, considerado como uma entidade mais ampla. De acordo com o elemento selecionado, os parâmetros seguintes da regra podem mudar.

[169] No caso de uma regra que envolve um volume de ativação, é necessário definir, além da entidade que ativa a condição e o volume de ativação (neste caso, o jogador e o volume correspondente), o tipo de interação que ocorre. Neste caso, a ativação se dá quando o jogador adentra o volume. Lógica completa no editor: "IF / PLAYER / TRIGGER / HALL DOOR TRIGGER / ENTER".

[170] O parâmetro "delay" (ou atraso) no topo é de zero segundos, significando que esta ação acontece assim que a condição é atendida (ao modificar este parâmetro, é possível adiar as consequências de uma regra).

Por exemplo, se for escolhido um personagem, o próximo parâmetro pode ser "State", indicando que a condição irá operar com base em um dos diferentes estados possíveis para este elemento - no caso, os estados são "Awake" (significando se ele está ativo ou não), "Highlighted" (se encontra-se selecionado) ou "Aggressive" (atacando o jogador). Em todos estes casos, o parâmetro seguinte é uma booleana - Verdadeiro ou Falso (True ou False). Assim, por exemplo, é possível criar uma regra cuja condição é um determinado personagem estar atacando o jogador ("If Character State Aggressive True", ou "Se Personagem Estado Agressivo Verdadeiro").

Outro possível parâmetro no caso de um personagem é "Number" ("Número"), que pode referir-se a algumas das propriedades numéricas associadas a este elemento na simulação, como energia (ou saúde; "Health") e as suas dimensões ("Scale", ou "Escala"). Neste caso, os próximos parâmetros permitem escolher o valor que a propriedade em questão precisa ter de forma a ativar a regra - por exemplo, uma regra pode ter como condição o jogador estar com o valor de energia menor que 50 (na lógica do editor: "If Player Number Health $<$ 50", ou "Se Jogador Número Energia $<$ 50").

O terceiro parâmetro que pode estar associado a um personagem é o "Clicked", representando a situação na qual ele é clicado pelo jogador. Neste caso, a condição é mais simples, e não exige parâmetros adicionais.[171]

A segunda parte da regra, referente à ação que resulta da condição descrita na primeira parte, obedece uma lógica semelhante. A diferença sendo que, neste caso, a lógica não descreve uma situação que precisa acontecer, mas sim uma ação que pode ser executada.

Por conta disso, no caso de valores numéricos, ao invés de fornecer parâmetros de comparação - igual ($=$), menor ($<$) ou maior ($>$) -, na regra o parâmetro indica uma operação a ser realizada. Assim, por exemplo, ao invés de testar se uma determinada propriedade é igual a um certo valor, na ação esta propriedade pode receber este valor. Ou então, ao invés de verificar se a propriedade é menor ou maior do que um certo valor, é possível acrescentar ou subtrair um número dela. Além disso, na ação é possível definir um atraso (delay) para a sua execução, uma vez atendida a condição da regra.

[171] Nesta iteração do MissionMaker2 ainda não havia sido implementada a possibilidade de utilizar as regras como parâmetros dentro de outras regras, uma funcionalidade que existe na primeira versão. Esta é uma operação lógica poderosa, pois permite ao designer acessar um nível adicional de abstração - e, consequentemente, de metacriação (esta funcionalidade pode ser parcialmente implementada na iteração atual a partir de regras adicionais, porém de maneira limitada e pouco prática).

Como mencionamos anteriormente, um dos diferenciais mais significativos desta nova versão do MissionMaker é a possibilidade de criar regras com múltiplas condicionais. A imagem abaixo ilustra isso (fig. 51), exemplificando a partir de uma regra que descreve um aspecto do comportamento de um personagem. De uma maneira simplificada, a regra determina que a criatura irá atacar se o jogador entrar em um determinado ambiente carregando um certo item (a poção verde) - isto é, estas duas condições (pegar a poção e adentrar o ambiente) precisam ser cumpridas de forma que a ação (o ataque da criatura) seja ativada. Como veremos, a entrada do jogador no ambiente será detectada através de um volume de ativação.

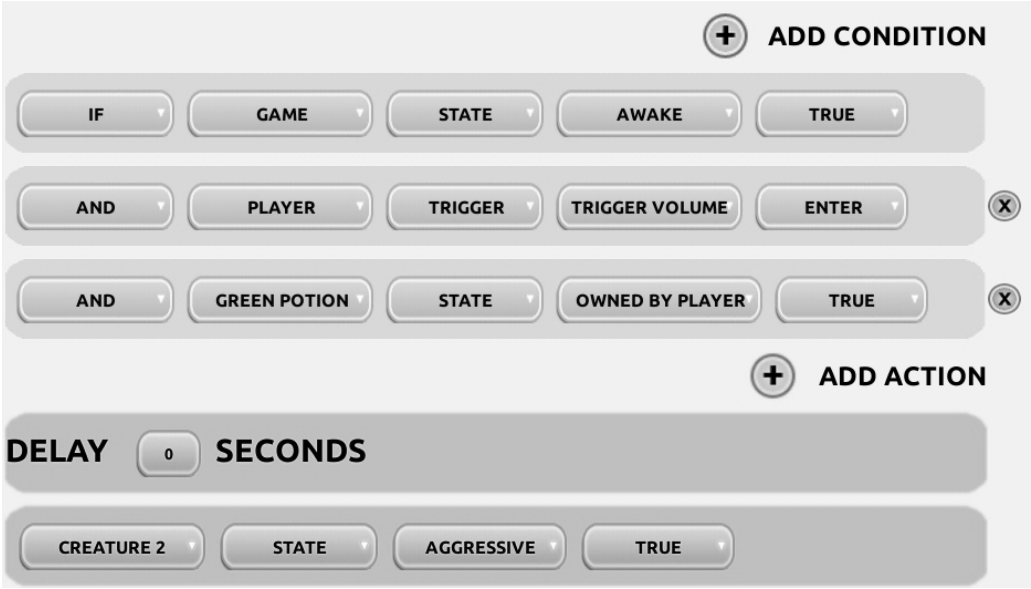


Figura 51 – Regra com múltiplas condicionais

Analisemos agora cada uma das linhas que compõem a regra, começando pelas condições, no topo (inicialmente, iremos ignorar o primeiro operador de cada condição). A primeira condição diz que o jogo deve estar ativo - ou: "Jogo Estado Ativo Verdadeiro" ("Game State Awake True"). A segunda diz que o jogador deve adentrar o volume de ativação, que está localizado logo após a passagem para a localidade em questão - ou: "Jogador Ativar Volume de ativação Entrar" ("Player Trigger Trigger volume Enter"). A terceira e última condição diz que o jogador deve estar de posse da poção verde - ou: "Poção verde Estado De posse do Jogador Verdadeiro" ("Green potion State Owned by Player True").

Agora, considerando os primeiros operadores de cada condição, temos que o primeiro é "If" (ou "Se"), significando que esta primeira condição deve acontecer, e os demais são "And" ("E"), significando que eles também devem acontecer - isto é, as três condições devem acontecer simultaneamente de forma a ativar a ação da regra.

Passando agora para a única ação desta regra, temos que a criatura irá tornar-se agressiva - ou: "Criatura Estado Agressiva Verdadeiro" ("Creature 2 State Aggressive True"). O atraso (delay) é de zero segundos, significando que a ação acontece imediatamente, assim que as condições forem atendidas.[172]

O resultado deste comportamento na prática é ilustrado nas imagens da composição abaixo (fig. 52). No primeiro quadro a criatura se encontra imóvel ou inativa - a poção está no chão, e o jogador está fora da localidade referenciada na regra (trata-se do ambiente escuro, visível ao fundo da imagem). O segundo quadro representa a simulação após o jogador pegar a poção - como esperado, não há mudança no comportamento da criatura.

O último quadro representa a situação após o jogador ter pegado a poção e caminhado para dentro do ambiente escuro (e, portanto, após ter entrado no volume de ativação) - a criatura está agora atacando ele, conforme determina a regra.

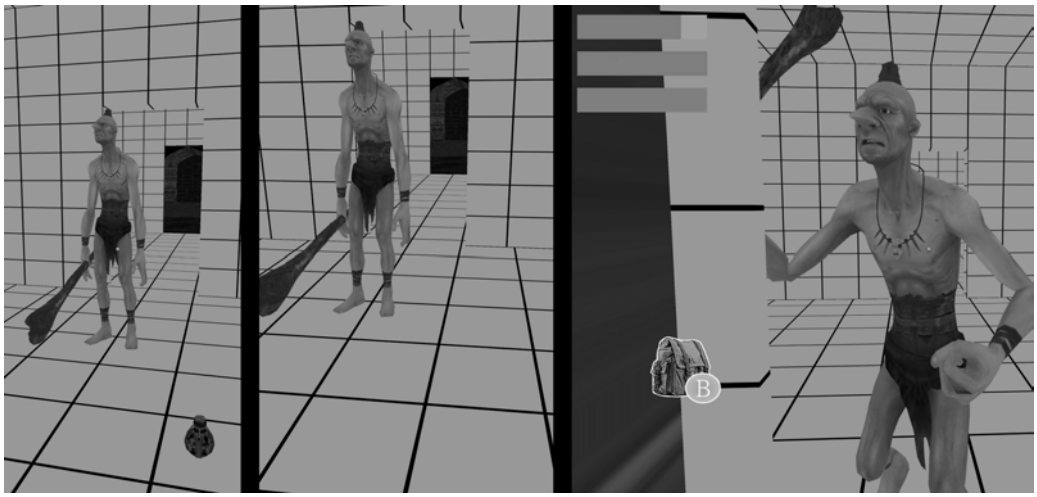


Figura 52 – Comportamento resultante da regra

Além da possibilidade de criar múltiplas condicionais, no MissionMaker2 as regras também podem ter mais de uma ação resultante. Isto é ilustrado na composição a seguir (fig. 53; p. 215). A imagem à esquerda na figura mostra a mesma regra descrita anteriormente, porém com ações adicionais. Nesta nova regra, além da criatura se tornar agressiva, as dimensões dela diminuem em 20% (há três ações, uma para cada eixo do espaço 3D). À direita na composição, é possível ver o resultado da ativação desta nova regra na simulação - a criatura, além de atacar o jogador, agora tem suas dimensões reduzidas.

[172] "Creature 2" é o nome padrão interno para este personagem no MissionMaker2.

Figura 53 – Regra com múltiplas ações: regra (esq.) e resultado

Jogo (interação)

A interação com o jogo no MissionMaker2 é similar em muitos aspectos a aquela da primeira versão, descrita anteriormente. Por exemplo, as setas servem para se deslocar (com o movimento lateral padronizado para o "strafe") e o mouse para controlar a visão do jogador. Existem algumas diferenças pontuais, como o HUD menos intrusivo, permitindo uma visão menos obstruída do espaço virtual.

A mecânica de interação também prioriza e incentiva a ação e o combate, em oposição à exploração. Por exemplo, no início de todo jogo, por um breve momento, o avatar tem os punhos levantados (pronto para o combate). Além disso, a velocidade padrão de deslocamento do avatar é o equivalente a uma corrida (ou caminhada rápida), e o pulo (tecla espaço) é extremamente alto e rápido (nesta nova versão não existe um comando para o avatar se abaixar).

No MissionMaker2 o movimento da perspectiva do jogador através do mouse acontece de maneira automática, sem a necessidade de clicar ou acionar alguma tecla modificadora. Por um lado, isso torna a locomoção no espaço do jogo um pouco mais fluída e intuitiva (após algum treino). Por outro lado, isso dificulta um pouco a interação com objetos e elementos no espaço, uma vez que de forma a posicionar o "alvo" do HUD sobre eles é necessário deslocar toda a perspectiva (ao invés de apenas o cursor de interação, como acontece no primeiro MissionMaker).[173]

[173]Quando o jogador está próximo de um item, ao posicionar o "alvo" sobre ele, é possível manipular (clicar e arrastar), pegar (adicionar ao inventário) ou examinar ele (abre uma janela com as informações).

Comparativo

Em comparação com a primeira versão, o MissionMaker2 segue a tendência de buscar um fluxo de trabalho mais direto e intuitivo, bem como uma interface mais simples e didática. De uma maneira geral, este objetivo tem sido alcançado, embora muitas funcionalidades e aspectos do aplicativo ainda não tenham sido implementados na iteração atual.

Outra diferença significativa em termos das regras no MissionMaker2, além da possibilidade de múltiplas condicionais, é que o processo não segue mais a lógica "invertida" da versão anterior (p. 143). Isto é, enquanto que no primeiro MissionMaker o usuário precisava primeiro selecionar um dos elementos envolvidos na regra antes de poder criá-la, no MissionMaker2 a regra é criada diretamente, e os elementos envolvidos são associados posteriormente, no processo de edição.[174]

Quanto à interface gráfica, o Missionmaker2 tem uma quantidade significativamente menor de elementos, e apresenta um visual mais leve, certamente mais acessível para jovens e crianças. De uma maneira geral, o fluxo de trabalho nesta nova versão deve tornar o processo de criação de jogos mais intuitivo e didático, principalmente nas próximas iterações do aplicativo, à medida em que as funcionalidades e elementos faltantes forem incluídos.

Outra diferença significativa do MissionMaker2 em relação à versão anterior é o fato de que esta nova é baseada no Unity, um engine 3D robusto e rico em funcionalidades, e que oferece compatibilidade com inúmeras plataformas, incluindo na web. Como vimos, a primeira versão do MissionMaker usa um engine proprietário, significando que foi uma criação inteiramente original.

Por um lado, isso certamente limitou a quantidade de funcionalidades que seriam implementadas, uma vez que a equipe precisou dividir tempo e recursos com o desenvolvimento do aplicativo em si. Por outro lado, ao empregar um engine próprio, os desenvolvedores tinham um controle maior sobre os elementos mais fundamentais do aplicativo e dos jogos nele produzidos.

[174] Como vimos, não existe um consenso quanto a qual dos dois processos é de fato "invertido". Além disso, é possível argumentar que este novo processo não difere fundamentalmente do anterior, pois toda nova regra é criada sempre com base em um padrão (ou "default"), incluindo os elementos envolvidos. Isto é, a princípio o usuário do MissionMaker2 também não pode criar regras vazias - a única diferença seria que na versão anterior do aplicativo não era possível criar regras padrão.

3.5.2 Próximas Iterações

A seguir, apresentaremos algumas das investigações e discussões realizadas durante o período em que participei da equipe de pesquisa e desenvolvimento do MissionMaker, e nos seus possíveis impactos nas próximas iterações do MissionMaker2.

Como vimos, o projeto Playing Beowulf tem como propósito principal o ensino da programação de computadores para jovens e crianças, mais especificamente de jogos e narrativas digitais, em um contexto interdisciplinar. Isso inclui a perspectiva do emprego do MissionMaker como uma leitura ou interpretação alternativa a uma obra tradicional. Isto é, o processo de criação de jogos baseados em Beowulf também permite novos olhares sobre os diferentes elementos e aspectos do poema, possivelmente trazendo perspectivas que nunca poderiam ser estabelecidas de outra maneira.

Tratamos anteriormente sobre como o caráter inerentemente sistêmico de uma obra como Beowulf a torna particularmente adequada para ser representada a partir de estratégias procedurais em meio digital. O fato de que o MissionMaker permite ao designer utilizar personagens, objetos e localidades do universo de Beowulf, além de imagens, músicas e gravações de voz, certamente contribui para a representação das cenas e situações do poema. Entretanto, estes tratam-se de elementos estáticos, associados a linguagens e formas expressivas tradicionais.

Um dos tópicos discutidos no planejamento desta nova versão da ferramenta foi a possibilidade de oferecer ao usuário bibliotecas com conteúdo de natureza procedural - isto é, regras e comportamentos, baseados nos temas e situações do poema. Por exemplo, o comportamento de lealdade entre personagens, implementado no jogo de Tracey, uma das participantes da oficina, poderia existir como um recurso procedural, no próprio editor do jogo (p. 141).

Indo além, também foram discutidas formas de interpretar o poema de uma maneira ainda mais fundamental no MissionMaker, integrada à própria lógica e estrutura do engine - e, consequentemente, no fluxo de trabalho do editor. A seguir analisaremos dois casos nos quais este processo poderia ser implementado, investigando como duas estruturas linguísticas proeminentes em Beowulf - a aliteração e os kennings -, poderiam ser representadas no engine do MissionMaker2.[175]

[175] Embora esta pesquisa apresente um caráter interdisciplinar, o nosso foco são as mídias digitais. O emprego de conceitos e terminologias de outras áreas, formas e linguagens - como é o caso das investigações apresentadas a seguir, sobre algumas das estruturas linguísticas empregadas no poema - deve ser considerado neste contexto.

Aliteração

Ao descrever a estrutura de Beowulf, mais especificamente a sua métrica, o tradutor Ary Gonzalez Galvão explica que, "[e]m vez da rima no final de cada verso, usa-se a aliteração como elemento de ligação entre palavras chaves. Numa linha comum do poema, uma ou duas palavras ou sílabas acentuadas na primeira metade do verso sofrem aliteração com uma palavra ou sílaba acentuada na segunda metade." (GALVÃO, 1992:26-27). Esta métrica peculiar dá forma ao poema, não apenas no sentido mais prático do termo, mas também na definição da sua poética, dando o ritmo da leitura, estabelecendo conexões e sugerindo uma coesão ao longo de toda a obra.

Há inúmeras maneiras como esta lógica poderia ser aplicada à própria estrutura do engine do MissionMaker. Apresentaremos abaixo um exemplo - isso será feito em duas etapas: primeiro veremos uma possível implementação para o caso mais simples da rima, e em seguida como ela pode servir como unidade para implementar a métrica particular de Beowulf, baseada na aliteração.

A rima é uma estratégia na linguagem verbal que permite estabelecer conexões entre diferentes partes de um texto ou poema, através da repetição de estruturas similares. Há diferentes maneiras pelas quais esta estratégia pode ser empregada, com diferentes efeitos e funções. A rima pode estabelecer um ritmo ao texto, evidenciar palavras de forma a reforçar um tema, produzir efeitos mais subjetivos, e assim por diante.

Uma possível implementação da rima no Missionmaker seria na forma de uma diretriz operando em um nível acima das regras já existentes, priorizando a ativação de regras que possuírem estruturas ou elementos similares dentro da simulação - desta forma, produzindo "rimas" entre as regras (ou "rimas procedurais").

De forma a ilustrar a aplicação desta diretriz, tomaremos como base o jogo de Tracey, apresentado anteriormente, que se passa no salão de banquetes de Heorot, e inclui diversos guerreiros como NPCs (p. 137). Porém, para este exemplo vamos supor que existe um número maior de regras e comportamentos ativos na simulação - por exemplo, personagens podem ter metas permanentes (buscar comida e bebida), eventos ocasionais (cantar e dançar, ou brigar entre si), reações a situações diversas (atração por aglomerações de outros personagens, repulsão por brigas entre outros personagens), e assim por diante.

Considere agora a aplicação de uma diretriz mais ampla, estabelecendo uma "rima procedural", tal qual sugerida anteriormente. Esta diretriz poderia determinar que a ativação de cada regra específica fosse condicionada a outras regras similares sendo ativadas ao mesmo tempo. Por exemplo, digamos

que em um dado momento alguns guerreiros decidam dançar ao mesmo tempo (enquanto que os demais optam por outras atividades diversas). O sistema de "rima" poderia detectar a sincronia entre os membros daquela atividade e, de forma a evidenciar esta ocorrência, fazer com que os demais NPCs fizessem o mesmo. Isto é, o resultado seria que as atividades não seriam mais realizadas de maneira independente pelos NPCs, mas sim de maneira "rimada". Esta diretriz seria percebida pelo jogador (embora de maneira indireta), na forma de uma ordem (ou estrutura) na simulação, que não existia anteriormente.

Note que, assim como no caso da rima na linguagem verbal, a princípio não haveria uma função poética única ou predeterminada para esta estratégia. Diferentes designers poderiam aplicar este mesmo mecanismo da "rima procedural", mas com propósitos expressivos distintos (ou mesmo para um propósito prático). Por exemplo, um pode utilizar esta diretriz como maneira de tornar os eventos mais evidentes para o jogador, e desta forma simplificando a simulação; enquanto que outro poderia empregar a "rima procedural" com a intenção de expressar algo mais subjetivo, como a ideia da comunhão entre os guerreiros.

Sabemos que os versos aliterativos empregados em Beowulf podem ser considerados como uma espécie de rima, baseada na repetição de sílabas iniciais das palavras, e interna a cada verso individual (entre a primeira e a segunda metade do verso). Assim, o sistema de "rima procedural" descrito acima será agora aproveitado para descrever uma possível implementação para esta métrica particular de Beowulf.

De forma a implementar esta métrica no engine do MissionMaker é necessário primeiro definir uma correspondência dentro da simulação ao que seriam os versos no poema. Para este exemplo, consideremos que o espaço físico do salão de banquetes representa a estrutura do verso - isto é, um dos lados do salão corresponde à primeira parte do verso, e o outro à segunda. Note que esta trata-se de uma escolha arbitrária - a princípio não importa qual seria a modelagem inicial desta estrutura na simulação, desde que o restante da implementação se dê de maneira consistente e coerente com ela.

Agora, acrescentemos à diretriz da rima, definida anteriormente, um critério adicional: uma vez que um determinado conjunto de regras similares for ativado, um conjunto similar deve ser ativado no outro lado do salão. O resultado disso na simulação, considerando que esta modificação fosse aplicada no exemplo anterior, seria algo do tipo: um determinado grupo de personagens começaria a dançar em um lado do salão, e imediatamente o mesmo aconteceria com outro grupo do outro lado do salão - uma espécie de "aliteração procedural".

Assim como no caso da rima, esta correspondência é essencialmente arbitrária, definida de acordo com as escolhas e sensibilidade do designer - há inúmeras formas de modelar e implementar esta tradução. Por exemplo, um modelo alternativo para implementar a aliteração seria considerar a regra em si como uma representação do verso, com a condicional correspondendo à primeira metade, e a ação à segunda. A aliteração neste caso aconteceria entre estes dois elementos - isto é, a regra poderia ser ativada quando houvesse um elemento em comum na condição e na ação, como o objeto ou sujeito das declarações, ou as operações lógicas, por exemplo. O resultado disso na simulação seria um novo comportamento para o sistema, com potenciais implicações poéticas ou expressivas.

A seguir, propomos um outro exercício teórico, desta vez examinando como a forma linguística dos kennings poderia ser implementada (ou traduzida) como um modelo de representação interno no engine do MissionMaker.

Kennings

O kenning consiste em uma estrutura literária, na qual determinados termos são substituídos por uma versão composta, expressando a mesma coisa a partir de outras palavras (p. 191) - por exemplo, "corpo" se torna "câmara-dos-ossos" (em Inglês, "body" - "bone-chamber"). Este dispositivo, ao mesmo tempo em que permite ao autor ampliar e enriquecer o termo original, também possibilita definir uma camada poética adicional, a partir das associações e relações estabelecidas.

Por conta da natureza particular desta estrutura linguística, ela se presta bem para a descrição de sistemas, relacionamentos e hierarquias. Daí surgiu a ideia de adotar o kenning como um modelo de representação no engine do MissionMaker, para a descrição interna de estruturas e elementos de lógica, bem como na própria metáfora de criação no editor.

Considere o exemplo de kenning mencionado acima, substituindo "corpo" pelo termo composto "câmara-dos-ossos". Note que esta substituição não visa acrescentar novas informações ou propriedades ao termo original (afinal, é senso comum que um corpo contém ossos), mas sim de cumprir uma função expressiva. Isso se dá tanto na composição em si, ao evidenciar um elemento específico do corpo - os ossos, possivelmente enfatizando o tema da mortalidade -; quanto na sua eventual contribuição para a métrica dos versos aliterativos.

Suponha agora uma implementação da estrutura do kenning como modelo de representação interna no engine do MissionMaker, atuando como uma estrutura lógica para a descrição dos diferentes elementos de um sistema, e os relacionamentos entre eles. Assim como na sua manifestação na

linguagem verbal, o modelo de representação do kenning no engine do MissionMaker poderia servir tanto para propósitos práticos quanto expressivos.

Um exemplo de um emprego prático seria na descrição das propriedades e "affordances" (comportamentos, ou possibilidades de interação) de objetos e outros elementos da simulação. Por exemplo, suponha que no mesmo sistema contendo a descrição para "corpo" como "câmara-dos-ossos", um martelo seja descrito com o kenning "quebrador-de-ossos". A princípio seria possível implementar a lógica da simulação de forma que esta possa inferir automaticamente qual seria a consequência de atacar um personagem com um martelo - se a arma quebra ossos, o corpo contém ossos e o personagem tem um corpo, conclui-se que o martelo pode ser usado para ferir o personagem.[176]

A lógica do kenning também poderia ser empregada no engine com propósitos poéticos, explorando o princípio da inferência. Por exemplo, em Beowulf a "espada" é referenciada como "luz-da-batalha" ("sword" - "battle-light"), por conta da luz refletia nas lâminas; e "morte", como vimos, é substituída pela composição "sono-da-espada". Com base nestes dois kennings, seria possível inferir a ideia de que a morte é um "sono iluminado", por exemplo.

Esta ideia poderia ser empregada na construção de alguma situação no espaço virtual (por exemplo, o personagem dormindo diante de um fundo branco), ou como um dos elementos do diálogo de um personagem, e assim por diante.[177]

Nos dois casos, independentemente do propósito ser predominantemente prático ou expressivo, o mecanismo interno que lida com os kennings é o mesmo. Isto é, do ponto de vista do algoritmo, não há diferença prática entre inferir que a morte pode ser simbolizada por um "sono iluminado" e deduzir que um martelo pode ser usado para ferir um personagem. Assim como acontece com qualquer outro meio ou linguagem, os significados sendo representados ou expressados em uma determinada produção (ou em algum elemento ou parte dela) dependem principalmente das escolhas e da intenção de quem a criou ou idealizou.

Os exercícios teóricos apresentados aqui ilustram possíveis rumos para as próximas iterações do MissionMaker2. A intenção é de trazer ainda mais elementos e aspectos poéticos e narrativos de

[176] Uma infinidade de outras regras e estruturas seriam necessárias para que uma situação como esta pudesse se realizar. Este exemplos ilustram apenas parte do sistema. Por exemplo, neste caso particular, o sistema também precisaria conter a informação de que todo personagem tem um corpo, alem de um modelo interno para o termo "quebrar", sua função no contexto da simulação, e assim por diante.

[177] Note que o potencial expressivo e de construção de significado do sistema sugerido aqui cresce exponencialmente, uma vez que a cada nova inferência surgem novos kennings.

Beowulf para dentro do engine, de forma que eles se reflitam não apenas nos jogos criados com esta ferramenta, mas também no próprio editor. Desta maneira, a própria experiência de design e criação do jogo podem servir como uma forma alternativa de leitura e fruição ativa desta obra.

Convém reiterar que, como os estudos práticos apresentados neste capítulo se deram no contexto do projeto Playing Beowulf, o foco da análise naturalmente foi na tradução de elementos e aspectos desta obra de referência em particular para a linguagem do jogo digital.

Porém, as mesmas ideias apresentadas aqui também poderiam ser identificadas e aplicadas a produções realizadas sem um material de referência, ou a outros tipos de criação, em outras áreas e linguagens. Neste sentido, estes exercícios também servem para ilustrar e informar a investigação central desta pesquisa sobre o emprego expressivo das estratégias procedurais de criação de uma maneira mais ampla.

4 Pelo Uso Expressivo do Meio Digital

> *Na arte a lei geral é a regra individual da obra a ser feita*. [A] *única lei da arte é o critério do êxito*. [Em arte,] a obra *triunfa porque triunfa*: triunfa porque resulta tal como ela própria queria ser [...]; desejada, na sua realidade, pelo autor, mas na sua **interna coerência**, por si mesma.
>
> (PAREYSON, 1989:139)

O processo criativo não é algo rígido - o artista ou criativo tem liberdade para criar, definindo ele mesmo as regras que devem ser obedecidas na produção do seu trabalho. Além disso, no embate com o meio, a ideia ou projeto de criação se transforma, se adapta, amadurece e se desenvolve, considerando as características e limitações técnicas do suporte, a natureza da linguagem, e assim por diante. No contexto da expressividade procedural, isso significa que o processo criativo não se resume a uma mera tradução de uma ideia "pronta" para a memória do computador na forma de uma série de instruções.[178]

Entretanto, se por um lado não existem limites para a criatividade e a expressividade, uma vez definida a proposta de uma determinada produção, esta se torna a regra da sua própria composição. O filósofo Luigi Pareyson descreve obras de arte como "organismos vivendo de vida própria e dotados de legalidade interna" (PAREYSON, 1989:33). Esta legalidade interna pode se transformar ao longo do processo criativo, mas ela sempre estará presente.[179]

Isso é válido mesmo nos casos em que a poética em questão é a própria ausência de uma consistência aparente, como acontece no Dadaísmo, por exemplo, ou quando a obra envolve elementos aleatórios ou que subvertem sua própria lógica interna. Na Glitch Art, por exemplo, a poética da obra se baseia justamente nos erros de execução do código.

Além disso, o autor deve ter consciência dos mecanismos e características dos meios e linguagens empregados para a realização e materialização da sua proposta poética ou criativa, de forma a garantir a coerência e consistência no seu uso.

[178] Epígrafe: Grifo em negrito adicionado (itálicos do texto original).

[179] A "legalidade interna" da qual Pareyson fala não existe de fato na obra em si, mas sim na forma de uma estratégia criativa na mente de quem cria. Em uma escultura, por exemplo, a expressividade não reside no material, mas sim no projeto de criação do escultor. Mesmo que a lei de formação de uma obra ou produção possa ser deduzida a partir da sua estrutura, a referência original e definitiva é sempre o criador. Como vimos, mesmo no caso da expressividade procedural, o projeto de criação não reside no algoritmo em si (embora neste caso exista a possibilidade de haver correspondências entre a lógica do sistema imaginado pelo criador e a daquele implementado na máquina).

Neste capitulo, resgataremos os principais temas e reflexões que resultaram das investigações, elaborações teóricas e estudos práticos realizados ao longo desta tese, expandindo a discussão para o contexto mais amplo da expressividade procedural, e do uso expressivo do meio digital.

Nas próximas seções, aprofundaremos sobre a questão da consistência (e coerência) no processo criativo, com ênfase no equilíbrio entre o emprego de estratégias procedurais de criação, e aquelas tradicionais (estáticas e lineares). Isto será feito a partir de três temas: (1) **abertura local e linearidade global** (p. 227); (2) **escolha de escopo** (p. 235); e (3) **coerência na mecânica de jogo** (p. 241).[180]

Para estas elaborações, serão analisados alguns exemplos e estudos de caso em diferentes áreas, incluindo o drama interativo Façade (Mateas e Stern, 2005), visto anteriormente, o sistema gerador de fábulas Tale-Spin (James Meehan, 1976), a "psicóloga virtual" Eliza/Doctor (Joseph Weizenbaum) e o gênero dos adventures (ou jogos de aventura), entre outros.

[180]Embora os conceitos de consistência e coerência tenham diferentes definições (de maneira simplificada, o primeiro enfatiza regularidade, e o segundo a lógica), nesta pesquisa eles são empregados principalmente em referência ao significado mais amplo compartilhado entre ambos.

4.1 Abertura Local e Linearidade Global

De forma a examinar este primeiro tema, começaremos resgatando o drama interativo Façade, apresentado no segundo capítulo. Como vimos, naquele caso a expressividade procedural se dá em conjunto com o uso de estratégias narrativas tradicionais, resultando em uma espécie de forma híbrida (p. 43).

Façade é dotado de um módulo de "gerenciamento de drama" (ou "*drama manager*"), que atua de maneira a garantir que, durante cada sessão de interação, a estrutura global da narrativa resultante obedeça a um "arco tensivo aristotélico bem formado" (cada sessão geralmente dura cerca de vinte minutos). Isto é, a narrativa é sempre direcionada de forma a atender uma certa sequência de etapas: um incidente, o crescimento da tensão, uma crise, o clímax e um desfecho ou solução (MATEAS; STERN, 2005:3). O sistema se baseia em uma biblioteca de eventos ou temas predeterminados, denominados "batidas" (ou "beats").

Estas são as 27 batidas utilizadas em Façade, agrupadas por tópicos (Idem:6):[181]

> **Recebendo a visita:**
> Jogador chega, Trip cumprimenta o jogador, Jogador entra e Trip chama Grace, Grace cumprimenta o jogador;
>
> **Tópicos de conversa e situações:**
> Discussão sobre a decoração, Explicação sobre o aniversário de namoro, Discussão sobre a viagem para a Itália, Discussão sobre fazer as bebidas, Telefonema dos pais, Transição para tensão, Trip reencena pedido de casamento, Crise, Pós-crise, Jogo da terapia, Revelações se intensificam, Revelações;
>
> **Situações envolvendo pessoas indo ou voltado da cozinha:**
> Grace vai até a cozinha incomodada, Jogador segue Grace até a cozinha, Grace volta da cozinha, Trip vai até a cozinha incomodado, Jogador segue Trip até a cozinha, Trip volta da cozinha;
>
> **Possíveis finais:**
> Final sem revelações, Final com revelações próprias apenas, Final com revelações de relacionamento apenas, Final em que ambos não estão completamente autoconscientes, Final em que ambos estão autoconscientes.

[181] O agrupamento das batidas por tópicos (assinalados em negrito) não existe no texto original, e foi feito de forma a facilitar a análise. Citação: tradução nossa - texto original: "PlayerArrives, TripGreetsPlayer, PlayerEntersTripGetsGrace, GraceGreetsPlayer, ArgueOverRedecorating, ExplainDatingAnniversary, ArgueOverItalyVacation, FixDrinksArgument, PhoneCallFromParents, TransitionToTension2, GraceStormsToKitchen, PlayerFollowsGraceToKitchen, GraceReturnsFromKitchen, TripStormsToKitchen, PlayerFollowsTripToKitchen, TripReturnsFromKitchen, TripReenactsProposal, BlowupCrisis, PostCrisis, TherapyGame, RevelationsBuildup, Revelations, EndingNoRevelations, EndingSelfRevelationsOnly, EndingRelationshipRevelationsOnly, EndingBothNotFullySelfAware, EndingBothSelfAware" (a ausência de espaços e o uso de "caixa-alta" para indicar o início de novas palavras é uma convenção comum para a nomeação de variáveis na programação - incluímos espaços na tradução de forma a facilitar a leitura)

Cada batida se refere a uma situação particular que pode ocorrer durante uma sessão de jogo em Façade. Algumas delas se referem a contextos específicos, como é o caso das situações relacionadas à chegada do jogador, ou às situações relativas a pessoas indo ou voltando da cozinha. A maior parte das batidas indica potenciais tópicos de conversa ou situações específicas, como uma discussão em torno da decoração do apartamento, algum tópico relacionado à viagem do casal à Itália ou a ocorrência de um telefonema. Há também batidas específicas para descrever cada um dos possíveis desfechos que a narrativa pode ter, de acordo com determinados fatores (como a ocorrência ou não de determinadas revelações, por exemplo).

Uma batida é composta de uma série de acontecimentos e possíveis desfechos, de acordo com a ocorrência ou não de determinados eventos e outros fatores. Por um lado, há uma relativa flexibilidade na maneira como estes elementos se relacionam entre si dentro da batida. Entretanto, do ponto de vista da narrativa global, há pouca variabilidade, uma vez que a agência do jogador se encontra limitada a aquele conjunto de possibilidades.

Embora exista uma certa variação na ocorrência ou não das batidas em si, bem como na ordem em que elas acontecem, este mecanismo age mais como uma espécie de "ruído" sutil, atuando sobre um sistema de bifurcações narrativas claramente definidas e determinadas.

Assim, todas as situações, eventos e comportamentos que podem acontecer em Façade se encontram descritos de maneira explícita no sistema, incluindo todos os temas de diálogo e possíveis bifurcações. Não é por acaso que o termo "batida" foi adaptado do Teatro ou Cinema, formas artísticas tradicionais, de natureza essencialmente linear e estática.

Mesmo a linguagem interna na qual os comportamentos e personalidades dos personagens são descritas em Façade (p. 46), já prevê em sua estrutura uma série de "metas" (ou *"goals"*), de forma a atender a cada uma das situações predeterminadas. Por exemplo, dentro da coleção de comportamentos à disposição dos personagens há um que descreve todas as ações necessárias para abrir a porta quando a visita chega (MATEAS; STERN, 2004).[182]

[182]No sistema de interpretação de linguagem natural, que atua em um nível mais fundamental da simulação, esta predeterminação parece não ocorrer. Este sistema é utilizado para analisar o texto digitado pelo jogador, e que será "dito" pelo protagonista no contexto do jogo. O sistema determina o significado das frases digitadas pelo jogador de acordo com uma lista de palavras-chave, bem como o contexto no qual elas são "ditas". Neste caso, a princípio o sistema não se encontra estruturalmente submetido aos mecanismos do gerenciador de drama (embora a informação sobre o contexto venha diretamente dele).

Artifícios

Assim, embora em Façade o comportamento dos personagens seja baseado em um sistema dinâmico de Inteligência Artificial, há um outro sistema atuando por cima deste, que modula o seu funcionamento de acordo com estruturas narrativas predeterminadas. Isto é, tratam-se de dois tipos de abordagem fundamentalmente distintas, e em alguns aspectos até mesmo contraditórias, atuando simultaneamente.

Esta inconsistência é reconhecida pelo criadores de Façade, e foi considerada no próprio design do projeto. São empregados uma série de artifícios de forma a encobrir os momentos em que o gerenciador de drama interfere diretamente no enredo, ou então para desviar a atenção do interator aos elementos ou aspectos que denunciam o caráter estruturado, estático e linear da experiência.

De fato, grande parte do sistema que governa Façade é dedicado a compensar por estes conflitos entre a abertura - permitida pelo emprego de mecanismos e estratégias procedurais de criação - e o controle - o sistema que garante uma determinada estrutura na narrativa global.

Por exemplo, Mateas explica que "as escolhas no nível da estória em Façade tem como intenção não se parecerem com pontos óbvios de bifurcação" (MATEAS; STERN, 2005:3), sugerindo que na prática estas bifurcações no enredo existem, apenas não de maneira explícita (isto pode ser identificado, por exemplo, no momento em que uma briga começa entre Trip e Grace, ou quando um certo tópico é introduzido na conversa). Porém, o sistema de gerenciamento de drama garante que esta mudança no estado do enredo não é informada diretamente para o jogador, fazendo **parecer** que a sequência de eventos se dá de uma maneira mais orgânica e aberta do que de fato é.[183]

Outro exemplo de um artifício empregado para encobrir estas limitações presentes no sistema de Façade é a caracterização dos personagens Trip e Grace como pessoas egocêntricas. Com isso, justifica-se o fato do casal ter um universo tão limitado de tópicos de conversação, sempre girando em torno deles mesmos e do seu relacionamento em crise.

Se o jogador propõe um tópico não previsto pelo sistema, ou mesmo um sutil desvio dentro de um tópico existente, os personagens muitas vezes ignoram o que foi dito, ou respondem com uma interjeição genérica de confusão ou desinteresse. Em seguida, simplesmente dão sequência ao assunto sobre o qual estavam conversando, ou introduzem um tema novo, sempre dentre aqueles que atendem à narrativa naquele dado momento. Trata-se de um comportamento frustrante, principalmente

[183] Citação: tradução nossa - texto original: "the story-level choices in Façade are intended to not feel like obvious branch points."

para um jogador que já conhece este artifício. Por outro lado, esta é uma solução relativamente elegante para justificar as limitações do sistema (este tópico será retomado mais adiante, ao tratar sobre a questão do escopo; p. 235).[184]

Entretanto, mesmo com o emprego destes artifícios, Façade frequentemente revela inconsistências inerentes ao seu sistema. Por exemplo, a tolerância de Trip e Grace a comportamentos inapropriados da visita - como ofensas e investidas sexuais - é bastante alto, provavelmente como uma tentativa de desestimular este tipo de abordagem, porém a ausência de reações a estas situações torna os personagens inverossímeis.

Também são comuns momentos em que as contribuições do interator são ignoradas mesmo quando são diretamente relacionadas ao tópico ou situação atual. Por exemplo, quando Trip oferece uma bebida ao protagonista, ele não reconhece uma diversidade de opções bastante comuns.

Drama Interativo

Façade é descrito pelos seus criadores como "um protótipo de **drama** interativo, um novo gênero de entretenimento com ênfase em personagens e **estória**". Isto é, embora a proposta seja de uma experiência interativa, esta interação deve acontecer dentro do contexto de uma **estória**, mais especificamente no gênero do **drama**. Na prática, isso significa que, embora o jogador tenha agência para fazer e falar o que quiser, a reação dos personagens Trip e Grace é sempre modulada pelo gerenciador de drama.[185]

Esta abordagem se reflete na linguagem empregada pelos criadores em seu discurso em torno de Façade, em entrevistas e nos artigos em que descrevem os diferentes sistemas que compõem a simulação. É comum o uso de termos como "**enredo**", e da ideia de uma "**trama** gratificante e bem formada" ou de "**narrativas** coerentes, focadas e com um bom andamento" (MATEAS; STERN, 2005).[186]

[184] Como veremos, os adventures clássicos adotam uma estratégia diferente para lidar com esta questão, limitando as falas do jogador a uma lista de opções predeterminadas. Para cada opção há uma resposta ou reação correspondente no sistema. Neste aspecto, Façade funciona de maneira similar, uma vez que as possíveis bifurcações e desfechos da narrativa são todos criados de antemão - há apenas uma maior **ilusão** de liberdade.

[185] Citação: grifo nosso; tradução nossa - texto original: "Façade is a prototype of interactive drama, a new genre of character and story-intensive interactive entertainment." - Texto obtido no site do projeto - URL: `www.interactivestory.net/faq` (a data informada nesta página é: 2005-2006; sem numeração de página)

[186] Citação: grifo nosso; tradução nossa - texto original para as citações, respectivamente: "**storyline**", "satisfying, well-formed **plot**" e "coherent, focused, well-paced **narratives**". (na referência original, o primeiro termo citado aparece em múltiplas páginas; as outras duas citações aparecem nas páginas 3 e 4, respectivamente)

Segundo Mateas, o modo de interação mais adequado para Façade seria uma técnica similar à do teatro de improviso (HARGER, 2006). Porém, embora este paralelo seja coerente, neste caso ele não se aplica de maneira plena.

No teatro de improviso, os atores são encorajados a aceitar as premissas e situações colocadas pelos demais participantes da cena, de forma a permitir que elas se desenvolvam. Assim, por exemplo, se um ator sugere que começou a chover, os outros não devem negar ou ignorar esta premissa, mas sim aceitar e incorporar ela à cena.

Quando Mateas cita esta técnica no contexto de Façade, ele quer dizer que, se o jogador aceitar a premissa global da experiência, prestar atenção no que está se passando e contribuir de acordo com as deixas dadas pelos personagens, ele será recompensado com uma experiência de interação mais completa, coerente e plena, e que atende ao **arco dramático pretendido**.

Entretanto, esta sugestão de aplicar a técnica do improviso em Façade é possível apenas de maneira parcial. No improviso, não basta considerar as contribuições anteriores - os participantes também são encorajados a trazer novos elementos à cena. Isto é, além de reconhecer a premissa do outro participante, o ator deve acrescentar a ela novas informações ou acontecimentos, de forma a contribuir para o desenvolvimento da cena. Entretanto, o sistema de Façade não suporta este tipo de interação, uma vez que, como foi visto, ele reconhece apenas situações previstas de antemão pelos seus criadores. Isto é, as contribuições do jogador só podem acontecer dentro das possibilidades previstas para cada situação.

Por exemplo, quando Grace questiona o conhecimento de Trip sobre bebidas (uma situação que faz parte de uma das batidas na lista citada anteriormente), a contribuição do jogador praticamente se limita a escolher do lado de quem vai ficar na discussão. Se o jogador negar a bebida de Trip, Grace fica satisfeita, pois significa que o protagonista apoia sua posição; se aceitar, Trip se sente vitorioso e a empatia de Grace com relação ao protagonista diminui.

Isto é, a contribuição do jogador deve se limitar aos ocasionais momentos em que uma bifurcação é prevista no enredo. Não seria possível, por exemplo, optar por acalmar os ânimos ensinando uma receita nova de bebida para o casal. Embora esta contribuição fosse perfeitamente compatível com a premissa da situação, ela não está entre aquelas previstas pelo sistema.

É possível encontrar inúmeros relatos e registros em vídeo na internet mostrando situações interessantes ou inusitadas de jogadores interagindo com Façade. Entretanto, na maior parte dos casos

estes registros são interessantes principalmente por conta da performance do usuário, e não do comportamento dos outros personagens. Por exemplo, em um destes vídeos o jogador decide entrar no apartamento de Trip e Grace pedindo socorro e dizendo que tinha acabado de ser baleado. A reação dos personagens é interessante, e até mesmo cômica, porém isso se deve justamente à dissonância entre a situação criada pelo interator e a incapacidade do sistema do jogo em interpretar e reagir a ela. Trip e Grace são incapazes de lidar com eventos não previstos, e portanto nunca poderão apresentar reações verdadeiramente inesperadas.

Em suma, Façade precisa limitar a agência do jogador nas interações locais (isto é, naquelas de curto prazo, entre os personagens de IA, e com o jogador) de forma a viabilizar a realização de um arco narrativo satisfatório e coerente. Isso é feito através do gerenciador de drama, que age de maneira a compensar por esta diferença - isto é, a intensidade procedural é alta nas interações locais, porém baixa na narrativa global, cuja estrutura fundamental é de natureza predominantemente tradicional e linear.

Um Gerador de Fábulas

Examinemos agora o sistema de geração de estórias Tale-Spin (James Meehan, 1976). Assim com Façade, este sistema também adota uma abordagem híbrida, integrando a simulação do comportamento de personagens virtuais com a noção de um gerenciador de narrativa. Entretanto, no caso de Tale-Spin, a simulação não se dá em tempo real, não há abertura para a interação do receptor e as estórias são geradas na forma de textos curtos.

Segue um exemplo de estória gerada pelo sistema:[187]

> Era uma vez a Formiga George vivia perto de um pedaço de terra. Havia um ninho em um freixo. O Pássaro Wilma vivia no ninho. Havia um pouco de água no rio. Wilma sabia que a água estava no rio. George sabia que a água estava no rio. Um dia Wilma estava com bastante sede. Wilma queria ficar próxima de um pouco de água. Wilma voou do seu ninho cruzando um prado através de um vale até o rio. Wilma bebeu a água. Wilma não estava com sede.
>
> (1976[188] apud WARDRIP-FRUIN, 2006:224)[189]

[187] Nesta tradução, feita por nós, procuramos preservar o tom do texto original, incluindo expressões e formas equivalentes a aquelas do Inglês, bem como determinadas estruturas incomuns encontradas.

[188] MEEHAN, James Richard. 1976. **The metanovel: Writing stories by computer**. Tese de doutorado, Yale University.

[189] Citação: tradução nossa - texto original: "Once upon a time George Ant lived near a patch of ground. There was a nest in an ash tree. Wilma Bird lived in the nest. There was some water in a river. Wilma knew that the water was

Nas estórias geradas por Tale-Spin, o índice de granularidade é relativamente alto, em comparação com outros projetos similares. A variação neste caso se dá no nível das palavras que compõem cada frase. Como comparação, no livro Cent mille milliards de poèmes (Raymond Queneau, 1961), a unidade mínima no sistema são as frases, e nos livros da série Choose Your Own Adventure, esta unidade mínima é a página.[190]

Por um lado, devido a esta alta granularidade Tale-Spin permite a geração de estórias relativamente complexas e variadas. Porém, isso também resulta em estruturas incomuns, como pode ser verificado no trecho reproduzido acima. Por exemplo, note a curiosa menção de que os personagens "[sabiam] que a água estava no rio", ou o tempo verbal empregado nas duas frases finais "Wilma bebeu a água" e "Wilma não estava com sede".

O sistema Tale-Spin possui diferentes modos de execução. Wardrip-Fruin explica que, em um deles, a simulação é executada de forma a garantir que a narrativa gerada seja "mais acessível":

> Meehan [, o criador de Tale-Spin,] não argumenta que simplesmente o ato de simular o mundo real sempre produz **estórias interessantes**. Tale-Spin inclui dois modos baseados na experimentação com o simulador, mas inclui também um modo narrativo que fixa o conteúdo do mundo de antemão, de forma a garantir **uma boa estória**. (WARDRIP-FRUIN, 2006:239)[191]

De fato, assim como acontece em Façade, há uma série de convenções e estratégias tradicionais de narrativa sendo empregadas no sistema. Isto é, a preocupação em produzir uma "estória interessante" é implementada no próprio sistema. Uma "boa estória" seria aquela que atende às expectativas que se aplicariam a uma narrativa tradicional - um certo tipo de relacionamento entre personagens, uma determinada estrutura ou lógica na sequência de acontecimentos, uma situação ou cena particularmente significativa e um final surpreendente ou dramático, por exemplo.

---

in the river. George knew that the water was in the river. One day Wilma was very thirsty. Wilma wanted to get near some water. Wilma flew from her nest across a meadow through a valley to the river. Wilma drank the water. Wilma was not thirsty." (segundo Wardrip-Fruin, na tese de Meehan a estória toda é apresentada em maiúsculas)

[190] O livro Cent mille... trata-se de uma obra de Arte Permutacional (ou combinatória) na qual as frases são dispostas em uma série de tiras que podem ser combinadas pelo leitor em uma infinidade de poemas. A série Choose Your Own Adventure, mencionada anteriormente, consiste em livros contendo bifurcações em determinados pontos da narrativa. A cada bifurcação o leitor é apresentado a algumas opções, e de acordo com sua escolha deve continuar a leitura na página indicada.

[191] Citação: grifo nosso; tradução nossa - texto original: "Meehan does not argue that simply simulating the real world always creates **interesting stories**. Tale-Spin does include two modes based on experimentation with the simulator, [b]ut it also includes a mode of storytelling that fixes the contents of the world in advance, in order to guarantee **a good story**."

Em Tale-Spin há ainda uma camada adicional de controle que aplica no enredo a estrutura e a estética do gênero da fábula. Isso se reflete, por exemplo, nas expressões e na linguagem utilizada na geração do texto, como a clássica introdução "Era uma vez", o uso de frases curtas e simples como "O Pássaro Wilma vivia no ninho" e assim por diante.[192]

Espen J. Aarseth, pesquisador na área de jogos digitais e literatura eletrônica, faz uma crítica a Tale-Spin argumentando que os "contadores de estória maquínicos" (ou "machine narrators") não deveriam ser forçados a simular contadores de história humanos.

> De forma a realizar uma literatura gerada por computador que seja interessante e significativa, é necessário descartar as poéticas da literatura narrativa e usar o potencial do computador para a combinação e para a **simulação de mundos**, de forma a desenvolver novos gêneros que podem ser valorizados e empregados em seus próprios termos. (1997[193] apud WARDRIP-FRUIN, 2006:274)[194]

A posição de Aarseth é radical, mas reflete o argumento apresentado nesta pesquisa, de que o emprego pleno do potencial expressivo das mídias digitais se dá a partir dos mecanismos próprios deste meio, através do aspecto procedural (em oposição ao uso de formas expressivas tradicionais).

[192] Também é possível notar o imaginário associado ao gênero da fábula nos personagens, que consistem em animais antropomorfizados, demonstrando características e atitudes humanas.

[193] AARSETH, Espen J. 1997. **Cybertext: Perspectives on ergodic literature**. Johns Hopkins University Press.

[194] Citação: tradução nossa - texto original: "To achieve interesting and worthwhile computer-generated literature, it is necessary to dispose of the poetics of narrative literature and to use the computer's potential for combination and world simulation in order to develop new genres that can be valued and used on their own terms." (na tradução, a estrutura da frase foi ligeiramente alterada para clareza - a palavra "worthwhile" foi traduzida como "significativa"; uma opção mais informal para esta tradução seria "que vale a pena").

4.2 Escolha de Escopo

A consistência e a coerência interna de uma produção criativa ou artística tem menos a ver com critérios como completude, complexidade ou realismo, e mais com a maneira como os seus diferentes aspectos e elementos são compostos e integrados a partir de uma determinada proposta ou projeto. Como veremos nesta seção, parte deste processo envolve a escolha adequada do escopo dentro do qual a produção irá existir ou ser realizada.

Em Façade, um dos possíveis tópicos de conversação previstos no sistema - isto é, que os NPCs podem introduzir ou reconhecer durante o jogo - é sobre a decoração que a personagem Grace fez para a sala de estar do seu apartamento. Porém, esta discussão é secundária, no sentido em que ela serve principalmente como um meio pelo qual o tema central da narrativa - a crise conjugal de Trip e Grace - pode se desenvolver.

Por conta disso, no jogo não existem situações ou diálogos particularmente elaborados aprofundando sobre este tópico, e tampouco há muitos detalhes na representação de elementos físicos presentes no ambiente virtual, como as esculturas, quadros e outros objetos de decoração.

Se o tema central de Façade fosse as preocupações de Grace com relação às suas habilidades em decoração, o apartamento do casal provavelmente seria representado no jogo de uma maneira mais detalhada, e haveriam mais ações e tópicos de conversação disponíveis relacionadas a estes elementos. Talvez o interator pudesse reposicionar as almofadas do sofá, examinar a textura do tapete, ter um diálogo sobre a luminária e assim por diante.

Entretanto, como este não é o caso, a agência limitada (ou ausente) do interator com relação a estes elementos não tem um impacto relevante na experiência pretendida. Isto é, a coerência interna do sistema é preservada a partir da escolha de um escopo que está de acordo com a proposta de criação, bem como a intenção expressiva dos seus criadores.

Eliza/Doctor

A "psicóloga virtual" Eliza/Doctor foi um programa pioneiro na área da Inteligência Artificial, criado por Joseph Weizenbaum (MIT) em meados dos anos 60. Nele, o interator faz o papel de um paciente, dialogando com a "psicóloga" através de uma interface de texto.[195]

Um dos aspectos nos quais a escolha de escopo se manifesta neste projeto é no modelo de representação da simulação. Ao contrário de Façade, em Eliza não existe uma representação do espaço físico. Tudo que o interator vê é o texto representando o diálogo com Eliza. Isto é, não é possível observar o ambiente ao redor, levantar-se e caminhar pelo consultório ou pegar um objeto, por exemplo. Embora a existência de uma suposta fisicalidade no universo da simulação seja inferida pelo contexto, ela não é levada em consideração no diálogo com a doutora virtual, que se concentra em tópicos de ordem psicológica. No contexto deste programa, esta escolha de escopo faz sentido.

Outra maneira como a definição de escopo se manifesta em Eliza é na escolha da abordagem específica de comunicação empregada pela doutora virtual. Ela geralmente baseia as suas perguntas e observações diretamente no que foi dito pelo paciente. Assim, por exemplo, se o interator diz "Eu briguei com meu irmão", uma pergunta típica de Eliza seria "E como você se sente por ter brigado com seu irmão?". Esta técnica é baseada em uma abordagem caracterizada pela ênfase no paciente, conhecida como "psicoterapia rogeriana". Esta caracterização da doutora permite compensar o fato de que o programa tem uma capacidade limitada de interpretar o texto digitado pelo usuário, além de uma base de conhecimento bastante reduzida.

Se o interator digita algo que o sistema não é capaz de interpretar, a doutora pode simplesmente responder "Interessante, fale mais sobre isso". Ela também pode fazer uma pergunta genérica como "Porque você diz isso?", ou mesmo devolver o que foi dito na forma de uma pergunta. Todos estes comportamentos são relativamente coerentes se considerados dentro do escopo estabelecido pelos designers do sistema.

A reação das pessoas a Eliza na época do seu surgimento foi de deslumbramento. Mesmo quem conhecia os mecanismos e artifícios empregados pelo sistema achava interessante e divertido conversar com a doutora virtual. É notável que, ao longo das décadas que se seguiram, mesmo com o surgimento de sistemas mais avançados de reconhecimento de linguagem natural, integrados a técnicas avançadas de IA e volumosas bases de conhecimento, Eliza ainda preserva grande parte do seu apelo.

[195] "ELIZA", com a grafia em maiúsculas, é o nome do programa principal. Existem diferentes roteiros (ou "scripts") que podem ser executados, sendo "DOCTOR" o mais popular deles. Neste texto o nome "Eliza" será empregado como abreviação para Eliza/Doctor, e sem a grafia em maiúsculas.

Barreiras Invisíveis (ou Artificiais)

Anteriormente apresentamos uma analogia entre a lógica de um sistema de IA e um engine de física, ilustrando a similaridade no emprego das estratégias procedurais de criação nestes dois casos (p. 47). A seguir aplicaremos uma analogia similar para comparar a **camada de controle** que modula o sistema de IA em Façade (isto é, o "gerenciador de drama" - p. 227) com as **barreiras físicas** que geralmente são empregadas para limitar o deslocamento de personagens e outros elementos em um espaço virtual.

Estas barreiras estão presentes em praticamente todos os jogos e simulações envolvendo espaços físicos virtuais, e muitas vezes são invisíveis. Geralmente elas se encontram "nos cantos" da simulação, distantes de localidades ou elementos relevantes para o jogo. As barreiras podem existir ao longo do perímetro de uma ilha, ou delimitando a área em torno de uma pista em um simulador de corrida, por exemplo. A imagem abaixo (fig. 54) representa uma destas barreiras - neste caso, trata-se de uma estrada que simplesmente acaba, no "fim do mundo".[196]

Figura 54 – Série "the end of the virtual world" (Robert Overweg, 2010)

Muitas vezes as barreiras invisíveis são integradas diretamente à jogabilidade. Por exemplo, elas podem cumprir uma função narrativa, forçando o interator por um trajeto específico, levando-o a um determinado local ou situação que permita cumprir um enredo predefinido.

[196] Robert Overweg, o autor da imagem citada, se denomina como um "fotógrafo do mundo virtual". Fonte da imagem: site do artista (original colorida) - URL: `www.shotbyrobert.com`

Também existem barreiras artificiais que não são invisíveis, mas cumprem uma função similar na simulação, como portas trancadas ou paredes altas demais. Estas também limitam a o deslocamento dentro do mundo virtual, porém de maneira menos intrusiva do que as "barreiras invisíveis", pois os limites tendem a ser justificados de uma maneira coerente dentro da narrativa ou universo do jogo. Uma barreira artificial comum é a escolha de um cenário naturalmente limitado, como o interior de uma caverna, por exemplo.[197]

Limitações são inevitáveis em qualquer simulação que pretende representar o mundo real (ou aspectos dele), seja por questões técnicas, como o limite de memória ou de processamento da máquina, ou pela própria impossibilidade de considerar e modelar todos os possíveis elementos e aspectos envolvidos. Isso se aplica tanto à simulação do comportamento de personagens, como é o caso de Façade, quanto às barreiras físicas em uma simulação.

Mesmo em Façade, embora o jogador tenha uma certa agência no espaço virtual que representa o apartamento de Trip e Grace, ela é limitada à manipulação de alguns objetos, determinadas ações com os personagens e movimentar-se pela sala (é possível ir até a cozinha, mas a maior parte das interações acontece na sala).

Porém, de uma maneira geral estas limitações são coerentes com a proposta do jogo, cuja mecânica de interação central se dá no âmbito psicológico e comunicacional, principalmente através do diálogo e interação social e emocional com os personagens. Isto é, assim como acontece com Eliza, a limitação física da ação do jogador não prejudica a experiência (pelo menos não a princípio), e é justificada a partir destes elementos temáticos e narrativos.

Liberdade e Agência

Na comparação feita anteriormente entre o sistema de IA e o engine de física, um dos exemplos utilizados foi o jogo Angry Birds (p. 47). Este jogo foi escolhido pela sua simplicidade, o que permitiu identificar facilmente os seus elementos e características, e traçar as analogias e comparações necessárias.

Em Angry Birds, o escopo da interação permitida ao jogador é bastante específica e limitada - a única ação que pode ser realizada é arremessar os pássaros. Não existem outas formas de interação além desta (por exemplo, não é possível controlar o voo dos pássaros diretamente, nem manipular outros elementos). Porém, como este escopo é definido e comunicado ao jogador de maneira clara e coerente com o restante da experiência, o resultado é um grau elevado de agência. Isto é, dentro

[197] A barreira também pode se manifestar na forma de um "ciclo", ou um ambiente que se fecha em si mesmo, como um planeta, por exemplo.

deste escopo, o jogador tem liberdade total, no sentido em que o engine de física não dá preferência por uma escolha de arremesso sobre outra, ou de uma solução específica para uma determinada fase.

Note que o fato do jogador ter um alto grau de agência não significa que a autoria - o papel do designer - é necessariamente diminuída. Angry Birds tem uma série de regras, e uma lógica ou estrutura pela qual o jogador progride de fase em fase, definidas pelos designers do jogo. Entretanto, isso se manifesta na forma de uma **camada adicional** de jogabilidade, e que não influencia o comportamento do engine de física.

Por exemplo, considere a fase representada na figura abaixo (fig. 55). Há uma quantidade grande de inimigos (os porcos), e a estrutura que os protege a princípio parece ser difícil de derrubar, exigindo múltiplos arremessos até que seja possível eliminar todos os inimigos. Entretanto, o jogador mais atento pode perceber que existe uma pedra grande no topo da estrutura. Quando atingida, ela se solta e cai sobre os porcos, eliminando todos de uma vez. Esta solução foi prevista pelos criadores do jogo, e provavelmente é a única maneira de solucionar esta fase usando apenas um arremesso (de forma a obter o número máximo de "estrelas" na fase). Porém, é possível vencer a fase utilizando outra estratégia.[198]

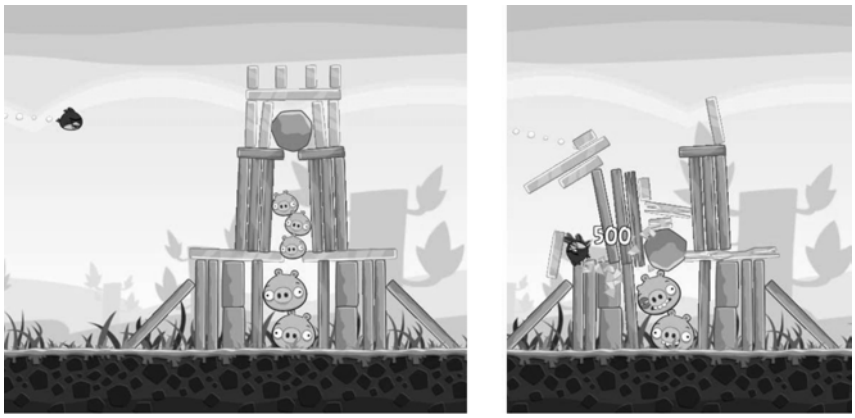


Figura 55 – Angry Birds (fase da torre)

No sistema de Inteligência Artificial de Façade, por outro lado, a agência do jogador não é tão ampla. Isso é inevitável, uma vez que este sistema tem um escopo muito mais abrangente e complexo, em comparação com o engine de física em Angry Birds. Isto é, enquanto que em Façade a intenção é de simular dinâmicas complexas de relacionamento entre personagens humanos, Angry Birds representa um universo fictício estilizado. A consequência disso é que as limitações no caso de Façade se tornam mais numerosas e evidentes.

[198] Este tipo de situação, na qual o design da fase favorece soluções bastante específicas, é relativamente comum em Angry Birds. Entretanto, no jogo predomina uma abordagem mais dinâmica, que valoriza a agência do jogador, dando abertura para uma ampla variedade de soluções e estratégias.

Por exemplo, em Façade é possível conversar com Trip e Grace sobre a viagem que o casal fez para a Itália, mas não sobre o prédio em que moram; o jogador pode interagir fisicamente com um vaso e pegá-lo, mas não há opção para mover uma mesa; o sistema de IA permite aos personagens reconhecer quando o jogador está concordando ou discordando deles, mas não detecta outros comportamentos como curiosidade, tristeza ou dó.

É evidente que Façade e Angry Birds tratam-se de propostas radicalmente diferentes. Embora compartilhem da mesma base procedural, estas duas produções empregam estes mecanismos de maneiras distintas, para atingir objetivos próprios. No caso de Façade, a intenção é gerar um drama interativo, enquanto Angry Birds trata-se de um jogo casual. Entretanto, do ponto de vista particular da escolha de escopo, um sistema como Angry Birds apresenta maior consistência no emprego das estratégias procedurais do que Façade.[199]

Em Angry Birds, todos os fatores e elementos que influenciam a simulação e a mecânica de interação são conhecidas e consistentes. Quando o jogador lança um pássaro, por exemplo, a sua trajetória é definida por fatores como a gravidade, o "peso" do projétil, a força e direção do arremesso. Por exemplo, não existe uma "força misteriosa" que guia o pássaro para um certo ponto da estrutura, de forma que ela desmorone de uma forma particularmente divertida ou interessante.

Já em Façade, como foi visto, há um gerenciador de drama, que atua constantemente regulando o comportamento do sistema, de acordo com critérios e objetivos preestabelecidos, garantindo que em determinados momentos da sessão de jogo certos tipos de eventos irão ocorrer.

Mesmo quando o design de uma fase de Angry Birds parece sugerir uma solução particular, o sistema dá abertura para que o jogador tente soluções alternativas. Além disso, mesmo quando o jogador segue à risca a solução sugerida para uma fase, nem sempre é possível prever o que vai acontecer, uma vez que tudo que ocorre na simulação é definido proceduralmente pela simulação de física do jogo.[200]

[199] O mesmo pode ser dito de outros jogos igualmente simplificados e estilizados, como Pac-Man, visto anteriormente (p. 49).

[200] Guardadas as devidas proporções, este mesmo tipo de agência e liberdade está presente na maioria dos jogos baseados em engines de física.

4.3 Coerência na Mecânica de Jogo

A seguir elaboraremos a respeito de outro tipo de experiência narrativa digital, os adventures. Neste gênero o jogador geralmente controla um personagem, visitando locais, dialogando com outros personagens e manipulando objetos, de forma a atingir um objetivo.

De uma maneira geral, os adventures apresentam uma intensidade procedural mais baixa em comparação com as abordagens apresentadas anteriormente. Isto é, neste gênero o conteúdo estático predeterminado apresentado ao interator é proporcionalmente maior do que aquele que emerge do sistema, na forma de situações e comportamentos dos personagens.

As potenciais bifurcações no enredo e nos diálogos com os personagens (ou NPCs) em um adventure geralmente não tem o mesmo nível de complexidade do que em um projeto como Façade, por exemplo. Os adventures também geralmente não apresentam o mesmo nível de granularidade e variabilidade na narrativa do que um sistema como Tale-Spin.

Nesta seção, analisaremos a questão da coerência na **mecânica de jogo**, examinando como ela se manifesta no contexto dos adventures.

Regras Instáveis

Embora a maior parte dos adventures funcione de maneira similar, eles tendem a ter nuances e características particulares. Geralmente os primeiros momentos de interação com um jogo deste gênero servem como uma maneira de transmitir ao jogador os seus mecanismos e lógica particulares.

Por exemplo, ao deparar-se com um personagem, o jogador aprende como se dá a mecânica de conversação; ao encontrar um objeto, os comandos para examinar e pegar objetos são descobertos; e assim por diante.

Entretanto, na maioria dos adventures estas "regras" não são estáveis, e muitas vezes são subvertidas, ignoradas ou modificadas, de acordo com o contexto ou situação de jogo. Por exemplo, eventualmente o interator encontra um personagem com o qual não é possível dialogar, ou então é impedido de pegar um determinado objeto, sem nenhuma justificativa lógica, e tampouco narrativa, para tal.

No adventure clássico Leisure Suit Larry in the Land of the Lounge Lizards (Al Lowe, 1987), em um determinado ambiente o interator pode pegar uma rosa que está sobre uma mesa, porém mais adiante é impedido de pegar um aparelho de televisão. Estas duas situações são ilustradas na figura abaixo (fig. 56) - quando o jogador dá a instrução para Larry (o personagem de terno branco) pegar a TV, o jogo exibe uma mensagem afirmando que "não há razão" para pegar este objeto.

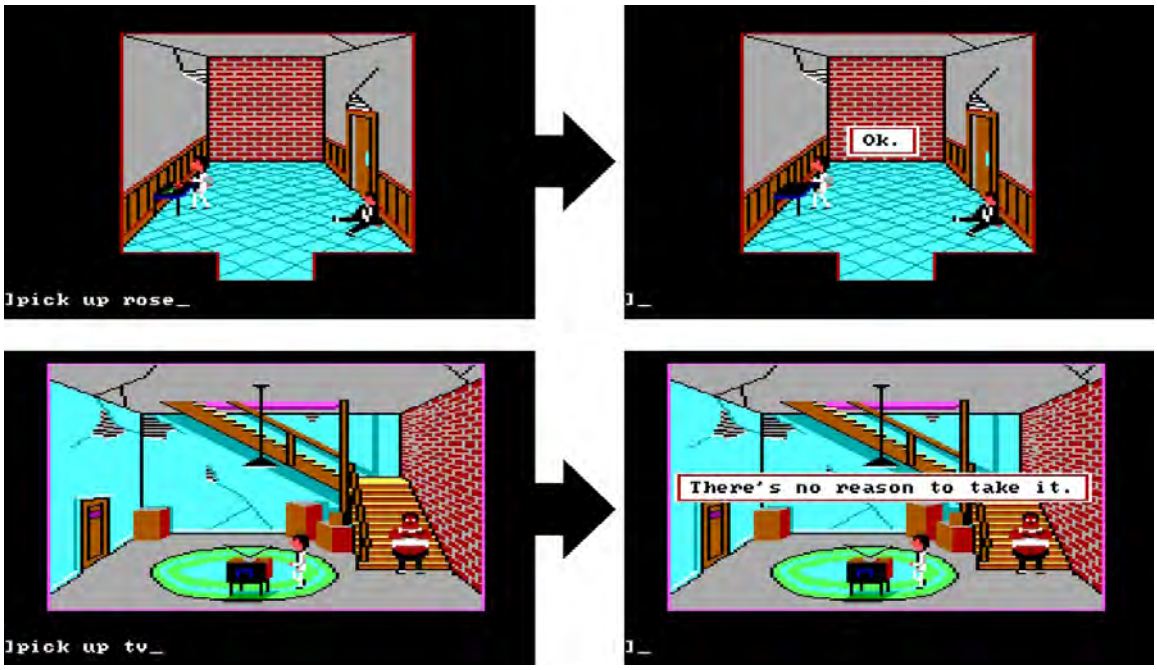


Figura 56 – Leisure Suit Larry: inconsistência na ação "Pegar"

Em seu estudo sobre adventures, Clara Fernández-Vara explica que "restringir o número de objetos com os quais se pode interagir e as possíveis ações associadas a eles facilita a geração da estória, simplesmente porque não há muito que resta para o jogador fazer." (FERNÁNDEZ-VARA, 2011a:137). Segundo ela, "este deslocamento para uma interação menos granularizada tem afetado os diferentes aspectos do design de jogos" (Idem, 2009:xii).[201]

Limites são inevitáveis, e mesmo indispensáveis, em um sistema ou simulação, uma vez que servem para definir os seus parâmetros, características e propriedades. Entretanto, a questão apresentada aqui não é o fato de que uma determinada ação não é possível, mas sim a coerência desta limitação. A partir do momento em que são estabelecidas a mecânica de jogo e o grau de agência que o jogador terá no sistema, cria-se também uma expectativa no interator. Se a lógica deste sistema for

[201] Fernández-Vara e sua equipe desenvolveram um sistema chamado Puzzle-Dice (MIT Game Lab), voltado para a geração procedural de conteúdo para adventures (FERNÁNDEZ-VARA; THOMSON, 2012). Citações: tradução nossa - textos originais: "[R]estricting the number of objects to interact with and the possible actions associated with them facilitates generating the story, simply because there is not much else left for the player to do." (FERNÁNDEZ-VARA, 2011a:137); "This move to a less fine-grained interaction has affected different aspects of game design" (Idem, 2009:xii).

subvertida, rompe-se também a consistência no universo ficcional, consequentemente frustrando as expectativas do interator e prejudicando a fruição da experiência.

Se o sistema afirma que "não há razão" para executar uma determinada ação que o interator escolheu realizar, isso significa que o jogo está decidindo pelo jogador, impondo uma determinada escolha ou caminho em detrimento de outro.

Embora esta limitação seja similar a aquelas encontradas em Façade, por exemplo, há uma diferença fundamental entre elas. Por um lado, nos dois casos as ações do jogador são limitadas de forma a garantir a realização de um determinado enredo predeterminado. Tanto nos adventures quanto em Façade, embora exista uma série de desfechos possíveis para a narrativa, estas variações produzem apenas uma ilusão de liberdade, uma vez que se baseiam em uma estrutura global fundamentalmente predeterminada e linear. Apesar da existência de bifurcações, e a possibilidade de múltiplas conclusões - ou finais - para a estória, cada um destes percursos possíveis é predeterminado (de maneira similar aos livros Choose Your Own Adventure, ou os "filmes interativos"). Entretanto, em Façade há uma preocupação maior em garantir uma experiência consistente para o jogador.[202]

Como foi visto anteriormente, em Façade são empregados uma série de recursos de forma a eliminar ou minimizar a percepção dos limites da simulação por parte do interator. Um destes recursos, por exemplo, é limitar o espaço físico ao apartamento, desta maneira restringindo drasticamente a quantidade de personagens e objetos em cena. Esta limitação faz sentido dentro da narrativa estabelecida em Façade - isto é, fica claro para o jogador que o foco da experiência se encontra na dinâmica social e psicológica entre aqueles personagens.

Em Leisure Suit Larry, por outro lado, quando o jogador é impedido de pegar a TV, por exemplo, este limite não é justificado dentro do contexto do jogo. O jogador é apenas informado que "não há razão" para tal ação, sendo que certamente ela existe (de outra forma, o jogador não teria tentado realizá-la), apenas não seria reconhecida pelo sistema de jogo.

Esta não é uma questão de natureza prática, que poderia ser resolvida ampliando a agência do jogador, oferecendo mais ações e opções de interação com objetos, por exemplo. Ao invés de uma ampliação quantitativa no sistema já existente, tratamos aqui de uma diferença qualitativa.

[202] Nesta análise, a comparação feita entre o gênero do adventure e Façade é focada na estrutura da narrativa. Há inúmeras distinções em outros aspectos da experiência, como por exemplo no comportamento dos personagens, que em Façade é significativamente mais granularizado.

Isto é, trata-se de garantir que os elementos já existentes, suas características, comportamentos e relacionamentos, e as interações permitidas ao jogador, configurem-se em um sistema coerente.

Na situação mencionada acima, no exemplo do Leisure Suit Larry, uma maneira simples de preservar a consistência entre a narrativa e o sistema seria através de uma redefinição de escopo. Por exemplo, poderia ser estabelecido que Larry não consegue carregar objetos pesados, aproveitando o fato de que o personagem é caracterizado como um fracote. Isso poderia ser comunicado ao jogador através de uma mensagem simples, informando que a TV é pesada demais. Note que esta solução influenciaria principalmente a percepção que o jogador tem do comportamento do sistema, que permaneceria fundamentalmente inalterado.[203]

Confusões Formais

Vimos que toda linguagem ou forma artística apresenta limitações e restrições. Além de serem inevitáveis, estas limitações também servem para definir e descrever as particularidades de cada linguagem. A princípio, ao escolher atuar em uma determinada forma expressiva, o artista ou criativo está também aceitando estas limitações. Por exemplo, um desenhista não se importa com o fato de que a sua ilustração não exibe movimento - de outra forma, ele seria um animador.

Da mesma forma, um designer de adventures a princípio não vê problemas deste gênero apresentar um baixo grau de intensidade procedural, em comparação com outras formas narrativas digitais. Este designer não quer dar ao jogador liberdade total, mas sim uma agência controlada, dentro de uma série de situações, quebra-cabeças e desafios específicos, utilizando gráficos e animações predeterminadas para contar uma estória particular. O jogador que gosta deste gênero, por sua vez, tem interesse em consumir experiências desta natureza.[204]

De qualquer maneira, como vimos, independentemente da linguagem sendo empregada, o artista ou criativo deve atentar para a consistência e coerência no seu uso. Para isso, precisa ter consciência sobre os mecanismos, significados e implicações das ferramentas e estratégias utilizadas.

Entretanto, o que se verifica em muitos casos, como exemplificamos aqui com os adventures, é uma confusão formal na escolha da linguagem ou meio para a realização de um determinado projeto ou ideia. Um caso exemplar disso são os jogos do designer David Cage, como Heavy Rain (Quantic Dream, 2010) e Beyond: Two Souls (2013), nos quais predomina o emprego de estratégias lineares

[203] Convém notar, entretanto, que esta solução se aplicaria unicamente a esta situação do jogo. De forma a resolver este mesmo tipo de inconsistência em todas as situações similares, seria necessário lidar com elas individualmente, ou adotar uma solução sistêmica (mais trabalhosa e demorada).

[204] De fato, os adventures tendem a ser valorizados principalmente pelos seus elementos expressivos estáticos - gráficos, música, diálogos e principalmente o roteiro, ou estória (em oposição a aspectos sistêmicos).

de narrativa, como as *cutscenes* (cenas não interativas) e os QTE (Quick-Time Events - partes em que a interação do jogador é extremamente guiada e limitada).

Em um jogo como Heavy Rain, embora existam oportunidades para bifurcação, cada ramificação já se encontra predefinida. Na maior parte do tempo o jogador se depara com situações específicas, com um grau de agência baixo, levando a uma experiência similar à de um filme (PETRY, 2013). Isto é, embora Cage esteja utilizando a linguagem dos jogos digitais, ele o faz predominantemente a partir de estratégias expressivas emprestadas de outra forma artística, mais especificamente do Cinema.

Outro exemplo deste mesmo tipo de confusão formal é Fable (Peter Molyneux, 2004), um jogo de aventura e ação utilizado como um dos estudos de caso por Wardrip-Fruin em sua pesquisa (p. 57). Segundo relata Wardrip-Fruin, a produção deste jogo teve como ponto de partida um roteiro - similar ao de um filme ou peça. Isto é, um texto estático e linear, contendo descrições para cada cena e situação, incluindo as ações e falas de cada personagem.

Reiteramos que a questão sendo levantada aqui não é quanto ao emprego das mídias digitais como forma aditiva (isto é, a partir de estratégias expressivas de outros meios e formas). Estas práticas, embora não explorem o potencial expressivo próprio das mídias digitais, são perfeitamente legítimas. Muitos escritores, músicos e desenhistas empregam o computador para produzir trabalhos de natureza essencialmente tradicional. O próprio gênero do "filme interativo", no qual Heavy Rain a princípio se insere, é legítimo a partir do momento em que há coerência entre a mecânica de jogo e o modelo de interação pretendido. Além disso, a análise que fazemos não se refere à qualidade técnica ou estética destas produções, e tampouco às escolhas temáticas de seus criadores.

A questão que tratamos aqui é fundamentalmente diferente, e refere-se unicamente ao aspecto da confusão formal. Isto é, às situações nas quais há um conflito evidente entre, de um lado, a proposta ou projeto de criação e, do outro, a forma ou linguagem escolhida para a sua realização. Por exemplo, para Ian Bogost, o maior potencial de Heavy Rain reside justamente nos momentos em que o jogo rejeita um dos elementos principais do Cinema - a edição -, em oposição à intenção de Cage, que era de se aproximar da experiência do filme tradicional (BOGOST, 2010).

5 Considerações Finais

Neste último capítulo, revisaremos os principais temas da pesquisa, resgatando o conceito da **expressividade procedural**, sua manifestação no contexto específico das **narrativas e jogos digitais** e na **autoria procedural** de uma maneira mais ampla. Em seguida, exploraremos alguns dos **questionamentos** que surgiram ao longo das nossas investigações. Finalmente, apresentaremos as principais **contribuições** que esperamos ter feito com este estudo.

Expressividade Procedural

De forma a empregar qualquer meio ou linguagem de maneira expressiva, o artista ou criativo precisa compreender a natureza própria das suas estratégias e mecanismos. Isso não envolve necessariamente um nível de conhecimento ou domínio técnico específico da linguagem, mas sim um reconhecimento ou familiaridade (que pode se dar em grande parte de maneira inconsciente) sobre a maneira fundamental pela qual ela se distingue das demais. É a partir destes aspectos e características próprias de uma linguagem que ela pode se realizar como uma forma expressiva.

Este reconhecimento permite ao autor garantir a consistência e a coerência entre o seu projeto de criação e a linguagem escolhida para a sua realização ou materialização (da mesma forma, a recepção de uma obra ou produção também exige do público a capacidade de decodificar ou "ler" a linguagem empregada na sua criação).

No caso das mídias digitais, elas se distinguem dos demais meios e linguagens a partir do seu aspecto procedural, isto é, pela sua programabilidade e autonomia. Entretanto, embora este aspecto seja a base para qualquer atividade ou operação realizada em um computador, o emprego deste meio como uma forma expressiva - isto é, a **expressividade procedural** - só se realiza quando (1) a intenção do criador **não** é prática ou utilitária; e (2) o conteúdo sendo criado ou manipulado **não** é de natureza estática (isto é, referente a formas expressivas tradicionais).

A princípio, o emprego do computador para fins práticos ou para emular outras linguagens é perfeitamente válido e legítimo. Porém, nesta pesquisa, nos interessa especificamente o emprego **expressivo** das mídias digitais, a partir da autoria **procedural**. Em outras palavras, nos interessa buscar "dentre as funções algorítmicas, e outras propriedades especificas do computador, quais as suas possibilidades expressivas" (de Laurentiz, 2007:1645).

A expressividade procedural tem sido explorada por artistas e criativos desde os primórdios da computação, e esta abordagem se encontra cada vez mais presente nas artes e outras práticas criativas. Entretanto, apesar da ubiquidade das mídias digitais em praticamente todos os aspectos da sociedade moderna, a quantidade de criativos empregando estratégias procedurais (isto é, criando e manipulando algoritmos, direta ou indiretamente) ainda é pequena, relativamente falando. Esta proporção é ainda menor se considerar o uso da autoria procedural para fins expressivos ou artísticos. Assim, ainda predomina o uso do computador unicamente para emular práticas e formas expressivas tradicionais.

O artista e programador Casey Reas, figura ativa na discussão pela conscientização sobre a literacia procedural entre criativos e artistas, faz referência a esta situação ao declarar que "[d]a mesma forma que a fotografia lutou para ser levada a sério como um meio, o software esta lutando pela mesma coisa" (LUBELL, 2016).[205]

Há diferentes fatores que podem estar causando esta situação. Para alguns autores, isso se deve ao fato de que as mídias digitais ainda são relativamente recentes em comparação com outros meios e linguagens mais consolidadas, e portanto ainda estão em processo de amadurecimento. Outro fator são as confusões e mal-entendidos que existem a respeito da natureza deste meio, como vimos.

Esta situação se mostra de maneira particularmente evidente nas narrativas digitais (e em jogos contendo elementos narrativos), nas quais é comum haver um conflito entre a forma expressiva predominante na sua concepção e o meio escolhido para a sua realização. Ao invés de explorar as estratégias e mecanismos próprios do meio digital, estas produções muitas vezes se baseiam em formas expressivas tradicionais, de natureza estática e linear (principalmente Cinema e Literatura).

Pudemos investigar esta questão na prática nos estudos realizados durante o período sanduíche desta pesquisa. Tivemos a oportunidade de (1) participar ativamente do trabalho de pesquisa e desenvolvimento de um aplicativo de criação de jogos, (2) observar o seu uso na prática, no contexto de uma oficina, em paralelo a outras atividades criativas em formas e linguagens expressivas tradicionais, e posteriormente (3) analisar os arquivos dos projetos e (4) conversar com os criadores sobre o seu processo criativo, suas impressões e reflexões a respeito da experiência. Foi um processo valioso, e que contribuiu para informar a nossa investigação sobre a expressividade procedural de uma maneira mais ampla.

[205] Citação: tradução nossa - texto original: "The same way that photography struggled to be taken seriously as a medium, software is suffering for the same thing" (sem numeração de página)

Narrativas e Jogos Digitais

Na indústria de jogos digitais há um investimento cada vez maior nas tecnologias relacionadas aos aspectos visuais ou físicos das produções, em comparação com os aspectos narrativos, e o comportamento dos personagens.

Consideremos o estado atual das tecnologias de simulação de física, por exemplo. Os engines de física que existem hoje são o resultado de uma evolução gradual ao longo das últimas décadas, na forma de novas técnicas e teorias de modelagem e implementação de sistemas (além de se beneficiarem do surgimento de computadores e processadores cada vez mais rápidos e poderosos).

Suponha a representação da situação de um vaso caindo de uma mesa, por exemplo. Nos primórdios da computação isso só poderia ser feito a partir de um processo similar ao da animação tradicional (através de trajetórias e quadros-chave predefinidos). Hoje em dia isso pode ser simulado plenamente a partir de algoritmos, com base na topologia do objeto e nas propriedades físicas do material.[206]

Uma evolução similar e igualmente significativa tem acontecido nas tecnologias de representação dos aspectos visuais do ambiente virtual.

Do ponto de vista técnico, grande parte destes avanços estão associados ao aumento no grau de intensidade procedural do sistema. Isto é, cada vez mais elementos e aspectos da simulação passaram a poder ser implementados proceduralmente, ao invés de serem representados de maneira estática. Da perspectiva do artista ou criativo, isso significa mais oportunidades para explorar as estratégias procedurais de criação.

Entretanto, este é apenas um potencial, que o criativo pode decidir explorar ou não. Alguns autores podem optar por empregar o computador unicamente para emular técnicas tradicionais de animação, por exemplo. Além disso, mesmo quando há intenção de se expressar a partir da autoria procedural, isso não significa necessariamente que esta abordagem precisa ser explorada de maneira intensa, ou em todos os aspectos da produção ou do processo criativo.

Um criativo pode se expressar de maneira plena através de uma única linha de código em uma linguagem simples, enquanto que outro pode ser incapaz de fazê-lo através de um sistema complexo,

[206] À medida em que a tecnologia evolui, as criações também podem se tornar mais complexas, tirando proveito das novas possibilidades. Por conta disso, mesmo os jogos mais modernos ainda usam animações predefinidas em determinados aspectos ou elementos, tanto nas *cutscenes* quanto durante o jogo em si. Na animação, o termo "quadro-chave" (em inglês, "key frame") refere-se aos quadros contendo os momentos chave de uma determinada cena.

usando uma linguagem mais sofisticada. É o caso dos sistemas de simulação de física, representação visual e geração procedural de conteúdo - apesar de tratarem-se de sistemas robustos, complexos e sofisticados, são relativamente pouco explorados de maneira expressiva.

A diferença entre os algoritmos que compõem um engine de física e qualquer outro tipo de sistema é essencialmente arbitrária. Entretanto, como vimos, certas tecnologias e áreas tendem a receber mais atenção do que outras, principalmente na indústria de jogos.

No caso dos elementos narrativos, muitos jogos continuam reproduzindo essencialmente o mesmo modelo baseado em bifurcações, encontrado nos "filmes interativos" e livros Choose Your Own Adventure. No comportamento de personagens, a expressividade da "doutora virtual" Eliza (p. 236) ainda supera a de muitos dos personagens em narrativas e jogos digitais modernos (mesmo considerando que muitos destes envolvem um grau de intensidade procedural significativamente superior ao de Eliza).

Esta situação se manifesta predominantemente em jogos comerciais, produzidos por grande estúdios e com ampla distribuição. Na arte, embora as estratégias procedurais de criação ainda sejam empregadas por uma quantidade relativamente pequena de profissionais, estes geralmente o fazem de maneira plena, explorando esta abordagem de diferentes maneiras, e em uma grande variedade de áreas e aplicações.

O mesmo pode ser verificado em outras práticas criativas que, embora não sejam predominantemente artísticas, enfatizam os aspectos expressivos, como acontece em determinados tipos de jogos independentes e produções experimentais. Por exemplo, é o caso dos jogos da produtora Thatgamecompany, como Flower (2009) e Journey (2012), descritos pelos pesquisadores Hermes Hildebrand e Bruno de Paula como sendo inspirados predominantemente por "sentimentos" (HILDEBRAND e de PAULA, 2013). Embora a rigor estes jogos podem ser considerados comerciais, eles se caracterizam por um forte viés experimental.

Autoria Procedural

O processo criativo envolvido na produção ou manipulação direta ou indireta de algoritmos - isto é, a autoria procedural - se baseia em mecanismos fundamentalmente distintos daqueles de linguagens e formas tradicionais. Por exemplo, suponha um autor de narrativa digitais que deseja explorar a expressividade procedural em seu trabalho. A princípio este criativo não deseja produzir instâncias específicas ou particulares de situações, gestos ou diálogos, como aconteceria no caso de um escritor empegando a forma tradicional da Literatura. Ao invés disso, este autor deseja criar sistemas e comportamentos, através dos quais diferentes instâncias ou situações podem se materializar ou realizar. Em alguns casos, isso também pode envolver variações de acordo com parâmetros aleatórios, ou de acordo com a interação do receptor. Muitas vezes o criador se surpreende com os resultados da execução do seu próprio algoritmo.

A pesquisadora Cecilia Almeida Salles, ao tratar sobre o processo de criação artística de uma maneira mais ampla, afirma:

> Aceitar a intervenção do imprevisto implica compreender que o artista poderia ter feito aquela obra de modo diferente daquele que fez. Ao assumir que há concretizações alternativas, admite-se que outras obras teriam sido possíveis. Chegamos, desse modo, à possibilidade de que mais de uma obra satisfaça as tendências de um processo. (SALLES, 2006:22)

No caso da autoria procedural em mídias digitais, por conta da natureza essencialmente metacriativa deste processo, de certa forma estas obras possíveis podem se manter em aberto, uma vez que as "tendências" podem ser codificadas na forma dos algoritmos. Evidentemente esta não se trata de uma simples transferência do projeto de criação, da mente do autor para a máquina digital, mas sim um deslocamento, que pode ser sutil ou radical, mas nunca completo.

Há inúmeros possíveis empregos e aplicações da autoria procedural para fins criativos ou expressivos, em diferentes áreas e suportes. Por exemplo, dez anos atrás o videoartista Bill Viola refletia sobre um futuro no qual o profissional responsável pela edição de vídeo não necessariamente realizaria a tarefa diretamente, mas sim via algoritmos - ele também previu que eventualmente as limitações destas práticas não residiriam mais no meio em si, mas sim na capacidade dos criadores em compreender e empregar estas ferramentas:

> Editar [vídeo] se tornará **o ato de escrever um aplicativo** que dirá ao computador como arranjar (isto é, a ordem das cenas, cortes, transições, etc.) a informação que se encontra na memória, reproduzindo-a em uma sequência determinada em tempo real ou permitindo que o observador intervenha [...] **Novos talentos e habilidades são necessárias para criar programas** - isto não é edição tal qual a conhecemos. [...] Depois de tantos anos, o vídeo está finalmente ganhando "inteligência", o olho está sendo reconectado ao cérebro. Como tudo mais, entretanto, veremos que as limitações que surgem se encontram mais nas habilidades e imaginações dos produtores e usuários, ao invés de nas ferramentas em si. (VIOLA, 2003:467)[207]

O artista Jer Thorp, cujo trabalho inclui produções de arte generativa e visualização de dados, expressa uma ideia similar, a partir da sua experiência com a linguagem de programação Processing:

> Como um "caderno de esboços" digital, Processing é um lugar onde você pode rabiscar ideias na forma de código [...] A diferença entre Processing e um caderno de esboços físico é que aquele esboço [em código] pode se tornar algo maior e mais elaborado. É como se o seu livro de esboços pudesse de repente se tornar um filme ou uma escultura. (ALAEJOS, 2013)[208]

A distinção entre o que é um conteúdo de natureza procedural e o que é de natureza estática é essencialmente arbitrária, e nem sempre tão clara - embora nem por isso deixe de ser significativa. Além disso, mesmo produções predominantemente procedurais geralmente envolvem elementos estáticos, inclusive nos níveis mais altos de abstração (como os modelos 3D dos personagens de Façade, ou as texturas dos ambientes nos jogos do MissionMaker). De uma maneira geral, sempre haverá um limite prático para o quanto pode ser modelado na forma de algoritmos.

Nos níveis mais baixos de abstração, o algoritmo que descreve os sistemas computacionais também é estático, no sentido em que ele não muda a cada nova execução (ou compilação), ou de acordo com

[207] O documentário CLOUDS (George e Minard, 2014), citado no início desta tese (p. 31), pode ser considerado como uma realização da previsão de Viola. Citação: grifo nosso; tradução nossa - texto original: "Editing will become **the writing of a software program** that will tell the computer how to arrange (i.e., shot order, cuts, dissolves, wipes, etc.) the information on the disc, playing it back in the specified sequence in real time or allowing the viewer to intervene. [...] **New talents and skills are needed in making programs** — this is not editing as we know it. [...] After all these years, video is finally getting "intelligence," the eye is being reattached to the brain. As with everything else, however, we will find that the limitations emerging lie more with the abilities and imaginations of the producers and users, rather than in the tools themselves." (o texto referente a esta citação é de 1995, mas faz parte de uma compilação de 2003)

[208] Citação: tradução nossa - texto original: "As a digital sketchbook [Processing is] somewhere where you can scribble down your ideas in code (. . .) [T]he difference between Processing and a real sketchbook is that that sketch can become something bigger and more elaborate. It's as if your art [or drawing] sketchbook could suddenly become a movie [or a] sculpture."

elementos ou estímulos externos. Isto é, embora os algoritmos representem ou produzam processos dinâmicos (**procedurais**), em seu nível mais fundamental eles são sempre compostos de elementos **estáticos** - os caracteres ou blocos fundamentais da linguagem de programação ou sistema de autoria sendo empregados.[209]

Independentemente da linguagem empregada, o processo criativo sempre se dá em torno dos pensamentos de um artista ou criativo, muitas vezes de maneira difusa e sem estrutura. Entretanto, os gestos realizados pelo autor para a realização ou materialização da produção ou performance são sempre direcionadas a elementos estáticos.

Na maior parte das artes tradicionais, estes elementos geralmente são os próprios componentes da obra final - a tinta na pintura, as palavras na literatura, os movimentos do corpo na dança e assim por diante. Na expressividade procedural, por outro lado, estes gestos são de natureza metacriativa - produzem algoritmos (mesmo quando isso acontece através de uma interface gráfica), que por sua vez são executados de forma a realizar (ou gerar) a produção ou performance.

Em essência, de forma a se expressar, o criador precisa sempre determinar (isto é, tornar estático, fixar) algum elemento ou aspecto dentro do sistema através do qual irá compor a sua produção. Reside justamente nesta escolha (ou conjunto de escolhas) a sua ideia ou proposta poética. Assim, os conceitos de dinâmico e fixo, procedural e estático, não são absolutos ou excludentes, mas sim dois extremos de uma escala contínua - e, no caso das estratégias expressivas procedurais, ambos são necessários.

Alguns criadores buscam evitar ou minimizar a influência destes elementos estáticos na produção final a partir de uma escolha de escopo particular. Outros fazem o mesmo a partir de uma estratégia de certa forma oposta, dedicando-se justamente a enfrentar, explorar e expandir estes limites.

É o caso dos criadores de "demos" e os criativos na área da "procgen", mencionados anteriormente na tese. Na área de jogos, um exemplo disso é No Man's Sky (Hello Games), que promete um universo com "todo átomo procedural" (ou "every atom procedural"). Michael Mateas, um dos criadores de Façade, também chega a sugerir que eventualmente todo conteúdo pode ser procedural - nas suas palavras, um "tudo procedural" ("*procedural everything*") (MATEAS, 2011).[210]

[209] Mesmo no caso de um código que é gerado dinamicamente, o algoritmo que o gera é estático.
[210] A declaração associada a No Man's Sky faz parte da campanha promocional do jogo.

Questionamentos

> Como o software tem afetado as artes visuais?
> Qual é o potencial para o software dentro das artes visuais?
> Como designer ou artista, porque eu iria querer ou precisar escrever software?
>
> (REAS, McWILLIAMS; LUST, 2010:8)

Durante o processo desta pesquisa, à medida em que nos aprofundamos no nosso objeto de estudo e desenvolvemos as nossas elaborações teóricas e reflexões sobre a autoria e a expressividade procedural, também identificamos uma série de questionamentos e novos rumos de investigação possíveis. Alguns destes questionamentos existiam desde os momentos iniciais da pesquisa, contribuindo para definir o nosso objeto central de estudo, outros se materializaram gradualmente ao longo do processo, e outros foram inspiradas a partir de referências externas.[211]

As categorias nas quais agrupamos os questionamentos a seguir não devem ser consideradas de maneira rígida e tampouco definitiva. Além disso, que há claras intersecções e conexões entre elas.

Sobre **o artista ou criativo**, a partir do qual se deu a perspectiva deste estudo, podemos perguntar: Qual é o perfil de quem opta pela autoria procedural? Qual é o seu repertório, seus talentos e habilidades? Quais são as suas motivações em escolher esta abordagem expressiva, em detrimento das formas tradicionais? E também, quais são as causas e significados nos casos em que há dissonância entre esta escolha e o projeto de criação?

Quanto **ao estudo destas práticas**: Quais são as ferramentas teóricas pelas quais é possível identificar, descrever e analisar as produções, os artistas e criadores, seu processo criativo e os meios empregados, associados à autoria procedural? Como se manifestam nesta forma expressiva determinadas ideias relacionadas às práticas artísticas e criativas, como por exemplo a noção de estilo?

Sobre o **público**: Como se configura a recepção de uma produção expressiva procedural, ou que contenha elementos ou aspectos desta natureza? De que maneira esta fruição pode ser explorada de maneira criativa, e como ela se compara à de outras formas e linguagens tradicionais? Que repertório e literacia são exigidos do público neste caso? Em que contexto este tipo de produção pode ser apresentada e consumida?

[211] A epígrafe acima se refere especificamente à influência da autoria procedural nas artes visuais, entretanto consideramos ela no contexto mais amplo da criação em mídias digitais. Epígrafe: tradução nossa - texto original: "How has software affected the visual arts? What is the potential for software within the visual arts? As a designer or artist, why would I want or need to write software?"

Em relação à **proposta de criação e a obra ou produção em si**: Quais são os mecanismos, estruturas e regras fundamentais que regem a autoria procedural (e como elas se relacionam com aquelas das demais formas e linguagens tradicionais)? Que tipos de tema, gênero, estética e abordagem são facilitados e incentivados pela expressividade procedural (e quais são prejudicados ou mesmo impraticáveis por ela)?[212]

E também: A expressividade procedural é uma forma artística ou uma abordagem alternativa a outros meios e linguagens? Como se dá a tradução de outras formas e linguagens tradicionais para a linguagem própria do meio digital (e como estas linguagens podem inspirar e influenciar umas às outras)? Considerando que a estrutura e lógica particulares de cada linguagem podem produzir, sugerir, facilitar ou inspirar conexões e escolhas expressivas arbitrárias (como é o caso da rima na poesia, por exemplo), como isso se manifesta na expressividade procedural?

Sobre **as aplicações práticas da autoria procedural**: Quais são as possíveis formas e linguagens que podem derivar da expressividade procedural? Assim como Façade sugere uma abordagem procedural ao drama, fundamentalmente distinta do Cinema e do Teatro, como esta abordagem pode se manifestar de outras maneiras? Até que ponto este tipo de produção tem o potencial de se consolidar culturalmente, de maneria similar a outras linguagens e formas expressivas tradicionais?[213]

Quais são as possíveis atividades ou profissões "do futuro" para aqueles artistas e criativos fluentes na autoria procedural? Por exemplo, práticas como a "edição procedural" sugerida por Viola, permanecerão restritas a projetos experimentais, e para um público restrito, ou poderão estar presentes em outras áreas, e de maneira mais abrangente? Ou então: Assim como a fotografia é empregada - e reconhecida na sociedade - como uma forma de comunicar e expressar ideias, o mesmo pode acontecer com a criação procedural?[214]

Estes questionamentos, ao mesmo tempo em que ilustram alguns dos tópicos e reflexões desta pesquisa, também sugerem possíveis rumos e linhas de investigação futuras sobre os temas tratados na tese.

[212] A composição na capa desta tese ilustra este questionamento: É possível afirmar que a obra de Verostko (parte esquerda da imagem) realiza de maneira mais plena e consistente a abordagem da expressividade procedural, em comparação com Façade, uma forma híbrida?

[213] Por exemplo, as técnicas generativas de criação são encontradas na música, porém predominantemente restritas a determinados gêneros específicos (principalmente na música eletrônica), e a uma etapa particular do processo criativo (para a geração de elementos que irão posteriormente compor a música). Este emprego eventualmente se tornará mais presente em outros gêneros ou momentos do processo criativo? Algumas destas ideias são exploradas, embora de maneira limitada, no "album app" Biophilia (Björk, 2011; p. 31), por exemplo.

[214] Além do jornalismo feito através de texto, temos manifestações na forma de vídeo, foto e desenho. Eventualmente será consolidado um jornalismo feito a partir de estratégias procedurais? Esta é a proposta de produções como Cutthroat Capitalism e do projeto Game-o-Matic, por exemplo (p. 31).

Contribuições

Através desta pesquisa esperamos contribuir na compreensão sobre o uso expressivo da autoria procedural em mídias digitais, e no seu papel fundamental na realização da expressividade própria deste meio.

Nossa investigação foi feita principalmente a partir da perspectiva do processo criativo, analisando artistas, criativos, obras e ferramentas (bem como relatos e reflexões dos criadores sobre o seu próprio trabalho). Neste percurso, também tratamos sobre os aspectos técnicos da lógica computacional, a questão da interatividade e a recepção, porém sempre com o objetivo de informar e embasar a nossa investigação principal mais ampla sobre a expressividade procedural.

Ao longo da tese, também apresentamos alguns exercícios teóricos elaborados por nós, de forma a ilustrar determinados temas e tópicos abordados. Estes exemplos variavam em termos de dimensão, complexidade e tipo de implementação. Alguns representavam ideias mais práticas, enquanto que outros tratavam-se de exercícios teóricos mais livres ou reflexivos. Através destes exemplos, a intenção foi de ilustrar como as estratégias procedurais de criação são empregadas para fins expressivos, e propor possibilidades ainda inexplorados para esta abordagem.

Esperamos também ter contribuído para outros temas, áreas e linhas de investigação que fizeram parte desta pesquisa. Durante o período de estudos no exterior, além da minha participação na pesquisa e desenvolvimento do MissionMaker no contexto do projeto Playing Beowulf, também contribuí de maneira ativa para a área de estudos multimodais (uma das ferramentas teóricas adotadas nos estudos práticos desta pesquisa), a partir de elaborações teóricas a respeito da proceduralidade como Modo (rodapé 93; p. 91). Também esperamos que nossa metodologia de análise para o estudo dos processos criativos em mídias digitais possa ser útil para a realização de futuras investigações a respeito da autoria procedural e outras práticas criativas neste meio (p. 79).[215]

[215] As produções artísticas e experimentais realizadas por mim ao longo desta pesquisa, bem como em colaboração com outros artistas, incluindo com o Grupo Poéticas Digitais (ECA/USP), também serviram como uma forma de investigação ativa do temas tratados (mais informações no rodapé 4; p. 21).

Referências

BOGOST, Ian. **Unit Operations**: an approach to videogame criticism. MIT Press, 2006. 264 p.

_____. Persuasive Games: The Picnic Spoils the Rain. **Gamasutra**. Mai., 2010.
URL: www.gamasutra.com/view/feature/132800/persuasive_games_the_picnic_.php

BURN, Andrew. Reworking the text: online fandom. In: CARR, Diane; BURN, Andrew; BUCKINGHAM, David; SCHOTT, Gareth. **Computer Games: Text, Narrative and Play**. Cambridge: Polity, 2006. 7, p. 88-102.

_____. From Beowulf to Batman: Connecting English and Media Education. In: DAVISON, Jon; DALY, Caroline; MOSS, John. (Eds.). **Debates in English Teaching**. London: Routledge, 2010. 8, p. 117-130. (The Debates in Subject Teaching Series)
URL: aburn2012.files.wordpress.com/2014/04/from-beowulf-to-batman-final.pdf

_____. The Kineikonic Mode: towards a multimodal approach to moving-image media. In: JEWITT, Carey (Ed.). **The Routledge Handbook of Multimodal Analysis**. Ed. 2. London: Routledge, 2014. 27, p. 373-383.

COHEN, Harold. **How to Make a Drawing**. Transcript of a talk given at the Science Colloquium, National Bureau of Standards, Washington DC. Dez. 1982. 15 p.
URL: haroldcohen.com/aaron/publications/how2make.pdf

_____. Off the shelf. **The Visual Computer**, v. 2, p. 191-194, 1986.
URL: www.kurzweilcyberart.com/aaron/pdf/offtheshelf.pdf.

COX, Geoff; WARD, Adrian. Perl. In: FULLER, Matthew (Ed.). **Software Studies**: a Lexicon. Massachusetts: The MIT Press, 2008. p. 207-212 (Software Studies).

CRAWFORD, Chris. Process intensity. **Journal of Computer Game Design**, v. 1, ed. 5, 1987.
URL: www.erasmatazz.com/library/the-journal-of-computer/jcgd-volume-1/process-intensity.html

DARE. **MissionMaker2**: User Guide (DARE Collaborative; PPT). 2015.

FERNÁNDEZ-VARA, Clara. **The Tribulations of Adventure Games**: integrating story into simulation through performance. Tese (Doutorado), Georgia Institute of Technology, 2009. 322 p.
URL: smartech.gatech.edu/handle/1853/31756

_____. From "Open Mailbox" to Context Mechanics: Shifting Levels of Abstraction in Adventure Games. In: **Proceedings of the 6th International Conference on Foundations of Digital Games**, 2011a, p. 131-138.
URL: www.academia.edu/733186/From_Open_Mailbox_to_Context_Mechanics_Shifting_Levels_of_Abstraction_in_Adventure_Games

Referências de acordo com a Associação Brasileira de Normas Técnicas (ABNT) - NBR 6023. Todas as referências online foram acess**ada**s em Março de 2016. Algumas das referências físicas incluem versões online. Referências em vídeo e filme estão apresentadas em separado.

_____. Game Spaces Speak Volumes: Indexical Storytelling. In: **Proceedings of DiGRA 2011 Conference: Think Design Play**, 2011b. 13 p.
URL: www.digra.org/wp-content/uploads/digital-library/Game-Spaces-Speak-Volumes.pdf

_____; THOMSON, Alec. Procedural Generation of Narrative Puzzles in Adventure Games: The Puzzle-Dice System. In: **Procedural Generation Content Workshop at Foundations of Digital Games 2012**, Raleigh, North Carolina, USA, 2012.
URL: gambit.mit.edu/readme/FDG-PGCWorkshop-Paper-Revised-Format.pdf

FERREIRA, Daniel Peixoto. **Poéticas Procedurais**: um olhar sobre o pensamento artístico e a expressividade do meio digital. Diss. (Mestrado), ECA/USP, São Paulo, 2011.

_____. **Beowulf Workshop: A MissionMaker Case Study**. Playing Beowulf: Gaming the Library (site do projeto), Nov., 2014a. 12 p.
URL: darecollaborativedotnet.files.wordpress.com/2015/02/
201502-beowulf-workhsop-a-missionmaker-case-study-v2.pdf

_____. **Beowulf Workshop: MissionMaker Conversations**. (Idem), Dez., 2014b. 26 p.
URL: darecollaborativedotnet.files.wordpress.com/2015/02/
201502-beowulf-workshop-missionmaker-conversations-v2.pdf

_____. **The Meaning Potential of Procedurality**: initial considerations on procedural semiotic resources. (Idem), Jul., 2015. 53 p.
URL: darecollaborativedotnet.files.wordpress.com/2015/03/
the-meaning-potential-of-procedurality-daniel-ferreira-july-2015.pdf

FOWLER, William. Manfred Mohr – the groovy German who taught computers to make art. **The Guardian**, Art and Design. Fev. 2016.
URL: www.theguardian.com/artanddesign/2016/feb/12/
manfred-mohr-the-man-who-taught-computers-to-make-art

GALVÃO, Ary Gonzalez (Trad.). **Beowulf**. (com notas) São Paulo: Editora Hucitec. 1992. 150 p.

GOV.UK. **National curriculum in England**: computing programmes of study. Department for Education. Set., 2013.
URL: www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study

GUMMERE, Francis B. (Trad.). **Beowulf**. Etext version by Robin Katsuya-Corbet (released into the public domain July 1993). 1909.
URL: www.poetryfoundation.org/poem/180445

HARGER, Brenda B. **Behind Façade**: An Interview with Andrew Stern and Michael Mateas. The Iowa Review Web (publicação original). 2006. 9 p.
URL: elmcip.net/critical-writing/behind-fa%C3%A7ade-interview-andrew-stern-and-michael-mateas

HÉRBERT, Jean-Pierre. algorist's statement (site do artista). 2015.
URL: jeanpierrehebert.com/13s.html

HILDEBRAND, Hermes Renato; de PAULA, Bruno Henrique. Potencial Expressivo nos Jogos Indie e Gamearte: Breve análise dos jogos Flower e Journey. **Obra Digital**: journal of communication and technology, v. 1, p. 1-10, 2013.

IMMERSIVE EDUCATION. **MissionMaker**. Material didático para educadores. 2010.

JEWITT, Carey (Ed.). **The Routledge Handbook of Multimodal Analysis**. 2nd Ed. London: Routledge, 2014. 544 p.

JUUL, Jesper. The Open and the Closed: Games of Emergence and Games of Progression. In: MÄYRÄ , Frans (Ed.). **Computer Games and Digital Cultures Conference Proceedings**. Tampere: Tampere University Press, 2002. p. 323-329.
URL: www.jesperjuul.net/text/openandtheclosed.html

KAY, Alan. Computer Software. **Scientific American**, v. 251, n. 3, Set., 1984. p. 52-59.

KHATCHADOURIAN, Raffi. World Without End: creating a full-scale digital cosmos. **The New Yorker**, Mai., 2015.
URL: www.newyorker.com/magazine/2015/05/18/world-without-end-raffi-khatchadourian

KOH, Yoree. New Education Bill to Get More Coding in Classrooms. Digits. **The Wall Street Journal**. Dez., 2015.
URL: blogs.wsj.com/digits/2015/12/10/new-education-bill-to-get-more-coding-in-classrooms

de LAURENTIZ, Silvia Regina Ferreira. Poesia Digital: uma estreita relação entre a poética e os códigos da lógica computacional. In: **Anais do XVI Encontro Nacional da ANPAP**: Dinâmicas Epistemológicas em Artes Visuais. p. 1645-1655, Florianópolis, 2007.

LEEUWEN, Theo Van. **Introducing Social Semiotics**. Psychology Press, 2005. 301 p.

LEWITT, Sol. Paragraphs on Conceptual Art (1967). In: STILES, Kristine; SELZ, Peter (Ed.). **Theories and Documents of Contemporary Art**: a Sourcebook of Artists' Writings. Berkeley: UCLA Press, 1996. p. 822-825.

LUBELL, Sam. Why Use a Paintbrush When You Can Make Mind-Bending Art With Code? **WIRED**. Fev. 2016.
URL: www.wired.com/2016/02/why-use-a-paintbrush-when-you-can-make-mind-bending-art-with-code

MACHADO, Arlindo. **Máquina e Imaginário**: o Desafio das Poéticas Tecnológicas. São Paulo: Editora USP, 1993.

MATEAS, Michael. Expressive AI: Games and Artificial Intelligence. In: **Level Up**: Digital Games Research Conference, Utrecht, Netherlands. Nov. 4-6, 2003. 9 p.
URL: users.soe.ucsc.edu/~michaelm/publications/mateas-digra2003.pdf

_____; STERN, Andrew. A Behavior Language: Joint Action and Behavioral Idioms. In: PRENDINGER, Helmut; ISHIZUKA, Mitsuru (Eds.). **Life-Like Characters**: Tools, Affective Functions, and Applications. Springer Science & Business Media, 2004, p. 135-162.
URL: classes.soe.ucsc.edu/cmps148/Spring07/MateasSternLifelikeBook04.pdf

_____; _____. Procedural Authorship: A Case-Study of the Interactive Drama Façade. In: **Digital Arts and Culture**: Digital Experience: Design, Aesthetics, Practice (DAC 2005), Copenhagen, Denmark, Dez. 1-4. 2005. 8 p.
URL: users.soe.ucsc.edu/~michaelm/publications/mateas1-dac2005.pdf

MCCLOUD, Scott. **Making Comics**. Harper Collins, 2011. 272 p.

MOLES, Abraham. (1971). **Arte e Computador**. Com a colaboração de Elisabeth Rohmer. Tradução de Pedro Barbosa. Porto: Edições Afrontamento, 1990. (Coleção Grande Angular/3).

MOLNAR, Vera. **Artist's Statement**: "Inconceivable Images". 1990.
URL: dam.org/artists/phase-one/vera-molnar/artist-s-statement

_____. **Touchware** (exposição; declaração da artista). 1998.
URL: www.siggraph.org/artdesign/gallery/S98/pione/pione3/molnar.html

MURRAY, Janet H. (1997). **Hamlet no Holodeck**: o Futuro da Narrativa no Ciberespaço. São Paulo: Itaú Cultural: Unesp, 2003.

PAREYSON, Luigi. (1966). **Os Problemas de Estética**. Ed. 2. São Paulo: Martins Fontes, 1989.

de PAULA, Bruno Henrique. **Jogos digitais como artefatos pedagógicos**: o desenvolvimento de jogos digitais como estratégia educacional. Diss. (Mestrado). UNICAMP, Instituto de Artes. 2015.

_____; VALENTE, José Armando; BURN, Andrew. O Uso de Jogos Digitais para o Desenvolvimento do Currículo para a Educação Computacional na Inglaterra. **Currículo sem Fronteiras**, v. 14, n. 3, p. 46-71, Set./Dez. 2014.
URL: www.curriculosemfronteiras.org/vol14iss3articles/paula-valente-burn.htm

PETRY, Arlete dos Santos. Narrativas fílmicas em Videojogos?. In: José da Silva Ribeiro & Carlos Eduardo Viana. (Org.). **Encontro de Cinema**: conferência internacional. 1ed.Viana de Castelo: Publisitio, 2013, v. , p. 141-151.

PRADO, Gilbertto; ASSIS, Jesus de Paula; JANOVITCH, Paula; GABBAI, Lívia; GOSUEN, Luciano; OLIVEIRA, Fábio; ARGUELLO, Gaspar; FURLAN, André; VANNUCHI, Hélia. Apontamentos para o game Cozinheiro das Almas. **Texto Digital** (UERJ), Florianópolis, SC, v. 2, p. 1-7, 2006.

REAS, Casey. **Electric Objects** (entrevista). Fev. 2015.
URL: zine.electricobjects.com/interviews/casey-reas

_____; McWILLIAMS, Chandler; LUST. **Form + Code**: in Design, Art, and Architecture. Princeton Architectural Press, 2010.

SALLES, Cecília Almeida. **Redes de criação**: construção da obra de arte. Vinhedo: Editora Horizonte, 2006.

_____. (1998). **Gesto Inacabado**: processo de criação artística. São Paulo: FAPESP: Annablume, 2009.

SCHWARTZ, Gilson. **Brinco, Logo Aprendo**: Educação, Videogames e Moralidades Pós-Modernas. 1. ed. São Paulo: Paulus, 2014.

SICART, Miguel. Against Procedurality. **Game Studies**, Ed. 3, v. 11. Dez. 2011.
URL: gamestudies.org/1103/articles/sicart_ap

SOGABE, Milton Terumitsu. Instalações interativas mediadas pela tecnologia digital: análise e produção. **ARS** (São Paulo), v. 9, n. 18, p. 1, 2011.
URL: dx.doi.org/10.1590/S1678-53202011000200004

TAYLOR, Grant D. (curadoria). **The American Algorists**: Linear Sublime. Catálogo da exposição (PDF). 2013. 37 p.

TEDRE, Matti; EGLASH, Ron. Ethnocomputing. In: FULLER, Matthew (Ed.). **Software Studies**: a Lexicon. Massachusetts: The MIT Press, 2008. p. 92-100 (Software Studies).

TONSKI, Jacob. **Artist's statement**. Prix Ars Electronica 2014. 2014.
URL: prix2014.aec.at/prixwinner/13363/

TURKLE, Sherry (1984). Video Games and Computer Holding Power. In: WARDRIP-FRUIN, Noah; MONTFORT, Nick Montfort. **The New Media Reader**, Volume 1. MIT Press. 2003. 34, p. 500-514.

VENTURELLI, Suzete; MACIEL, Mario Luiz Belcino . Gamearte: uma poética de interação. **Revista FAMECOS**, v. 1, p. 4, 2004.

VIOLA, Bill (1995). Will There Be Condominiums in Data Space? In: WARDRIP-FRUIN, Noah; MONTFORT, Nick Montfort. **The New Media Reader**, Volume 1. MIT Press. 2003. 31, p. 463-470.

WARD, Adrian. How I drew one of my pictures. In: **Auto-Illustrator** (User Guide). London: Signwave. 2002.

WARDRIP-FRUIN, Noah. **Expressive Processing**: On Process-Intensive Literature and Digital Media. Tese (Doutorado), Special Graduate Studies, Brown University, Providence, Rhode Island. Mai. 2006.

Vídeo e Filme

ALAEJOS, Raúl (diretor). **Hello World! Processing**. 2013. Duração: 41min.
URL: vimeo.com/60735314

APTED, Michael (1997). **Inspirations**. Documentário sobre criatividade (seleção de trechos com o músico David Bowie; youtube/B0WIElover). 2011. Duração: 20min21s.
URL: youtu.be/pDmb_aR_OnY

KAY, Alan (1987). **Doing with images makes symbols**: Communicating with computers. Registro em vídeo (youtube/Kenneth Friedman). The Distinguished Lecture Series: Industry Leaders in Computer Science. 2015. Duração: 1h37m44s.
URL: youtu.be/p2LZLYcu_JY

MATEAS, Michael. **Palestra inaugural** (keynote). Registro em vídeo (vimeo/UCSC Games & Playable Media). Inventing the Future of Games (IFOG 2011), Center for Games and Playable Media (UCSC). Abr. 2011. Duração: 45min59s..
URL: www.vimeo.com/22969930

ROHRER, Jason; CRAWFORD, Chris. **Into the night with...** (série de TV), Jul. 2009. Duração: 51m33s.
URL: youtu.be/zA_O_dSD3-Q

WARDRIP-FRUIN, Noah. **Software Studies Retrospective**. Registro em vídeo (vimeo/Media, Culture, Communication). Media and Literature Program, New York University. Abr. 2014. Duração: 33min16s.
URL: vimeo.com/93740183

Outras Referências

Jogos Digitais

Angry Birds (Rovio, 2009-)

Bastion (Supergiant Games, 2011)

Beginner's Guide, The (Davey Wreden, 2015)

Broken Sword (Revolution Software, 1996)

Call of Duty (Activision)

Civilization (MicroProse, 1991-)

Command & Conquer (Electronic Arts, 1995-)

Cozinheiro das Almas (Poéticas Digitais, 2006)

Crash Bandicoot (Naughty Dog, 1996)

Crush the Castle (Armor Games, 2009)

Cutthroat Capitalism (WIRED, 2009)

Dear Esther (The Chinese Room, 2012)

Dwarf Fortress (Bay 12 Games, 2006-)

Dys4ia (Anna Anthropy, 2012)

Eliza/Doctor (Joseph Weizenbaum - MIT, anos 60)

Everybody's Gone to the Rapture (The Chinese Room, 2015)

Façade (Mateas e Stern, 2005) URL: `www.interactivestory.net`

Flower (Thatgamecompany, 2009)

Gone Home (The Fullbright Company, 2013)

Grand Theft Auto (Rockstar Games) - série

Heavy Rain (Quantic Dream, 2010)

Incidente em Varginha (Perceptum, 1998)

Journey (Thatgamecompany, 2012)

Leisure Suit Larry in the Land of the Lounge Lizards (Al Lowe, 1987)

Minecraft (Mojang, 2009)

No Man's Sky (Hello Games, 2016 - previsão)

Oregon Trail, The (MECC, 1971)

Pac-Man (Namco, 1980)

Papers, Please (Lucas Pope, 2013)

POD: Planet of Death (Ubisoft, 1997)

Prom Week (Expressive Intelligence Studio, 2012). URL: games.soe.ucsc.edu/project/prom-week

RIOT (em desenvolvimento). URL: www.riotgame.org

Secret of Monkey Island, The (Lucasfilm Games, 1990)

Sim City (Will Wright, 1989-)

Sims, The (Maxis, 2000-)

Slower Speed of Light, A (MIT Game Lab, 2012)

Stanley Parable, The (Davey Wreden, 2011)

Sunset (Tale of Tales, 2015)

Super Mario Bros. (Nintendo, 1985)

Tale-Spin (James Meehan, 1976)

Tetris (Alexey Pajitnov, 1984)

Theme Park (Bullfrog, 1994)

Tomb Raider (Eidos Interactive, 1996)

Obras e Produções Digitais

Aaron (Harold Cohen, 1973-). URL: www.aaronshome.com

Algorithmic Poetry for a Three Story Wall (Roman Verostko, 2011)

Balance From Within (Jacob Tonski, 2012)

Biophilia (Björk, 2011)

end of the virtual world, the (Robert Overweg, 2010)

Evolved Virtual Creatures (Karl Sims, 1994)

Sand As Medium (Jean-Pierre Hébert, 1997-2002)

Exposições e Festivais

Cybernetic Serendipity: a Documentation. Instituto de Arte Contemporânea, Londres. Out. 2014.

Gameplay. Itaú Cultural, São Paulo. 2009.

Festival Games for Change América Latina. V. Nov. 2015.

FILE (Festival Internacional de Linguagem Eletrônica). 2015.

Procedural Generation Jam. 2015.
URL: www.procjam.com

Ferramentas, Aplicativos, Linguagens e Tecnologias

Adventure Game Studio (Chris Jones, 1997)

Adventurezator (Pigasus, 2015) (inclui jogo). URL: `adventurezator.com`

Anyflo (Michel Bret, 1984)

Automavision (Lars Von Trier).

Blender (1998)

Choreographic Language Agent (OpenEnded, 2010)

Construct (Scirra, 2007)

Dynabook (Alan Kay, 1977)

Flash (Macromedia, 1996)

Flixel (Adam Saltsman, 2009)

GameMaker (YoYo Games, 1999-)

Game-o-Matic (Expressive Intelligence Studio/UCSC). URL: `cartoonist.soe.ucsc.edu`

Garry Kitchen's GameMaker (Activision, 1985)

HTML5 (W3C & WHATWG, 2014)

Klik & Play (Clickteam, 1994)

Logo (Wally Feurzeig e Seymour Papert, 1967)

MAXScript (3D Studio Max, Autodesk)

MissionMaker (LKL)

OpenRelativity - URL: `gamelab.mit.edu/research/openrelativity`

Photoshop (Adobe)

Processing (Ben Fry e Casey Reas, 2001-)

Pure Data (Miller Puckette, 1996)

Puzzle-Dice (MIT Game Lab)

Scratch (MIT, 2008). URL: `scratch.mit.edu`

Sketchpad (Ivan Sutherland, 1963)

Twine (Chris Klimas, 2009-)

Unity (2005-)

Verbasizer (David Bowie e Ty Robert, 1995).

Vídeos e Filmes

Andrew Smith (gameplay video; youtube/Andrew Burn, 2015)
URL: youtu.be/D-TV1S8lV4M

Beowulf (Robert Zemeckis, 2007)

Beowulf Story: Embellishing Beowulf (youtube/Andrew Burn; Theo Bryer, 2015)
URL: youtu.be/ARdgI0QZSKI

Boss of It All, The (Lars Von Trier, 2006)

CLOUDS (James George e Jonathan Minard, 2014)
URL: cloudsdocumentary.com

GTA 5 (youtube/VanossGaming, 2013)
URL: youtu.be/FvEb2Em9hLs

My Trip to Liberty City (Jim Munroe, 2003; youtube/nomediakings)
URL: youtu.be/fxpDHiH5PKk

Oregon Trail Gameplay, The (youtube/Shortmandesigner, 2010).
URL: youtu.be/FfbGEP087HM

Pac-Man original 1980 by Namco (Puck-Man). M.A.M.E emulated (youtube/Egon H, 2010)
URL: www.youtube.com/watch?v=glYGt7mFCaQ

Silêncio dos Inocentes (Jonathan Demme, 1991)

Six Games (gameplay video compilation; youtube/Andrew Burn; Daniel Ferreira, 2015)
URL: youtu.be/uYQEZi3WcIA

Terminal, The (Steven Spielberg, 2004)

Sites

compart (center of excellence digital art)
URL: dada.compart-bremen.de

DAM - Digital Art Museum
URL: www.dam.org

Edge
URL: www.edge-online.com

Gamasutra (The Art & Business of Making Games)
URL: www.gamasutra.com

Game/Show (PBS)
URL: www.youtube.com/user/pbsgameshow

Indie Statik
URL: indiestatik.com

Kill Screen
URL: killscreendaily.com

MIT video. Arquivo de vídeos (Massachusetts Institute of Technology).
URL: video.mit.edu

Open Processing
URL: www.openprocessing.org

State of Play
URL: www.stateofplayshow.com

Grupos, Organizações e Laboratórios

Algorists, The.
URL: www.algorists.org

Center of Games & Playable Media (UC Santa Cruz). Arquivo de vídeos.
URL: vimeo.com/playableucsc

DARE Collaborative
URL: darecollaborative.net

Expressive Intelligence Studio, The.Universidade da California (Santa Cruz).
URL: eis-blog.ucsc.edu

Games for Change (América Latina)
URL: www.gamesforchange.org.br

MIT Game Lab.
URL: gamelab.mit.edu

Mode: Multimodal Methodologies for Researching Digital Data and Environments (LKL IOE UCL).
URL: mode.ioe.ac.uk

Poéticas Digitais (ECA/USP)
URL: poeticasdigitais.net

Processing SP (Comunidade)
URL: www.facebook.com/processingsp

SCIArts
URL: sciarts.org.br

Software Studies Initiative
URL: lab.softwarestudies.com

Livros

Beowulf (Gareth Hinds, 2008) - arte sequencial

Beowulf (Ary Gonzalez Galvão, 1992) - tradução em Português

Beowulf (Frances B. Gummere, 1909) - tradução em Inglês moderno

Caves of Steel, The (Isaac Asimov, 1974)

Cent mille milliards de poèmes (Raymond Queneau, 1961)

Choose Your Own Adventure (Bantam Books, 1979-1998)

Dune (Frank Herbert, 1979)

Dungeons & Dragons (Gary Gygax e Dave Arneson, 1974) - RPG

Glossário

Agência: potencial de interferência que um sistema digital dá ao interator (extensão ao conceito da **Interatividade**).

Análise Multimodal: ver **Multimodalidade.**

Algoritmo: série finita de instruções para a realização de uma determinada operação. Em meio digital, refere-se às instruções que compõem um programa de computador.

Autoria Procedural: ver **Proceduralidade.**

Criatividade: qualidade ou capacidade de gerar ideias originais, de dar origem, inventar ou imaginar algo.

Engine: parte do programa que contém funções, estruturas e elementos genéricos fundamentais do sistema.

Expressividade: qualidade ou capacidade de expressar uma ideia ou significado de uma maneira criativa e particular, não necessariamente artística. Nesta pesquisa, empegamos o termo principalmente em referência à forma expressiva (em oposição a aditiva; p. 25). Também opõe-se a fins práticos ou utilitários.

Expressividade Procedural: ver **Proceduralidade.**

Intensidade procedural: idem**.**

Interatividade: nesta pesquisa o termo é empregado em referência ao uso do computador, de um aplicativo ou elemento específico por parte do usuário ou público (não necessariamente implica em **Agência**).

Jogos Digitais: (videogame) aplicativo cujo elemento predominante é a jogabilidade.

Linguagem de Programação: notação particular para a criação de algoritmos, de forma que possam ser compilados e executados em um computador. Oferece uma camada de abstração entre o programador a linguagem de máquina (ou a uma linguagem de nível mais baixo).

Literacia Procedural: ver **Proceduralidade.**

Mídias Digitais: (Meio Digital, computador): sistema composto de hardware e software, programável e autônomo.

Modelo: representação conceitual (ou abstração) de um objeto, processo ou informação.

Modo: ver **Multimodalidade.**

Multimodalidade: área ou metodologia de análise para o estudo da linguagem, com foco na construção de significado. **Modo** refere-se ao conceito da linguagem em um contexto mais amplo. Recurso **Multimodal** (ou semiótico) refere-se a qualquer elemento psicológico ou tecnológico que pode ser usado para comunicar, representar ou expressar uma ideia (no contexto desta tese, "recurso" pode se referir à noção mais ampla de um conteúdo, ferramenta ou funcionalidade específica) - pode ser estático ou procedural.

Narrativa: nesta pesquisa, empregamos o termo em referência à noção de uma estrutura dinâmica de eventos, relacionamentos e ideias conectadas (isto é, não se refere necessariamente a uma sequência linear e fixa de acontecimentos ao longo do tempo).

Nível de Abstração: (ou Grau) em meio digital, tratam-se de camadas dispostas entre os mecanismos mais fundamentais do computador e o usuário ou programador (exemplos de camadas de abstração são linguagens de programação, aplicativos e sistemas operacionais).

Proceduralidade: caráter programável e autônomo do meio digital (também pode se referir à qualidade algorítmica). Opõe-se a estático, fixo ou linear. **Literacia** procedural se refere à capacidade de produzir, interpretar e se expressar a partir da proceduralidade. **Autoria** procedural refere-se ao ato de criar ou manipular conteúdo procedural. **Intensidade** procedural trata-se da proporção de conteúdo procedural em um determinado sistema (em contraste ao estático). **Expressividade** procedural consiste na autoria procedural realizada com fins expressivos.

Pseudocódigo: representação de um algoritmo em um nível de abstração mais alto, legível por um ser humano.

Recurso Multimodal: ver **Multimodalidade**

Tradicional: refere-se a formas expressivas que não são fundamentalmente baseadas em mídias digitais (também refere-se a meios não digitais).

Lista de Figuras

1 Tabela: Atividades criativas em mídias digitais 28
2 Diagrama: Autoria procedural: etapas do processo criativo 30
3 Autorretrato (Vera Molnar, 1939) 40
4 Variação na série Hypertransformations (1954) 40
5 Structure de Quadrilatères (1988) 40
6 Façade (Mateas e Stern, 2005) 43
7 Angry Birds (Rovio, 2009-) 49
8 Pac-Man (Namco, 1980) 49
9 Diagrama: tipos de mídia (WARDRIP-FRUIN, 2006) 58
10 Diagrama: modelo do meio digital (WARDRIP-FRUIN, 2006) 61
11 Ilustração representando mulher no século XIX (esq.), desenho de Robert Crumb para texto de Bukowski, cena de Amélie Poulain e tela do jogo The Sims 62
12 Sim City (esq.), My Trip to Liberty City (GTA) e Tetris 69
13 Cutthroat Capitalism (esq.) e RIOT 72
14 Beowulf: primeira página do manuscrito original (dir.: detalhe) 92
15 MissionMaker: interface 98
16 Beowulf (Robert Zemeckis, 2007) 101
17 (Gareth Hinds, 2008) 101
18 Oficina: improviso em dramaturgia 102
19 Oficina: criação de jogos com MissionMaker 102
20 O Servo Fiel (Andrew Smith) 107
21 Gone Home (2013) 109
22 Dear Esther (2012) 109
23 A Caveira de Æschere (Alice Hawkins e Soraya Shabaan) 115
24 Voltando Atrás (Harriet Piercy) 119
25 Vingando Æschere (Katie Smith) 125
26 Através do Pântano (Laura Scott) 127
27 A Recompensa (Niamh Hickey) 133
28 Comer, Beber, Dormir (Tracey Matthews) 137
29 Passo 1: seleção da porta 143
30 Passos 2/3: função "abrir" 144
31 Passo 4: tipo de ativador 144
32 Regra completa 144

33 Rega em funcionamento no jogo 144
34 The Oregon Trail (MECC, 1971) 160
35 Witcher 3 (CD Projekt RED, 2015) 161
36 Grand Theft Auto 5 (Rockstar) 161
37 Diagrama: níveis de abstração (MissionMaker) 171
38 Modelos de representação: MissionMaker (esq.), Pac-Man e Calc. 173
39 Modelos de apresentação: Psychonauts (esq.), StarCraft e FM2015. 174
40 No Man's Sky (esq.) e capas de livros de ficção científica 177
41 A Slower Speed of Light (esq.) e Return of the Obra Din 178
42 Diagrama: recursos modais no MissionMaker 185
43 Diagrama: recursos modais agrupados por linguagem 190
44 Oficina: atividade de desenho 193
45 Oficina: escrita criativa (apresentação) 194
46 Oficina: Artes Dramáticas e improviso 196
47 MissionMaker2: modo de edição de localidades 209
48 Criando objetos: porta (topo) e volume de ativação. 209
49 MissionMaker2: personagens de Beowulf 210
50 Criação de uma regra para abrir a porta 211
51 Regra com múltiplas condicionais 213
52 Comportamento resultante da regra 214
53 Regra com múltiplas ações: regra (esq.) e resultado 215
54 Série "the end of the virtual world" (Robert Overweg, 2010) 237
55 Angry Birds (fase da torre) 239
56 Leisure Suit Larry: inconsistência na ação "Pegar" 242

Documento criado em LyX (www.lyx.org), uma interface gráfica para o sistema LaTeX.
Fonte: Latin Modern Sans

Apoio:
CAPES - Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico

Blog desta pesquisa: www.7luas.com.br/pesquisa