

PONTIFÍCIA UNIVERSIDADE CATÓLICA  
DO RIO DE JANEIRO



**Francisco Oliveira de Queiroz**

**A CIÊNCIA COMO JOGO:  
Abordagens lúdicas ao design de interfaces  
para software científico**

**Tese de Doutorado**

Tese apresentada ao Programa de Pós-graduação em Design da PUC-Rio como requisito parcial para obtenção do grau de Doutor em Design.

Orientadora: Profa. Rejane Spitz

Rio de Janeiro,  
Novembro de 2018



**Francisco Oliveira de Queiroz**

**A CIÊNCIA COMO JOGO:  
Abordagens lúdicas ao design de interfaces  
para software científico**

Tese apresentada ao Programa de Pós-Graduação  
em Design da PUC-Rio como requisito parcial para  
obtenção do grau de Doutor em Design. Aprovada pela  
Comissão Examinadora abaixo assinada.

**Profª Rejane Spitz**

Orientador

Departamento de Artes & Design – PUC-Rio

**Prof. Jorge Roberto Lopes dos Santos**

Departamento de Artes & Design – PUC-Rio

**Prof. João de Sá Bonelli**

Departamento de Artes & Design – PUC-Rio

**Prof. Antonio Augusto Passos Videira**

Universidade do Estado do Rio de Janeiro

**Prof. Delmar Galisi Domingues**

Universidade Anhembi Morumbi- UAM

**Profª Monah Winograd**

Coordenadora Setorial do Centro de Teologia  
e Ciências Humanas – PUC-Rio

Rio de Janeiro, 29 de novembro de 2018

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, da autora e do orientador.

### **Francisco Oliveira de Queiroz**

Graduou-se em Comunicação Social na Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio) em 2000. Obteve grau de Mestre em Design de Jogos Digitais pela University for the Creative Arts (UCA) em 2005. É professor no curso de graduação de Design da PUC-Rio desde 2007. É professor visitante na University of Leeds, Reino Unido, desde outubro de 2018. Atua como Designer para o Instituto Tecgraf / PUC-Rio desde 2007.

Queiroz, Francisco Oliveira de

A ciência como jogo : abordagens lúdicas ao design de interfaces para software científico / Francisco Oliveira de Queiroz ; orientadora: Rejane Spitz. – 2018.

276 f. : il. color. ; 30 cm

Tese (doutorado)–Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Artes e Design, 2018.

Inclui bibliografia

1. Artes e Design – Teses. 2. Software científico. 3. Gamificação. 4. Interação humano-computador. 5. Design participativo. 6. Filosofia da ciência. I. Spitz, Rejane. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Artes e Design. III. Título.

COD : 700

Para Ana, Alice e Olivia,  
sempre



## Agradecimentos

À minha orientadora Professora Rejane Spitz, pelo encorajamento, conselhos e (des)orientações.

À PUC-Rio, pelos auxílios concedidos, sem os quais este trabalho não poderia ser realizado.

Ao Instituto Tecgraf / PUC-Rio, pelo contínuo apoio e parceria.

Ao CCCI / PUC-Rio, pela oportunidade oferecida.

À Dra. Maria Lonsdale, pela supervisão e ajuda.

À School of Design, University of Leeds, Reino Unido.

Aos meus colegas de PUC-Rio, Instituto Tecgraf, Departamento de Artes & Design e Laboratório de Arte Eletrônica.

Ao Professor Marcelo Gattass, e Eduardo Thadeu Corseuil, do Instituto Tecgraf.

Aos membros da banca de qualificação, pelas observações.

Ao Professor Antônio Augusto Passos Videira, pelas sugestões e trocas de ideias.

Aos meus pais, família e amigos, pelo apoio.

À Ana, Alice e Olivia, por tudo.

## Resumo

Queiroz, Francisco Oliveira de; Spitz, Rejane. **A ciência como jogo: abordagens lúdicas ao design de interfaces para software científico**. Rio de Janeiro, 2018. 276p. Tese de Doutorado – Departamento de Artes & Design, Pontifícia Universidade Católica do Rio de Janeiro.

De grande importância para pesquisa científica, o software científico se mostra, por vezes, desafiador no que se refere ao design de interfaces com o usuário. Por outro lado, a gamificação se oferece como potencial solução em questões de experiência com o usuário e interatividade. Nesta tese, desenvolvemos um método para a gamificação de software científico voltado para processos de design colaborativo, orientado por um modelo baseado em jogo, informado por aspectos de uso e desenvolvimento deste tipo de software e, também, por aspectos similares ao jogo na prática científica.

## Palavras-Chave

Software científico; ciência computacional; gamificação; design de jogos; usabilidade; interação humano-computador; design colaborativo; design participativo; co-design; pesquisa-ação; filosofia da ciência, história da ciência

## Abstract

Queiroz, Francisco Oliveira de; Spitz, Rejane (Advisor). **Science as a game: playful approaches to scientific software interface design**. Rio de Janeiro, 2018. 276p. Tese de Doutorado – Departamento de Artes & Design, Pontifícia Universidade Católica do Rio de Janeiro.

Vastly important to scientific research, scientific software is often challenging regarding the design of its user interfaces. On the other hand, gamification is seen as a potential solution in matters of user experience and interactivity. In this thesis, we develop a method for the gamification of scientific software aimed at collaborative design processes, guided by a game-based model, informed by aspects of use and development of that type of software and, also by game-like aspects of scientific practice.

## Keywords

Scientific software; computational science; gamification; games design; usability; human-computer interaction; collaborative design; participative design; co-design; action-research; philosophy of science, history of science.

# Sumário

1	Introdução	15
2	Software científico, design de interfaces e gamificação	20
2.1	Introdução ao software científico	22
2.1.1	Exigências da prática científica e acadêmica	23
2.1.2	Aspectos de cunho colaborativo	24
2.1.3	Plataformas de uso e linguagens de programação	29
2.1.4	Etapas do trabalho em software científico	31
2.2	Questões em design de interfaces e usabilidade em software científico	32
2.2.1	Quanto às alternativas às interfaces gráficas	34
2.2.2	Quanto ao uso de linguagens de programação por usuários	35
2.2.3	Quanto às etapas do trabalho científico	36
2.2.4	Quanto às exigências da prática científica e profissional	43
2.2.5	Quanto às plataformas de uso e desenvolvimento	44
2.2.6	Avanços no design de interfaces para software científico	47
2.3	Gamificação, jogo e software científico	52
2.3.1	Gamificação	53
2.3.2	Gamificação da ciência	56
2.3.3	Gamificação de software científico	59
2.4	Conclusão do capítulo	63
2.4.1	Características de uso e desenvolvimento	64
2.4.2	Design de usabilidade e gamificação	65
2.4.3	Diretrizes para a elaboração do método: do software e seu desenvolvimento	67
3	Jogo e ciência	68
3.1	Usos prévios do jogo como modelo para a ciência	71
3.2	Uma definição clássica como base para o modelo	73

3.3 Regras e realidades	75
3.4 Medidas e valores	94
3.5 Esforço e recompensa	105
3.6 Conclusão do capítulo	131
3.6.1 Aspectos da pesquisa e trabalho em ciência	131
3.6.2 Questões quanto à aplicação de uma visão sistêmica	133
3.6.3 Diretrizes para a elaboração do método: da ciência e suas dinâmicas	134
3.6.4 Jogo, ciência e software	135
4 Experiências em design de gamificação para software científico	138
4.1 Migrando padrões de interatividade: dos video games para a simulação integrada em engenharia de óleo e gás	139
4.1.1 Metodologia	140
4.1.2 Desenvolvimento	141
4.1.2.1 Primeira fase: definição do problema	141
4.1.2.2 Segunda fase: produção e recepção a protótipos iniciais	145
4.1.2.3 Terceira fase: pesquisa a jogos e protótipos de alta fidelidade	148
4.1.2.4 Encerramento da investigação e reação ao protótipo	158
4.1.3 Discussão	158
4.1.3.1 Quanto a desafios peculiares do software científico	159
4.1.3.2 Quanto à pesquisa e design de gamificação	161
4.1.4 Conclusão	163
4.2 Co-design de interfaces baseado em jogos: casos em matemática, física e engenharia	164
4.2.1 Metodologia	166
4.2.1.1 Materiais e ferramentas	167
4.2.1.2 Procedimento	169
4.2.1.3 Coleta de dados	171
4.2.1.4 Acompanhamento	171
4.2.2 Estudo de caso I: plotagem de gráficos em física de matéria condensada	171
4.2.2.1 Problema	172
4.2.2.2 Exploração	172

4.2.2.3 Rascunho e diretrizes	173
4.2.2.4 Acompanhamento	174
4.2.3 Estudo de caso II: visualização em matemática física	174
4.2.3.1 Problema	175
4.2.3.2 Exploração	175
4.2.3.3 Rascunho e diretrizes	177
4.2.3.4 Acompanhamento	178
4.2.4 Estudo de caso III: análise de dados de fluxo em tomografia de processo	178
4.2.4.1 Problema	179
4.2.4.2 Exploração	179
4.2.4.3 Rascunho e diretrizes	181
4.2.4.4 Acompanhamento	182
4.2.5 Discussão	183
4.2.5.1 Quanto a relação entre jogo e ciência	183
4.2.5.2 Quanto ao método e suas ferramentas	186
4.2.6 Conclusão	188
4.3 Análise comparativa e considerações	189
5 Conclusão e considerações finais	192
5.1 Abordagens à gamificação	192
5.2 Contribuições	194
5.3 Limitações	195
5.4 Pesquisas futuras	196
5.5 Considerações finais	196
6 Referências Bibliográficas	199
7 Anexos	232
7.1 Termos de Consentimento	232
7.2 Atas das reuniões da pesquisa-ação	240
7.3 Conjunto de cartas elaborado para sessões de co-design	268

## Lista de Figuras

Figura 1 - Relações entre Capítulos, Questões e Objetivos	17
Figura 2 - Relações entre capítulos e métodos de pesquisa	18
Figura 3 - Interface do LabVIEW	36
Figura 4 - Interface para diagramação de modelo (Bunus, 2006).	37
Figura 5 - Interface para bases de conhecimento científico (Keller et al., 1994).	37
Figura 6 - Interface para análise semântica de código (Stewart, 2001).	38
Figura 7 - Workflows no Taverna Workbench	39
Figura 8 - MeVisLab (Bitter et al., 2007)	39
Figura 9 - GenAMap (Curtis et al., 2011).	40
Figura 10 - Petri net Toolbox (Julvez et al., 2014)	42
Figura 11 - Biok (Letondal & Zdun, 2003)	42
Figura 12 - Simulação em RV diagramada via web (Kovalchuk, 2012).	46
Figura 13 - Conquistas e sistemas de pontos do ResearchGate	57
Figura 14 – Foldit, Gamificação de ciência cidadã	58
Figura 15 - BioStar (biostar.usegalaxy.org)	60
Figura 16 - Missão em <i>GamiCAD</i> (Li et al., 2012)	61
Figura 17 - Virtual Astronaut (Wang et al., 2012)	62
Figura 18 - Visão sistêmica das definições de Juul (Queiroz et al., submetido)	74
Figura 19 - Diagrama de livro didático (Kuhn, 1977)	100
Figura 20 - Folhas - truth-to-nature (Daston & Galison, 2010)	108
Figura 21 - Folha - objetividade mecânica (Daston & Galison., 2010)	109
Figura 22 - Rede postal - objetividade estrutural (Daston & Galison, 2010)	110
Figura 23 - Radiografia, julgamento treinado (Daston & Galison., 2010)	111
Figura 24 - Simulação de campos de vorticidade (Daston & Galison., 2010).	112

Figura 25 - Átomos manipulados para soletrar “IBM” (Daston & Galison, 2010)	112
Figura 26 - Planetas e órbitas no TRAPPIST-1 (Gillon et al., 2017)	114
Figura 27 - Visão ilustrativa do TRAPPIST-1 (Landau, 2017)	115
Figura 28 - Modelo híbrido jogo-ciência-software.	137
Figura 29 - Visão geral da pesquisa (Queiroz et al 2016)	140
Figura 30 - Captura de tela do SiVIEP. (Queiroz et al., 2016):	142
Figura 31 - Timeline (acima) e Graph (abaixo) (Queiroz et al., 2016)	143
Figura 32 - Protótipo de baixa fidelidade (Queiroz et al., 2016)	146
Figura 33 - Timeline e instruções (Queiroz et al., 2016)	147
Figura 34 - Cities in Motion 2 (acima) e Braid (abaixo)	149
Figura 35: Viva Piñata (esquerda) e Mass Effect (direita)	150
Figura 36 - Sim City 4 (acima) e proposta (abaixo) (Queiroz et al. 2016)	151
Figura 37 - Tela inicial do protótipo de alta fidelidade. (Queiroz et al. 2016)	152
Figura 38 - Menu circular ativado (Queiroz et al. 2016)	153
Figura 39 - Importação de objetos georreferenciados (Queiroz et al. 2016)	154
Figura 40 - Menu circular para carregamento de objetos (Queiroz et al., 2016)	154
Figura 41 - The Elder Scrolls V: Skyrim – menu de seleção de armas	155
Figura 42 - Sim City Build It (esquerda) e Farmville (direita)	156
Figura 43 - Translação adaptável de objetos (Queiroz et al., 2016)	157
Figura 44 - Mapa de funcionalidades do SiVIEP (Queiroz et al., 2016)	157
Figura 45 - SiVIEP (esq.) e Cities in Motion 2 (dir.) (Queiroz et al., 2016)	161
Figura 46 - Cartas diretrizes. Traduzido de Queiroz et al., submetido	167
Figura 47 - Visão sistêmica. Adaptado de Queiroz et al., submetido	168
Figura 48 - Amostra de cartas ilustradas. (Queiroz et al., submetido)	169
Figura 49 - Diagrama do procedimento (Queiroz et al., submetido)	170
Figura 50 - Cartas e comentários de P1 (Queiroz et al., submetido)	172
Figura 51 - Rascunho de P1 (Queiroz et al., submetido)	173
Figura 52 - Comentários às diretrizes - P1 (Queiroz et al., submetido)	173



Figura 53 - Protótipo feito sobre capturas de tela (Queiroz et al., submetido)	174
Figura 54 - Cartas e comentários de P2 (Queiroz et al., submetido)	176
Figura 55 - Rascunho de P2. (Queiroz et al., submetido)	177
Figura 56 - Comentários às diretrizes – P2 (Queiroz et al., submetido)	178
Figura 57 - Cartas e comentários de P3 (Queiroz et al., submetido)	180
Figura 58 - Rascunho de P3 (Queiroz et al., submetido)	181
Figura 59 - Captura de tela obtida de P3. (Queiroz et al., submetido)	181
Figura 60 - Comentários às diretrizes – P3 (Queiroz et al., submetido)	182
Figura 61 - Protótipo baseado em capturas de tela (Queiroz et al., submetido)	183

## Lista de Tabelas

Tabela 1 - Resultados de buscas atualizadas no dia 4 de dezembro de 2015.	21
Tabela 2 - Amostra de manifestações dos elementos do jogo na ciência e paralelos no software científico	136

# 1

## Introdução

Na medida em que se dedica à ciência normal, o pesquisador é um solucionador de quebra-cabeças (...) assemelha-se mais ao enxadrista que, confrontado com um problema estabelecido e tendo à sua frente (física ou mentalmente) o tabuleiro, tenta vários movimentos alternativos na busca de uma solução.

Thomas S. Kuhn

Há mais de quatro décadas, jogos eletrônicos e software científico vêm promovendo intensas transformações em suas áreas de uso, estabelecendo novos paradigmas na indústria cultural e prática científica, respectivamente. No entanto, atividades lúdicas e científicas realizadas em ambiente digital diferem perceptivelmente no que diz respeito à qualidade das interações entre usuários e produtos. Por um lado, nota-se a oportunidade, quando não necessidade, de melhorias na interação que o software científico oferece a seus usuários. Por outro lado, percebe-se que jogos digitais lançam mão de recursos capazes de proporcionar imersão, motivação e aprendizado. Observa-se ainda que o design de jogos vem sendo aplicado em áreas externas à própria produção de jogos – inclusive em ciência e tecnologia – como forma de estimular o uso, aprendizado e engajamento de usuários. À esta tendência, convencionou-se referir-se como “gamificação”. Parece vantajoso e natural, portanto, transpor elementos de design de jogos para o software científico. À luz destas questões, esta pesquisa apresenta o seguinte problema: *como organizar processos de gamificação que atendam às especificidades e características do software científico e suas interfaces?*

O objetivo geral desta tese — investigar estratégias de gamificação de software científico — foi alcançado através dos seguintes objetivos específicos: (1) identificar e descrever peculiaridades no desenvolvimento e uso de software científico; (2) discutir paralelos entre jogo e ciência; (3) desenvolver métodos para o design de interfaces de software científico baseados nas características identificadas.

Para atingir os objetivos propostos, esta pesquisa buscou responder às seguintes questões norteadoras: (1) quais os processos de design mais apropriados ao software científico? (2) que paralelos podem ser traçados entre jogos e o trabalho científico — particularmente quando realizado através de software? (3) como a dinâmica dos jogos pode ser transposta ao software científico através do design?

A investigação de estratégias para aplicação de design de jogos digitais em software científico se mostrou relevante sob vários aspectos. Em primeiro lugar, em caráter pessoal, o tema reúne duas atividades nas quais o autor concentra sua vida profissional e carreira docente. Este duplo envolvimento foi um estímulo na busca por uma aproximação entre os dois campos. Em segundo lugar, quanto ao objeto de estudo, buscou-se o aperfeiçoamento de métodos de desenvolvimento de software científico. Em terceiro lugar, para o profissional de design, esta pesquisa buscou ampliar o campo de atuação do designer de mídias digitais e, também, seu repertório de soluções. Finalmente, para os campos de estudos teóricos abordados, a pesquisa realizada traz contribuições ao debate corrente quanto à aplicação de design de jogos em contextos profissionais voltados à ciência. Portanto, esta tese oferece como contribuição original um método para o design de interfaces de software científico e, também, um modelo comparativo entre ciência, jogo e software científico, desenvolvido para argumentação e elaboração do método proposto.

A tese está estruturada da seguinte forma: O capítulo 2, *Software científico, design de interfaces e gamificação*, apresenta o objeto de pesquisa — definições, relevância, principais características, desafios e métodos de desenvolvimento, e formas de uso. O capítulo apresenta também um panorama do estado da arte em design de usabilidade e gamificação do software científico. Ao final do capítulo, propõe-se uma abordagem à gamificação baseada em elementos que compõem uma definição clássica de jogo. A partir desses elementos, o capítulo 3, *A ciência como jogo*, discute paralelos entre estes dois campos de atividade, buscando mostrar manifestações de aspectos lúdicos na prática científica. Tal discussão é baseada, em

sua maior parte, em perspectivas de historiadores e filósofos da ciência. Finalmente, o capítulo 4, *Experiências em design de gamificação para software científico*, relata duas pesquisas de campo distintas, realizadas separadamente, primeiro no Instituto Tecgraf / PUC-Rio e, posteriormente, na Universidade de Leeds, no Reino Unido – ambas baseadas em abordagens voltadas para o design participativo. O capítulo apresenta um método para o co-design de interfaces de software científico, fruto da segunda pesquisa de campo. Tal método é baseado nas informações e características apresentadas nos capítulos 2 e 3, levando em consideração características técnicas e teóricas discutidas ao longo da tese. A seção *Conclusão e considerações finais* encerra a tese, ponderando e refletindo quanto ao conteúdo apresentado, avaliando seus resultados e sugerindo possíveis desdobramentos da pesquisa.

As relações entre os capítulos da tese, objetivos específicos e questões norteadoras estão ilustradas na Figura 1.

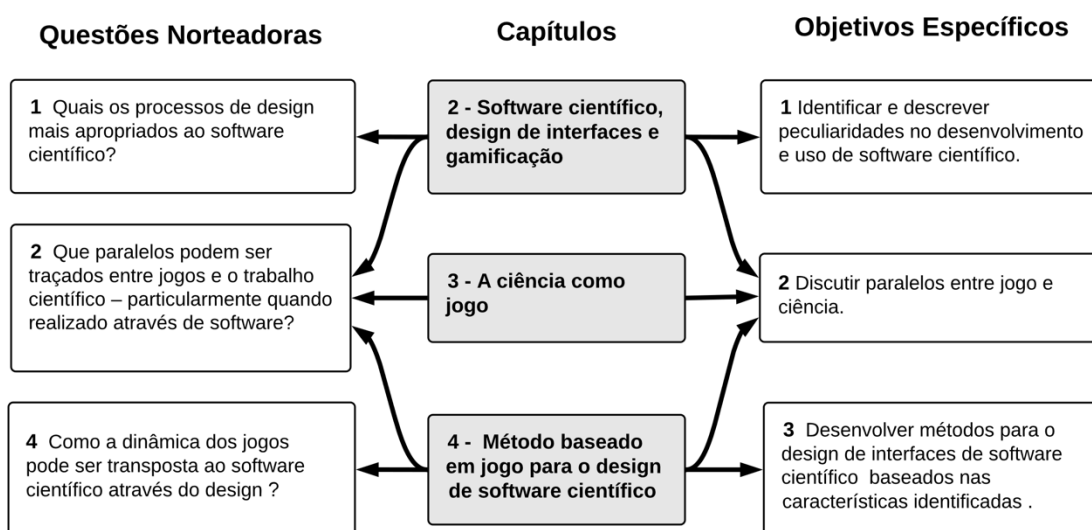


Figura 1 - Relações entre Capítulos, Questões e Objetivos

No que se refere a aspectos metodológicos, esta pesquisa evidencia seu caráter exploratório lançando mão de múltiplas abordagens a fim de adequar-se tanto à complexidade quanto à multidisciplinaridade exigidas pelo tema. Desta forma, cada capítulo adota procedimentos metodológicos apropriados aos tópicos abordados e aos objetivos aos quais se dedicam. Em primeiro lugar, partiu-se de uma revisão de literatura semi-sistemática, realizada em pesquisa a bancos de dados de artigos científicos, complementada por artigos selecionados. Tal revisão abarcou

os temas do desenvolvimento, uso, usabilidade e gamificação do software científico, e serviu como base do capítulo 2, *Software científico, design de interfaces e gamificação*. Em segundo lugar, realizou-se uma pesquisa bibliográfica a autores e obras de referência em filosofia e história da ciência para fundamentação teórica – base do capítulo 3, *A ciência como Jogo*. Finalmente, pesquisas de campo baseadas em métodos de design participativo informaram o capítulo 4, *Método baseado em jogo para o design de software científico*. Cada capítulo inclui informações detalhadas a respeito de seus respectivos métodos de pesquisa. As relações entre os métodos de pesquisa e os capítulos da tese estão ilustradas na Figura 2.

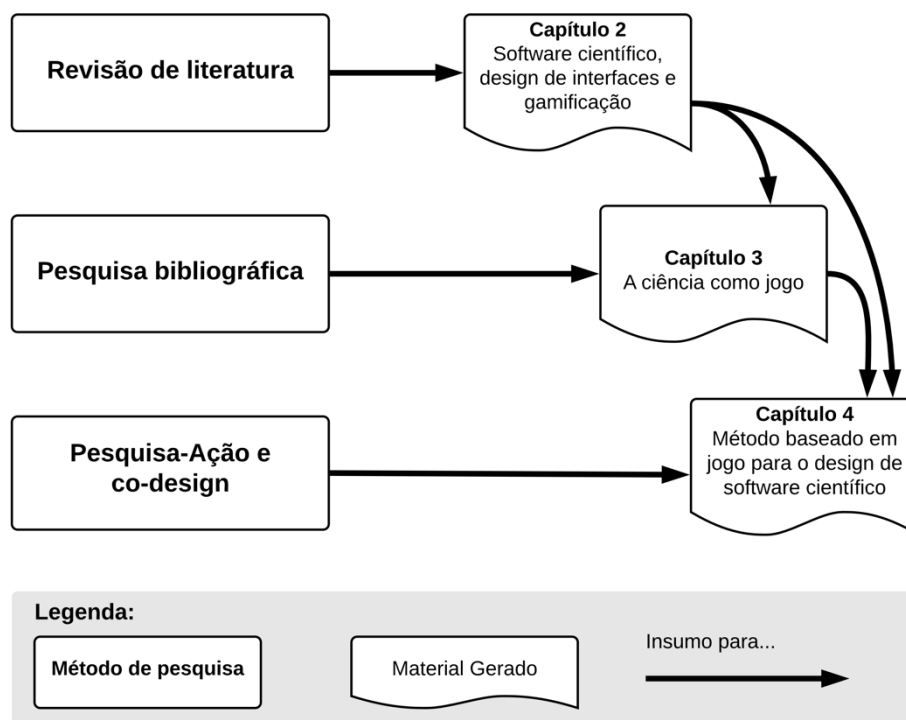


Figura 2 - Relações entre capítulos e métodos de pesquisa

Vale observar que os capítulos 2, 3 e 4 refletem uma divisão estrutural da tese em *revisão de literatura e trabalhos relacionados* (capítulo 2); *quadro teórico e conceitual* (capítulo 3); e *experimentos e resultados* (capítulo 4). Percorre-se, nesta sequência, o caminho da pesquisa – desde a assimilação das regras e paradigmas correntes, passando pela elaboração de um modelo que permita vislumbrar soluções para o problema apresentado e que culmina na demonstração experimental do

método de design proposto. Ao final da leitura deste trabalho, espera-se, será nítida a semelhança entre este percurso – da identificação de problemas à realização dos objetivos propostos, passando pelas regras e protocolos – e o próprio ato de jogar. Foi esta consonância entre jogo e ciência que, mais do que qualquer tendência em voga no campo do design, nos permitiu propor novas soluções e melhorias baseadas em jogos no campo da ciência computacional.

Ao longo de seu desenvolvimento esta pesquisa originou conteúdo para os seguintes artigos:

- *The Lens of the Lab: Design Challenges in Scientific Software* (Queiroz & Spitz, 2016) – originalmente apresentado na *10th International Conference on Design Principles and Practices* e posteriormente publicado no *International Journal of Design Management and Professional Practice*;
- *Position Paper: Collaborative Gamification Design for Scientific Software* (Queiroz & Spitz, 2016b), apresentado no *Fourth Workshop on Sustainable Software for Science: Practice and Experiences (WSSSPE4)*;
- *Video games as inspiration for scientific software* (Queiroz et al., 2016), apresentado no *XV Simpósio Brasileiro de Jogos e Entretenimento Digital (SBGames)*;
- *Good Usability Practices in Scientific Software Development* (Queiroz et al., 2017), apresentado no *Workshop on Sustainable Software for Science: Practice and Experiences (WSSSPE5.1)*;
- *[Artigo em processo de revisão]* (Queiroz et al., submetido).

## 2

### Software científico, design de interfaces e gamificação

O software científico pode ser definido por três características: é “desenvolvido para responder a uma questão científica, (...) depende do envolvimento de um cientista expert na área à qual se destina, (...) fornece dados para serem examinados pelo cientista que irá, de fato, responder à questão científica” (Kelly, 2013. p.348, tradução do autor). Trata-se, portanto, de um software “desenvolvido por cientistas, para cientistas” (Sletholt et al., 2012, p.24, tradução do autor). Tendo como principal objetivo contextualizar a pesquisa, o presente capítulo apresenta uma revisão de literatura a respeito do design de interfaces e de gamificação dedicados ao software científico, além de descrever, em maior detalhe, características, uso e desenvolvimento deste tipo de software. Neste caso, busca-se não apenas apresentar o estado da arte na pesquisa e elaboração de interfaces, mas também discutir peculiaridades da ciência computacional. Trata-se de uma investigação necessária para a identificação de desafios e exigências muito particulares, e que devem ser devidamente considerados por processos de design. Além disso, o capítulo apresenta a gamificação, suas propostas, aspectos críticos e aplicações em ciência.

Para que se pudesse chegar a uma visão ampla e aprofundada dos temas pesquisados, optou-se por uma metodologia baseada na revisão de literatura semi-sistemática, através da qual buscou-se investigar as seguintes questões e temas de interesse:

- Como o software científico é desenvolvido e usado?
- Como são projetadas as interfaces do usuário com o software científico?
- Como o software científico pode ser gamificado?

Foram selecionadas como fontes de consulta os seguintes bancos de dados de artigos científicos: ACM Digital Library, IEEE Explore, Scopus, AIS Electronic Library e Web of Knowledge. Para uma maior abrangência de resultados, foram realizadas buscas pelos seguintes termos: (1) ‘scientific software’ AND ‘gamification’; (2) ‘citizen science’ AND ‘gamification’; (3) ‘scientific software’



AND ‘user experience’; (4) ‘scientific software’ AND ‘user interface’; (5) ‘scientific software’ AND ‘HCI’; (6) ‘scientific software development’. Após a exclusão de material não-relacionado aos temas de interesse e remoção de duplicatas, chegou-se ao número de 270 referências. O número de resultados para os termos pesquisados em cada banco de dados, seguido pelo número de artigos selecionados após exclusões e remoção de duplicatas, pode ser visto na Tabela 1.

	ACM DL	IEEE Exp.	Scopus	AIS EL	WoK	Total
‘scientific software’ AND ‘gamification’	0	1	0	1	0	1
‘citizen science’ AND ‘gamification’	8	17	11	5	3	21
‘scientific software’ AND ‘user experience’	0	17	4	9	1	6
‘scientific software’ AND ‘user interface’	2	201	42	55	10	132
‘scientific software’ AND ‘HCI’	2	11	0	7	0	5
‘scientific software development’	13	140	81	62	54	105
Total	25	387	138	139	68	270

Tabela 1 - Resultados de buscas atualizadas no dia 4 de dezembro de 2015.

Artigos em temas correlatos, encontrados através de bancos de dados como Science Direct, Google Scholar e Gamification-Research.org, foram incluídos na pesquisa. Assim, buscou-se reconhecer contribuições significativas de artigos que, de outra forma, seriam ignorados. Além disto, parte do material selecionado foi excluída após nova triagem. Ao final do processo, foram selecionados 221 artigos como referência. Para a inclusão de material publicado após a data de busca originalmente realizada (dia 4 de dezembro de 2015), monitorou-se daquela data em diante, através do agregador de banco de dados Google Scholar, a publicação de artigos contendo as expressões ‘scientific software’, ‘scientific software’ AND ‘gamification’ e ‘scientific software’ AND ‘user interface’. Novamente, resultados considerados relevantes foram incorporados à pesquisa.

Ao longo das próximas seções, os resultados desta revisão serão discutidos de acordo com a seguinte sequência: primeiro, *Introdução ao software científico* define e descreve o objeto de pesquisa, salientando particularidades e desafios em seu desenvolvimento e uso. Em seguida, *Questões em design de interfaces e usabilidade em software científico* examina o design de interfaces naquele campo, levando em consideração pesquisas relacionadas e relatos de desenvolvimento. Depois, *Gamificação, jogo e software científico* discute a aplicação da gamificação

àquele tipo de software, considerando trabalhos previamente desenvolvidos e aspectos teóricos do design de gamificação. Finalmente, a *Conclusão do capítulo* apresenta um sumário das informações obtidas, a partir das quais sugere-se uma abordagem original para o design de gamificação do software científico, a ser desenvolvida nos capítulos subsequentes.

## 2.1

### Introdução ao software científico

O software científico ganhou muita importância nas últimas três décadas, transferindo a prática científica de tubos de ensaio para simulações em chips de silício (Woollard et al., 2008) e permitindo a condução de pesquisas em condições que, de outro modo, seriam “complexas demais, grandes demais, pequenas demais, perigosas demais ou caras demais para se explorar no mundo real” (Segal & Morris, 2008, p.18, tradução do autor). De fato, uma pesquisa realizada pelo *Software Sustainability Institute* (Hettrick et al., 2014) aponta que 92% dos pesquisadores do Reino Unido utilizam software científico, sendo que sete em dez consideram impossível conduzir suas pesquisas sem ele e, ainda, mais de 50% desenvolvem seus próprios softwares. Outra pesquisa, realizada em 2009, aponta que, dentre os cientistas que utilizam este tipo de software, 84.3% julga importante ou muito importante desenvolver seu próprio software. Para aqueles que desenvolvem seu próprio software, 30% do tempo de trabalho é utilizado no desenvolvimento de software, e 40%, no seu uso. Além disso, o software desenvolvido internamente por cientistas pode ser importante para outros pesquisadores (Hannay et al., 2009).

O software científico pode ser usado para “processar, analisar, visualizar, gerenciar, compartilhar, experimentar e (..) até gerar novos dados” (Ahmed & Zeeshan, 2014, p.55, tradução do autor), e funciona ainda como laboratório de testes para tecnologias experimentais (Mills et al., 1995), servindo de ligação entre abstração teórica e a indústria (Prego & Seisdodos, 2011). Além disso, ele pode ser usado para controle de equipamento de pesquisa em campo (Mielke et al., 2005) ou equipamentos laboratoriais distribuídos geograficamente (Gertz et al., 1994). Cabe ressaltar a grande diversidade, em tamanho e escopo, daquilo que pode ser considerado software científico: dependendo do contexto, o termo pode descrever um aplicativo completo, um *plug-in* ou um único arquivo de biblioteca, podendo

ser desenvolvido a partir do zero ou como um módulo para soluções desenvolvidas por terceiros (Frank et al., 2007). Finalmente, além de sua importância para aqueles que, de fato, o utilizam, cabe ressaltar que a vasta maioria das pessoas se beneficia dos avanços resultantes deste tipo de software (Kelly & Skordaki, 2015). O software científico é, geralmente, desenvolvido para as seguintes finalidades: (1) pesquisa científica em primeira mão; (2) treinamento e educação; (3) suporte a tomada de decisão externa (Sanders, 2008). Uma pesquisa realizada em 2010 aponta que a maior parte dos projetos é desenvolvida em universidades – seguidas por indústrias, institutos de pesquisa e órgãos governamentais – geralmente por equipes formadas por até seis membros. Vale notar que a mesma pesquisa se refere a equipes de sete a doze membros como “equipes grandes” (Nguyen-Hoan et al., 2010, p.4). De fato, muito frequentemente o software é construído para o próprio desenvolvedor ou para grupos de dez pessoas ou menos – embora haja um número significativo de projetos direcionados a grandes grupos (cem ou mais usuários). Há, ainda, casos de software que alcançam mais de cinco mil usuários e comunidades internacionais. Em sua maior parte, estes são projetos comerciais ou *open-source* (Hannay et al., 2009; Nguyen-Hoan et al., 2010).

### **2.1.1**

#### **Exigências da prática científica e acadêmica**

A natureza acadêmica é responsável por algumas das características, desafios, e necessidades mais pronunciadas no desenvolvimento de software científico. Em primeiro lugar, há a *motivação*: ao contrário de outros ambientes de desenvolvimento, o objetivo principal não está na criação de um produto ou serviço, mas no suporte à pesquisa sendo realizada (Basili et al., 2008). No meio acadêmico, estabelece-se reputação profissional através da publicação de artigos. Neste caso, o software pode ser percebido por cientistas como meros meios para publicação. Esta atitude pode afetar a qualidade do software (Killcoyne & Boyle, 2009) e desmotivar seus desenvolvedores (Howison & Herbsleb, 2011), que podem, inclusive, ter dificuldades em obter um engajamento maior, por parte dos cientistas, no processo de elaboração do software (Segal, 2009). De fato, a colaboração entre as diversas partes envolvidas – embora vital para o processo – pode ser um desafio, e deve ser apoiada pelo estabelecimento de uma boa comunicação entre todos os envolvidos

(Morris & Segal, 2012; Taweel et al., 2009). Isto pode ser particularmente importante quando participantes possuem diferentes formações profissionais, encontram-se em diferentes partes do globo (Marinovici et al., 2014), trabalham em projetos de longa duração e mantêm visões particulares acerca do projeto (Spencer, 2015). Neste último caso, práticas e filosofias de trabalho devem ser adotadas para que sentimentos individuais de autoria não comprometam a colaboração (Turk, 2014).

O software científico apresenta, ainda, necessidades particulares associadas à prática científica. *Correção*, um “valor central da ciência” (Howison & Herbsleb 2011, p.3, tradução do autor), permanece como principal qualidade e maior preocupação – à frente de usabilidade, manutenção e custo (Kelly & Sanders, 2008). A *Reprodutibilidade* também se mostra essencial para o método científico (Cimiano & Sagerer, 2015): possibilitando a validação e revisão de resultados (Recio-Garcia et al., 2013), este aspecto costuma exigir automação de processos e uma adequada portabilidade do software para outras máquinas. Outro aspecto a ser considerado é a necessidade do software científico de lidar com *dados complexos* em grandes níveis de *precisão* (Hatton & Roberts, 1994). A complexidade dos dados e processos envolvidos podem exigir a *interoperabilidade* entre o software desenvolvido e outros sistemas (Fdez-Riverola et al., 2012), desde outros softwares, até sistemas robóticos e industriais (Picón et al., 2006) ou aplicativos como editores de planilhas (Trlica, 1997; McKiney, 2003). *Mudanças nas teorias científicas* adjacentes ao software (ou nas suas interpretações) podem resultar em dificuldade no estabelecimento de requisitos, ou seja, na definição daquilo que deverá, de fato, ser implementado. Finalmente, *teste* e validação podem ser desafiadores, uma vez que a comparação de resultados com dados obtidos no mundo real é impraticável (Segal & Morris, 2008; Heaton & Carver, 2015), e o teste por código pode exigir conhecimento avançado do domínio científico ou a ajuda de um *expert* do meio (Kelly et al., 2011).

### 2.1.2

#### Aspectos de cunho colaborativo

Projetos de desenvolvimento de software científico frequentemente dependem da colaboração entre cientistas do domínio ao qual o software se dedica e engenheiros

de software (Holthouse & Greenberg, 1978). No entanto, Kelly (2007) denuncia a existência de um abismo entre membros da comunidade científica e da comunidade de engenharia de software. Os primeiros, pesquisadores especialistas em um domínio científico, não detêm conhecimento ou prática em engenharia de software – ainda que detenham conhecimentos em programação. Os segundos, não associados a um domínio científico específico, detêm conhecimento especializado no projeto e desenvolvimento de software. Não raro, cientistas tecnicamente aptos – mas não especialistas – em desenvolvimento de software desenvolvem, eles mesmos, os softwares que utilizam ou distribuem. Segal (2007) refere-se a indivíduos deste grupo como o ‘usuário desenvolvedor profissional’, para quem o desenvolvimento de software é uma atividade secundária e que desenvolve, sozinho ou na companhia de outros cientistas, para uso próprio ou de suas comunidades.

Ambientes de desenvolvimento de software científico são frequentemente descritos como adversos às boas práticas propostas pela engenharia de software, se utilizando de processos informais e procedimentos não muito claros (Ahmed & Zeeshan, 2014) – uma falta de planejamento que pode ter efeitos negativos no desenvolvimento e comercialização (Sanders, 2008) e, embora possa servir a pequenos programas desenvolvidos para pesquisas pontuais, é inadequada a sistemas complexos (Spencer, 2015). Uma pesquisa publicada em 2009 mostra que a maior parte dos cientistas consultados acredita que gerenciamento de projeto e educação formal em engenharia de software não são particularmente importantes (Hannay et al., 2009). Práticas de engenharia de software podem ser percebidas como burocráticas e frustrantes por cientistas, que podem se sentir desmotivados a adotá-las (Spencer, 2015), não reconhecendo a necessidade de melhorias no processo de desenvolvimento. Neste caso, defensores destas práticas podem ser mais convincentes ao inserirem-se no processo de desenvolvimento e demonstrarem ganhos em potencial (Brebner, 1998). Por outro lado, tanto a comunidade científica quanto a comunidade de engenharia de software parecem concordar que a qualidade do software científico pode aumentar através da adoção de práticas de engenharia de software (Mohammad, 2010; Heaton & Carver, 2015), incluindo a especificação de requisitos, automação de testes e documentação (Kosteka & Mishev, 2013). Da mesma forma, seria aconselhável que engenheiros de software adquirissem conhecimento sobre a área científica a que se destina o

software (Marinovici et al., 2014) e sobre como cientistas trabalham e reportam seu progresso (Killcoyne & Boyle, 2009).

No entanto, a mentalidade do ‘usuário desenvolvedor profissional’ não é o único obstáculo entre o software científico e práticas de engenharia de software: a computação científica oferece outros desafios, a começar pela já mencionada dificuldade na definição de requisitos. Brooks afirma que “a parte mais difícil de se construir um software é decidir com precisão o que se deve construir” (Brooks apud Ovaska et al., 2005, p.32, tradução do autor). A afirmativa é particularmente verdadeira em relação ao software científico, uma vez que as teorias nas quais o software se baseia podem mudar durante a pesquisa, forçando requisitos a mudarem, evoluírem, e emergirem durante o desenvolvimento. Requisitos são, neste caso, compreendidos ao longo do processo (Segal, 2007), “à medida que o software e o entendimento do domínio científico progridem concomitantemente” (Segal & Morris, 2008, p.19, tradução do autor). Frequentemente, requisitos não são formalmente documentados ao longo do processo de desenvolvimento – embora, em alguns casos, seja exigida sua documentação em formas diversas, como UML (Daniluk, 2012), protótipos de baixa e alta fidelidade (Aragon et al., 2008) ou *storyboards* interativos (Sanders, 2008). Idealmente, requisitos devem ser consolidados como descrições técnicas e não ambíguas de funcionalidades específicas (Marinovici et al., 2014). Em alguns casos, o uso de personas pode ser produtivo na definição de requisitos (Scheidewind et al., 2012). Deve-se, no entanto, evitar buscá-los através de perguntas informais aos usuários (Morris & Segal, 2012).

Outro aspecto desafiador é a *longevidade* do software científico, que pode ser desenvolvido, usado e mantido por longos períodos de tempo, frequentemente medido em décadas (Basili et al., 2008; Sanders, 2008; Kelly, 2009). Por este motivo, ele pode exigir refatoração – melhorias na estrutura do código (Heaton & Carver, 2015; Spencer, 2015), devendo ser planejado para ser sustentável (Morris & Segal, 2012) e para ser expansível, uma vez que pode crescer além do escopo inicialmente planejado, dependendo de seu sucesso (Basili et al., 2008). É recomendável, ainda, manter o software simples, evitando o acréscimo de funcionalidades desnecessárias (Gorton, 2014). Em alguns casos, é melhor que se faça a incorporação de novas funcionalidades através de ‘módulos’ acrescentados ao software principal (Sanders, 2008). Por outro lado, há casos em que a vida útil

do software é, de fato, curta – sendo o software descartado em curto prazo (Segal & Morris, 2012).

Para melhor abordar esses desafios – longevidade do software e dificuldade na definição de requisitos – o software científico é normalmente desenvolvido *iterativamente* (Sanders, 2008), ou seja, em pequenos e repetidos ciclos de desenvolvimento seguidos da avaliação da última versão implementada. A realização de mudanças incrementais é vista como uma boa prática em computação científica (Ackroyd et al., 2008; Wilson et al., 2014). Iterações regulares refletindo mudanças incrementais “permitem que usuários influenciem o desenvolvimento e estabeleçam requisitos” (Ackroyd et al., 2008, p.46, tradução do autor). Do ponto de vista do projeto como um todo, responder às necessidades de uma base de usuários bem definida pode ser mais eficaz do que buscar uma solução completa que possa ser generalizada para todos os casos (De Roure & Goble, 2009). Neste contexto, a adoção de metodologias de desenvolvimento pode ser problemática. A natureza iterativa do software científico torna-o, de certa forma, incompatível com o tradicional modelo de desenvolvimento em cascata (Holthouse & Greenberg, 1978). A metodologia *Agile* se mostra como uma melhor opção (Ahalt et al., 2014): por adotar a mudança contínua de requisitos (Nantaamornphong et al., 2014), esta metodologia parece ser adequada “à natureza exploratória, iterativa e colaborativa da inquisição científica” (Kane et al. apud Crabtree et al., 2009, p.22, tradução do autor). Entretanto, desenvolvedores de software científico podem ser seletivos na adoção de práticas da metodologia *Agile* (Sletholt et al. 2012), algumas das quais podem ser impraticáveis no trabalho com cientistas como, por exemplo, reuniões diárias (Kelly & Smith, 2010). De fato, ao invés de se adaptarem a uma metodologia pré-estabelecida, desenvolvedores podem adotar apenas algumas práticas selecionadas (Nanthaamornphong & Carver, 2015) ou “adaptações livres” daquelas metodologias (Basili et al., 2008). Em alguns casos, equipes de desenvolvimento optam, em nome da praticidade e eficiência, por uma abordagem “ametodológica” (Kelly, 2015). Por várias vezes, metodologias especializadas, voltadas para o software científico, foram criadas (Cort et al., 1985, Platz, 1986; Pereira Júnior, 2007; Ahmed et al., 2014), reforçando aspectos normalmente negligenciados (ou difíceis) como requisitos, teste e design. Já em 1986, Platz observou que problemas e soluções poderiam não ser detectados durante as primeiras fases do projeto, e criou um sistema para gerenciamento de projetos que, enquanto rigidamente

estruturado, era projetado para permitir mudanças de requisitos, reconhecendo a necessidade de equilíbrio entre medidas de engenharia de software e liberdade criativa (Platz, 1986).

Percebe-se, no entanto, que há em curso um amadurecimento dos métodos e processos de desenvolvimento de software científico. Se até a década passada não havia consenso quanto às melhores práticas de desenvolvimento (Sanders, 2008), várias destas práticas foram identificadas e organizadas (Wilson et al., 2014; Wilson et al., 2017). Além disso, a preocupação em educar e orientar alunos para práticas adequadas estimulou a elaboração de disciplinas especializadas (Wolff, 2015; Ponce et al., 2016).

Para além da colaboração entre membros de diferentes especialidades dentro de uma mesma equipe, é interessante notar que comunidades para recursos de computação científica gratuitos e abertos têm estimulado esforços colaborativos há bastante tempo, evoluindo a partir de associações individuais, criadas nos anos 1950, para comunidades globais através de serviços de assinatura via *e-mail* e servidores de *Arpanet* e *Internet* durante os anos 1980 e 1990 – práticas a partir das quais se originou o movimento *Open Source* (Dongarra et al. 2008), que contrapõe-se a modelos de desenvolvimento e comercialização fechados que por vezes buscam, por meio de sistemas de patentes, impedir a produção de alternativas abertas – prejudicando o desenvolvimento de soluções e perpetuando estruturas de poder que sustenta companhias produtoras de software científicos comerciais (Jacob, 2016).

É importante destacar, também, que a ciência tem se tornado uma atividade cada vez mais coletiva e colaborativa, o que pode ser visto no surgimento de “colaboratórios”: coletivos de institutos voltados a uma mesma pesquisa, facilitados pelo uso de uma mesma ciberestrutura (Coulter et al., 2016). Nos últimos dez anos, novos esforços para colaboração em massa – como comunidades abertas, *open data*, ciência cidadã e ciência coletiva – vêm se tornando um novo paradigma (Vanschoren et al., 2014). Franzoni & Sauermann (2014) identificam, neste sentido, duas rupturas com a ciência “tradicional”: abertura de pesquisas a uma ampla base de participantes (em alguns casos, centenas de milhares) e a divulgação de informações sensíveis à pesquisa normalmente restritos (ex.: dados e algoritmos). Essas formas abertas de se fazer ciência são benéficas ao cientista de várias formas, dentre as quais maior número de citações a suas publicações, oportunidades de



financiamento e reconhecimento por parte de empregadores (McKiernan et al., 2016). Em alguns casos, iniciativas de ciência coletiva são gamificadas – uma relação que será debatida em uma próxima seção. Estes novos modelos de ciência coletiva desafiam o modelo tradicional onde grupos fechados de pesquisa competem por publicações, sugerindo novas formas de autoria coletiva e influenciando o modo como a ciência é tradicionalmente feita no que se refere à colaboração, motivação, geração de conhecimento, abertura, transparência, organização de equipes e, inclusive, novas formas de se incutir procedimentos de pesquisa e rigor científico em software (Franzoni & Sauermann, 2014).

No contexto do desenvolvimento de software, destacam-se ainda novas práticas de trabalho colaborativo, como o *trabalho cooperativo auxiliado por computador* (CSCW, do inglês *Computer Supported Cooperative Work*) – onde a colaboração é facilitada por tecnologias de comunicação como fóruns de discussão, plataformas de trabalho online, etc., e o *engajamento de codificação comunitária* (CCE, do inglês *Community Code Engagement*), caracterizado por curtos períodos de trabalho colaborativo intenso (Trainer, 2016), potencialmente capaz de sustentar o engajamento de comunidades de software científicos *Open Source* (Hwang, 2016). Esta cultura colaborativa funciona como contraponto ao desenvolvimento individual e pontual de software para uso exclusivo de pequenos grupos fechados. Neste caso, usuários e desenvolvedores podem ser classificados quanto a seus níveis de reciprocidade, determinados pelos graus de contribuição, colaboração e abertura do software (Goble, 2016). Idealmente, o próprio software deveria dar suporte a esforços colaborativos, facilitando integração de dados, compartilhamento de conteúdo, documentação e gestão do conhecimento (Pinto et al., 2002). De fato, ao apoiar a colaboração entre membros de comunidades para o compartilhamento de *know-how*, o software pode ter seu uso e adoção estimulados (De Roure & Goble, 2009).

### 2.1.3

#### Plataformas de uso e linguagens de programação

Desenvolve-se software científico para diversas plataformas. Computadores *desktop* e estações de trabalho são duas plataformas bastante comuns (Hannay et al., 2009). Mais raramente, desenvolve-se para equipamentos portáteis como *tablet*

PCs (schraefel et al., 2004) e *smartphones* (Clark, 2014). Em alguns casos, no entanto, o software pode exigir *Computação de Alta Performance* (ou HPC, do inglês, *High Performance Computing*), obtida através de recursos como supercomputadores (Hatton & Roberts, 1994; Hannay et al., 2009), *mainframes* (Chen e Fu, 1996), *clusters* (Cohen et al., 2013), *grids* (Choi et al., 2006; Frank et al., 2007) e, mais recentemente, serviços de computação em nuvem (Hou et al., 2010; Church et al., 2012; Mendez et al., 2013). Recursos de HPC podem ser estruturalmente complexos (Kovalchuk et al., 2012), sendo normalmente acessados pelos usuários através de computadores desktop ou ferramentas web (Ernst et al., 2003; Kolberg et al., 2007; Gomes et al., 2015). Permitir acesso a recursos HPC por parte de usuários de sistemas menos poderosos tem sido, há décadas, uma questão importante – não somente pela democratização de recursos HPC como também por questões de reuso de código e reprodutibilidade de pesquisas (Gesing et al., 2016). No entanto, garantir que todos os usuários de uma rede tenham experiências de uso semelhantes já foi um desafio maior (Atkins & Phillips, 1986; Phillips et al., 1986). Neste sentido, serviços baseados na *web* e serviços de nuvem têm provido uma democratização destes recursos (Afgan et al., 2015), permitindo aos cientistas concentrarem-se em problemas de seus domínios científicos, e não problemas computacionais (Gomes et al., 2015), com altos nível de eficiência e segurança (Kryukov et al., 2016).

No que se refere às tecnologias empregadas no desenvolvimento do software científico, desenvolvedores costumam adotar um posicionamento conservador, adotando tecnologias mais antigas, ou compatíveis com estas – além de tender a iniciar projetos a partir do zero, ao invés de adotar *frameworks* preexistentes (Basili et al., 2008). Ao discutir computação em nuvem e implementação de software científico em *smartphones*, Clark detecta um atraso do campo na adoção de novas tendências tecnológicas, o qual atribui à “alta complexidade e à estreiteza do segmento de mercado” (Clark, 2014, p.237, tradução do autor). O software científico pode ser desenvolvido em várias linguagens de programação como Fortran, Binary, Assembler, C, Lisp, C++, Java, C#, Pascal, Perl, Python, R, PHP, Matlab, etc. (Nguyen-Hoan et al., 2010; Ahmed & Zeeshan, 2014), e podem ser, ainda, desenvolvidos com a ajuda de *frameworks* específicos para o domínio científico ao qual se destina (Glez-Peña et al., 2010; Fdez-Riverola et al., 2012). É interessante notar, no entanto, uma tendência, surgida da década de

2010, do uso de *engines* de jogos no desenvolvimento de ferramentas de visualização científica (Miles, 2011; Wang et al., 2012; Dos Santos et al., 2016; Bergman et al., 2017).

#### 2.1.4

##### **Etapas do trabalho em software científico**

De modo geral, o trabalho em software científico – particularmente aquele realizado através de recursos em rede – pode ser dividido em três etapas: *modelagem*, *simulação* e *análise de resultados* (Kovalchuk et al., 2012) – não necessariamente obtidas através do mesmo software. Primeiro, durante a fase de *modelagem*, o problema científico é representado em termos matemáticos (Li, 2011) e programado para representar, durante a simulação, o sistema científico sendo estudado (Daniluk, 2012). Ao propor linhas-guia para notações científicas digitais – as quais considera a principal interface humano-computador para pesquisa – Hinsén destaca que tais modelos são conhecimento científico, e devem ser projetados para serem lidos e compreendidos por cientistas, e não simplesmente assimilados ao código-fonte do software (Hinsén, 2016).

Uma vez prontos, modelos podem ser utilizado na fase seguinte, a de *simulação*, na qual dados são submetidos e processados por uma série de tarefas computacionais. Em casos de simulações complexas, interdependências e conexões entre tarefas individuais podem compor um *workflow* (Woollard et al., 2008). *Workflows* podem ser automatizados através de programação, editores visuais ou frameworks especializados (Vidger et al., 2008; De Roure & Goble, 2009; Silva, 2010). Eles podem envolver cálculos complexos, possivelmente exigindo que os dados submetidos sejam processados sequencialmente por uma série de tarefas computacionais – às vezes passando por diversos softwares diferentes. Para minimizar o risco de falhas no processo – que pode levar dias – sugere-se que *workflows* sejam divididos em pequenas etapas e utilizem sistemas de *checkpoints* capazes de registrar por quais destas etapas os dados já passaram (Gross et al., 2008; Nguyễn et al., 2011). Em alguns casos, *workflows* podem ser gravados para reutilização futura, servir de base para novos, exportados e compartilhados (Bergmann et al. 2011).

Finalmente, após a *simulação*, os dados de saída estão prontos para a fase de *análise de resultados*. Esta etapa deve ser a culminação do trabalho em software científico, uma vez que "o propósito da computação é gerar *insight*, e não números" (Hamming apud Kelly & Sanders, 2008a, p.3, tradução do autor). A análise de resultados depende da visualização e manipulação de dados quantitativos e/ou qualitativos. Descobertas decorrentes da análise de resultados podem ser publicadas ou avançar a busca pela resposta à questão científica sendo estudada. Como veremos mais adiante, cada etapa do trabalho em software científico pode oferecer desafios e oportunidades diferentes na elaboração de suas interfaces.

## 2.2

### Questões em design de interfaces e usabilidade em software científico

A Interação Humano-Computador (IHC) é frequentemente vista como deficitária, “a etapa mais ignorada e abandonada do desenvolvimento de soluções em software científico” (Ahmed et al., 2014, p.6, tradução do autor). De várias formas, esta negligência reflete a mentalidade do “usuário desenvolvedor profissional”. A proximidade do desenvolvedor com o domínio científico pode ser um complicador, podendo fazê-lo acreditar que usuários com experiências profissionais semelhantes terão facilidade em entender o funcionamento do software, negligenciando testes com usuários e documentação (Kelly & Sanders, 2008b; Sanders, 2008). De fato, usabilidade é considerada um fator menos importante por desenvolvedores de softwares direcionados a usuários que saibam, eles mesmos, programar (Nguyen-Hoan et al., 2010). Preocupados, principalmente, com o motor computacional do software (Sanders, 2008), desenvolvedores tendem a acreditar que, caso o software seja útil, “quase qualquer interface com usuário será tolerada” (MacLeod et al., 1992, p.415, tradução do autor). De fato, a usabilidade do software científico pode ser complicada desde a instalação e configuração (Geimer et al., 2014), possivelmente ofuscando a utilidade e funcionalidades do software (Papadimitriou et al., 2011). Somado a tudo isso, a criação de interfaces gráficas requer habilidades especializadas e são, possivelmente, “a última coisa com a qual um cientista quer lidar” (Lundstrom & Klimeck, 2006, p.497, tradução do autor). Em alguns casos, encara-se a documentação como um substituto para usabilidade. No entanto, a documentação de software científico é, muitas vezes, incompleta e, não raro,

baseada em questões levantadas pelos usuários – quando não criada pelos próprios (Pawlik et al., 2012). Adicionalmente, o acesso à documentação durante uso do software pode ser pouco prático (Sanders, 2008). Como possível reflexo destas dificuldades, o *aprendizado* bem-sucedido do uso do software científico geralmente resulta de um processo cíclico, que exige grande investimento de tempo por parte do usuário para compreensão das ferramentas, busca por ajuda de outros usuários *in loco*, busca por recursos de aprendizado *online*, e busca por feedback das comunidades virtuais do software. (Skordaki & Bainbridge, 2018).

Desenvolvedores não são, entretanto, o único grupo responsável pela falta de atenção à usabilidade. Em alguns casos, pessoas e entidades responsáveis pelo financiamento do software podem interpretar mal as necessidades dos usuários (Morris & Segal, 2012). Podem, ainda, oferecer resistência ao investimento em pesquisas em usabilidade, uma vez que a influência do design sobre a geração de *insights* é difícil de medir e demonstrar (De Matos et al., 2013).

A falta de atenção à usabilidade representa um grande risco, ampliado pelo aumento da complexidade dos softwares científicos (Kendall et al., 2007). Este risco pode ser ainda maior nos casos de desenvolvimento para uso externo (Sanders, 2008) – interfaces complicadas e problemas na visualização de informações podem levar usuários a abandonarem aplicativos (Ahmed & Zeeshan, 2014). Em contrapartida, interfaces amigáveis podem ser um grande motivo para adoção de determinado software (Joppa et al., 2013), podendo até estimular a adoção de softwares com aparente falta de desempenho (Manjunatha et al., 2011). Cientes dos riscos envolvidos, há desenvolvedores que adotam boas práticas no projeto de interfaces, como: elaboração de *storyboards*; testes de usabilidade contínuos; observação do campo de trabalho; testes em ambientes controlados; elaboração de sistemas de ajuda (Sanders & Kelly, 2008); além do planejamento de usabilidade em UML (La Rue et al., 2014).

No entanto, ao se projetar para software científico, deve-se atentar para algumas de suas características bastante peculiares – algumas das quais descritas na seção anterior. As subseções a seguir descrevem como o design de interface aborda algumas destas características, além de discutir avanços em pesquisas, diretrizes e soluções de design.

### 2.2.1

#### Quanto às alternativas às interfaces gráficas

Interfaces gráficas (ou GUIs, do inglês *Graphical User Interfaces*) não são, em muitos casos, estritamente necessárias para o uso de software científico (Chen & Fu, 1996; Jarvis et al., 2006). Muitas vezes, versões iniciais dos programas se limitam a recursos acessíveis somente por linhas de comando, tendo suas interfaces gráficas construídas em etapas mais avançadas do processo de desenvolvimento – às vezes através de integração com software de terceiros (Chancelier et al., 2014). De fato, usuários podem achar conveniente ignorar GUIs e utilizar interfaces baseadas em texto (Joshi et al., 1997), programação (Weerawarana et al., 1996), arquivos de configuração, DSLs, e até mesmo a substituição dos parâmetros diretamente no código e recompilação do programa (Beg et al., 2016). Interfaces baseadas em linhas de comando são consideradas convenientes por facilitar a automação e repetição de ações (Wilson et al., 2014). Além disto, interfaces gráficas aumentam em complexidade o desenvolvimento e documentação do projeto (Smith et al., 2016). Por estes motivos, recomenda-se manter o código da interface gráfica separado do código dos cálculos científicos (Kelly et al. 2009), o que facilita, ainda a customização e reconfiguração das interfaces (Gomes et al., 2015). A operação por meio de comandos pode ser muito importante e, como veremos adiante, pode ser combinada ao uso de interfaces gráficas.

Por outro lado, reconhece-se que GUIs podem tornar o software mais amigável aos cientistas (Murphy, 1996; Belsky et al., 2002), especialmente àqueles sem experiência na área de computação (Cohen et al., 2013). Embora haja casos, conforme discutido, em que linhas de comando são mais eficientes, softwares científicos complexos costumam exigir interfaces gráficas (Versek, 2013) – que podem ser muito importantes por mediarem “o processo visual e cognitivo do usuário com os numerosos cálculos computacionais (...) [ocupando] a distância entre o humano e o computador” (Foulser & Gropp, 1990, p.22, tradução do autor), providenciando “abstrações em alto nível das capacidades computacionais subjacentes” (Ramakrishnan & Rice, 1996, p.2, tradução do autor). Interfaces gráficas são valorizadas por uma categoria específica de software científico: o ambiente de solução de problemas (PSE, do inglês *Problem Solving Environment*). PSEs são definidos como “sistema[s] de computação que disponibilizam todas as

facilidades computacionais necessárias para se resolver tipos específicos de problemas (...) e comunicam-se nos termos dos usuários” (Gallopoulos et al., 1992, p.7, tradução do autor). Rice & Boisvert definem este tipo de software através da fórmula “PSE = interface com usuário + bibliotecas computacionais + base de conhecimento + integração” (1996, p.47, tradução do autor). PSEs não exigem conhecimentos especializados em programação por parte do usuário: problemas são formulados, simulados, resolvidos e exibidos com a ajuda de interfaces gráficas de alto nível em sua elaboração, visualização e seleção de métodos para resolução de problemas (Houstis & Rice, 2000).

### 2.2.2

#### Quanto ao uso de linguagens de programação por usuários

Não é incomum que usuários de software científico precisem, durante o uso dos mesmos, engajar-se em tarefas de programação (Wang et al., 2010). Para estes casos, linguagens mais tradicionais e complexas como as mencionadas na subseção anterior podem dar lugar a linguagens de programação visual (ou VP, do inglês *Visual Programming*) ou a linguagens especificamente projetadas para o domínio científico (ou DSL, do inglês *Domain Specific Language*), utilizadas na manipulação e visualização dos dados (Jones & Scaffidi, 2011). Soluções para desenvolvimento baseadas em DSLs são disponibilizadas tanto em produtos de código-fonte fechado – como Matlab, Maple e Mathematica – quanto em produtos de código-fonte aberto – casos de Octave, Scilab e jLab (Papadimitriou et al., 2009), trazendo linguagens mais simples e menos propensas a erro do que linguagens como C (Hinsen, 2013). Por outro lado, linguagens de programação visual permitem aos cientistas a construção de equações e modelos científicos através de uma interface que permite arrastar e conectar componentes, similar a ambientes CAD/CAM e outras ferramentas de diagramação de fluxo (Keller & Rimmon, 1992; Rijnders et al., 1993). Um exemplo deste tipo de ambiente de programação é o LabVIEW (Figura 3), software para automação e simulação laboratorial introduzido em 1986, que combina diagramas do tipo *data flow*, blocos de construção, instrumentos virtuais e uma linguagem de programação textual (Kodosky et al., 1991; Wang et al., 2010).

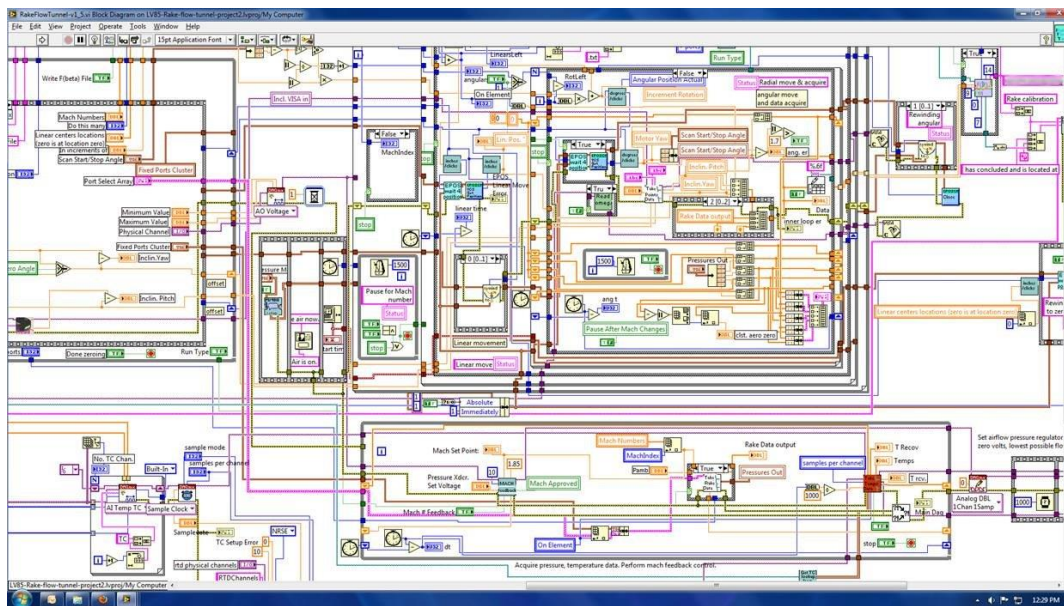


Figura 3 - Interface do LabVIEW

A usabilidade das linguagens de programação visual torna cientistas mais produtivos em seus trabalhos, exceto quando há grande impacto na velocidade da execução dos processos computacionais – normalmente mais lentos nestas linguagens do que em linguagens de programação tradicionais (Chambers, 2013). Esta deficiência na performance pode, em alguns casos, ser amenizada: algumas linguagens de programação visual permitem a recompilação de seus códigos em linguagens mais velozes, ou mesmo em linguagem de máquina (Bilmes, 1996). Outra abordagem possível é ensinar ao usuário, através do próprio software, técnicas de otimização para programação visual (Chambers, 2013). Como vantagem adicional, observa-se ainda que permitir ao usuário desenvolver e integrar suas próprias soluções pode conferir mais valor ao software (De Roure & Goble, 2009).

### 2.2.3 Quanto às etapas do trabalho científico

O design de interfaces para software científico pode oferecer desafios específicos, dependendo das etapas de trabalho científico a serem atendidas.

A fase de *modelagem*, por exemplo, pode exigir a combinação de conhecimentos no domínio científico, modelagem matemática e programação. Estas necessidades podem resultar em soluções interessantes. Modelos matemáticos



podem ser construídos através de interfaces para diagramação (Julvez et al., 2014; Bunus, 2006). A Figura 4 apresenta um exemplo deste tipo de interface.

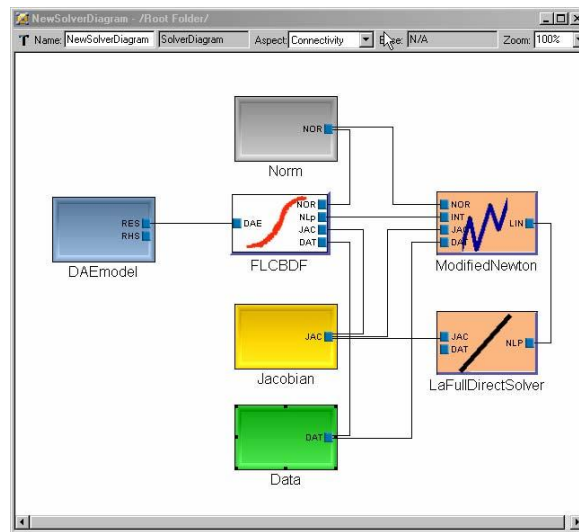


Figura 4 - Interface para diagramação de modelo (Bunus, 2006).

A fase de modelagem pode ser apoiada por métodos para a consulta às bases de conhecimento específicas do domínio científico. A Figura 5 mostra, como exemplo, um aplicativo capaz de filtrar oferecer ao usuário operações que sejam consistentes com o modelo sendo construído (Keller et al., 1994).

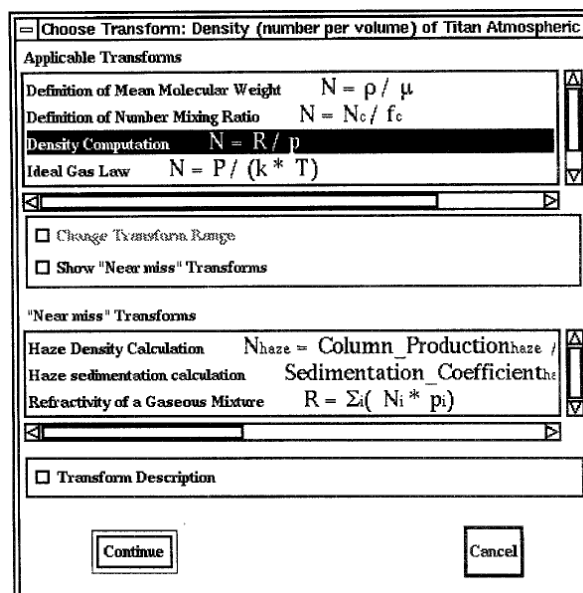


Figura 5 - Interface para bases de conhecimento científico (Keller et al., 1994).

Além disto, a interface pode providenciar um entendimento claro dos conceitos científicos subjacentes ao código de programação. Como exemplo, a

Figura 6 ilustra um software para análise semântica capaz de reinterpretar seções de código de programação em termos técnicos e científicos (Stewart, 2001).

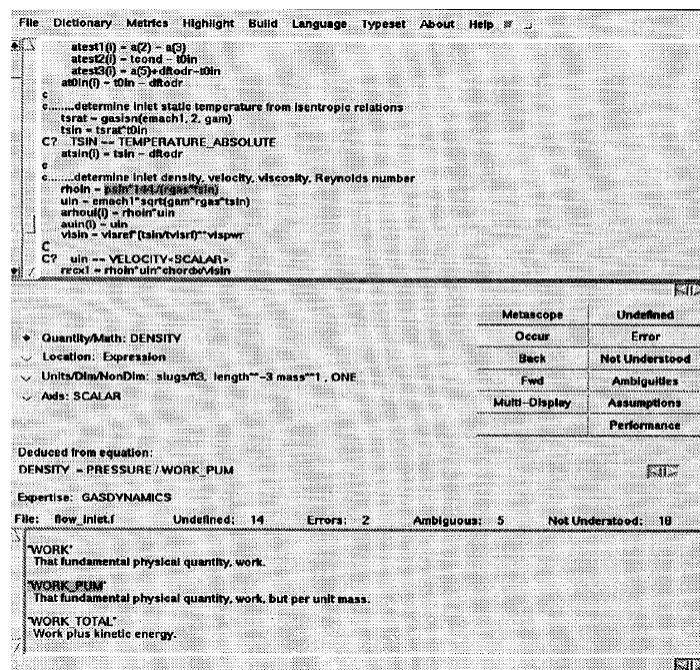


Figura 6 - Interface para análise semântica de código (Stewart, 2001).

Neste contexto, um grande desafio no projeto de desenvolvimento de software científico é elaborar ou utilizar um sistema de notação científica eficaz do ponto de vista computacional e, também, claramente legível por cientistas (Hinsen, 2016).

A fase de *simulação* pode exigir ferramentas dedicadas à construção e execução de *workflows* que podem ser criados, configurados, submetidos e monitorados através de interfaces gráficas para facilidade de uso (Bergmann et al., 2011; Dong & Wild, 2008) e para maior produtividade (Vidger et al., 2008; De Roure & Goble, 2009). Tal como ocorre com modelos, *workflows* são frequentemente editados através de diagramas – que neste caso interligam sequências de serviços e tarefas computacionais através dos quais os dados serão transformados. A Figura 7 apresenta *Taverna Workbench*, exemplo de software dedicado à criação de *workflows*.

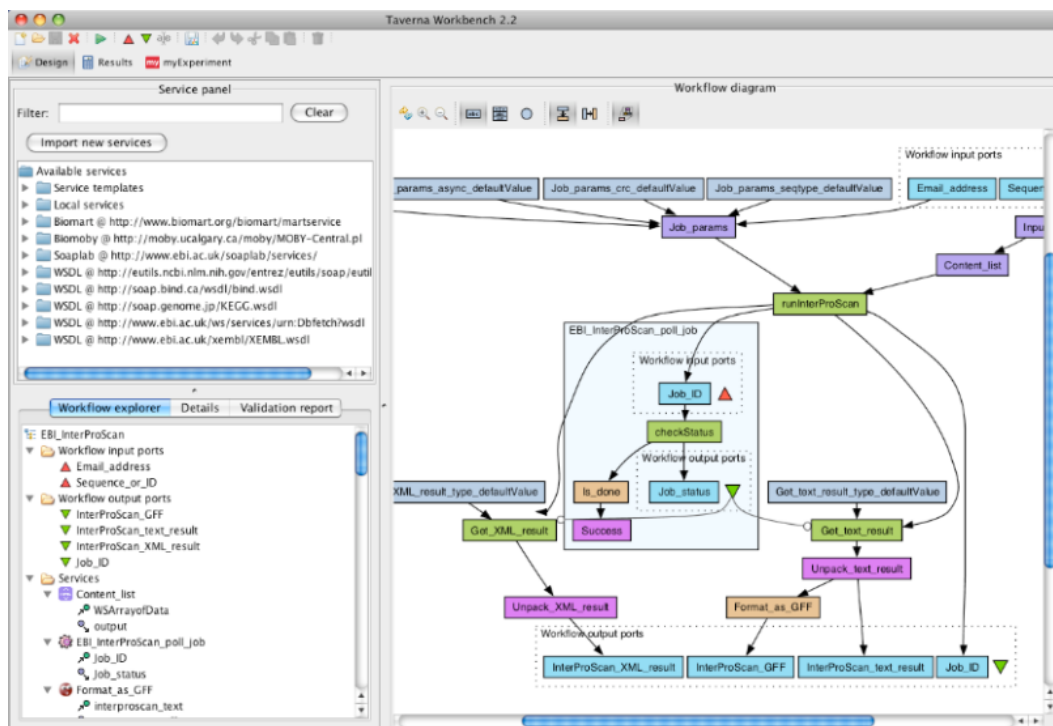


Figura 7 - Workflows no Taverna Workbench

A fase de *análise de resultados* é, geralmente, bastante visual, exigindo interfaces que ofereçam a visualização e manipulação dos dados obtidos pela simulação. Algumas vezes, estes dados podem ser visualizados como representações dos objetos de estudo, como estruturas de cristais (Belsky et al., 2002), terrenos geográficos (Feibush et al., 2000) ou imagens médicas como as exibidas na Figura 8 (Bitter et al., 2007).

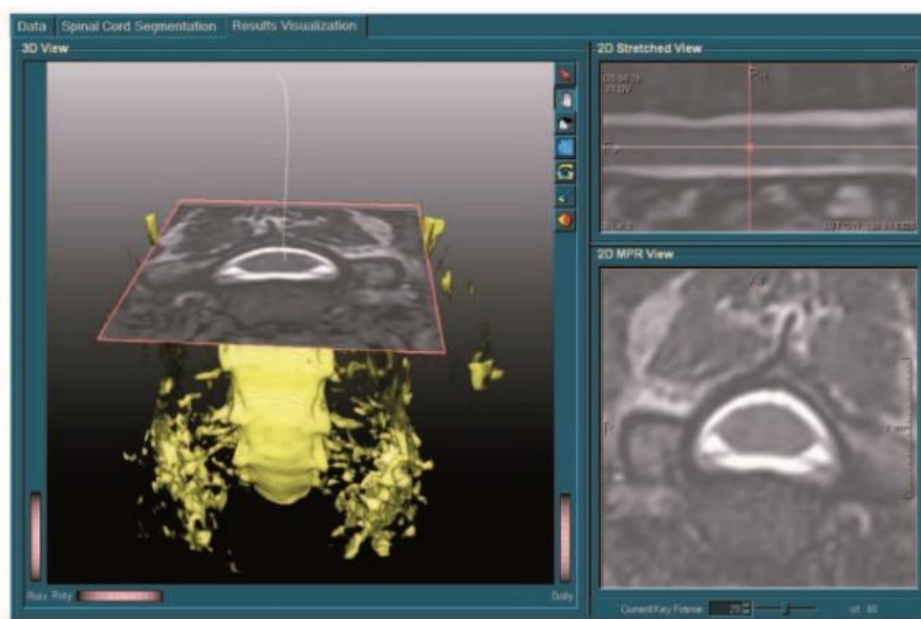


Figura 8 - MeVisLab (Bitter et al., 2007)

Em outros casos, a análise é feita através da visualização de dados quantitativos, envolvendo técnicas diversas como diagramas (Murali et al., 1993) e gráficos em duas ou três dimensões (Willson et al., 1992; Curtis et al., 2011; Ellis et al., 2013). A título de exemplo, a Figura 9 apresenta a interface do software GenAMap, software para mapeamento de associações entre dados e mutações genéticas.

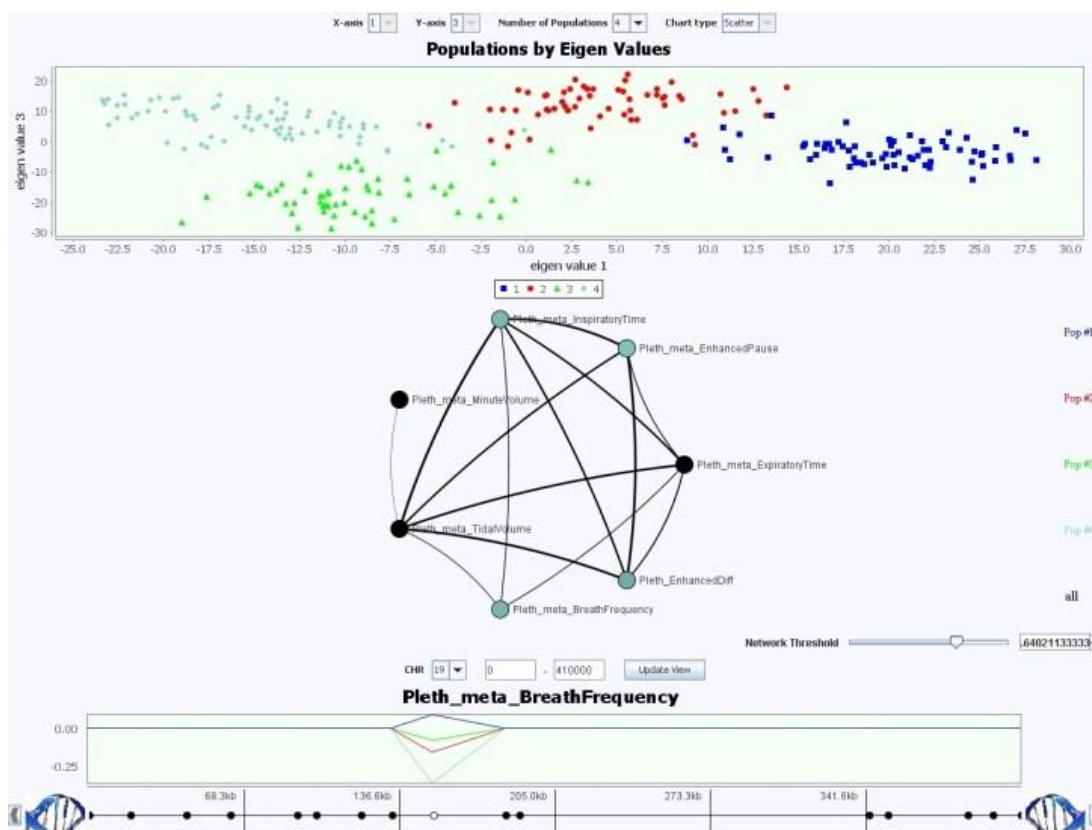


Figura 9 - GenAMap (Curtis et al., 2011).

Encontrar formas de representar conhecimento é uma questão crucial para o software científico (Ramakrishnan & Rice, 1996). Uma vez que “o cérebro humano trabalha melhor com imagens do que com caracteres” (Ntombela et al., 2005, p.247, tradução do autor), acredita-se que formas visuais possam ampliar e facilitar o *insight* (Kornbluh, 1993) – ressalva-se, no entanto, que comandos textuais podem, eventualmente, ser mais convenientes do que abordagens baseadas em representações gráficas, particularmente em trabalhos envolvendo grandes conjuntos de dados (Springmeyer, 1993).

A análise de resultados pode ser influenciada pela forma como os dados são visualizados, navegados e configurados pelo usuário. Funcionalidades desejáveis

podem depender do contexto e de necessidades específicas do domínio científico. Um apanhado de funcionalidades desejáveis inclui: tabelas de dados (Fischer et al., 2010); texto descritivo, representações visuais do objeto de estudo em duas ou três dimensões (Bitter et al., 2007; Verigan, 2007); gráficos estáticos, dinâmicos ou interativos (Murali et al., 1993); controles para navegação espacial em duas ou três dimensões (Bitter et al., 2007); exame e manipulação de objetos tridimensionais (Hu & Lill, 2014); visualização de transformação de dados ao longo do tempo (Springmeyer, 1993), como controles *time-lapse* (Eliceiri et al., 2012), visualização 4D (Bitter et al., 2007), e animações geradas a partir de simulações (Terranova & Magni, 2012); representações bidimensionais de terrenos geográficos (Savelli et al., 2010); representações tridimensionais de grandes terrenos geográficos, modelos tridimensionais ilustrativos, mapas de textura, projeção de texto sobre cena tridimensional (Feibush et al., 2000); figuras humanoides tridimensionais (Fourquet et al., 2007); ferramentas para manipulação e pós-processamento de imagem, acesso à configuração de visualização de objetos (Fischer et al. 2010, MacLeod et al., 1992). Além de todas estas funcionalidades, a exportação de imagens para geração de relatórios pode ser uma vantagem (Springmeyer, 1993).

A respeito da divisão de etapas do trabalho científico, nota-se que alguns programas podem dar suporte a mais de uma etapa (Bunus, 2006) – o que, por vezes, influencia o modo como a interface é construída. O software *Petri net Toolbox*, por exemplo, é uma extensão para o software *MatLab* que inclui dois modos diferentes de interface: o modo *desenho* (para a etapa de modelagem) e o modo *exploração* (para as etapas de simulação e análise). Através de um botão, alterna-se entre os dois modos, disparando pequenas modificações na interface e disponibilizando ferramentas e opções adequadas ao modo selecionado e desabilitando aquelas que não são (Julvez et al., 2014). Na Figura 10, observa-se, no centro e à esquerda, botões para controle da simulação que estariam desativados em modo *desenho*.

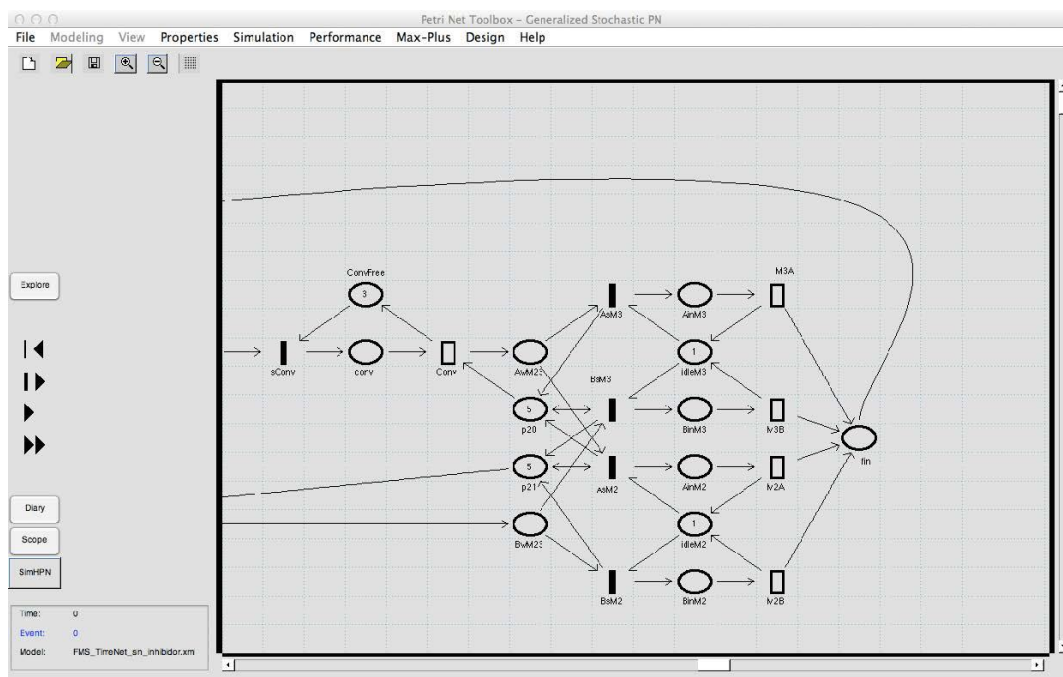


Figura 10 - Petri net Toolbox (Julvez et al., 2014)

Outro caso ilustrativo é *Biok* (Figura 11), um “aplicativo gráfico programável para biólogos” para análise de dados biológicos onde gráficos, sequências de proteínas e visualizações tridimensionais podem ter seus conteúdos e métodos de análise modificados pelo usuário através de programação (Letondal & Zdun, 2003).

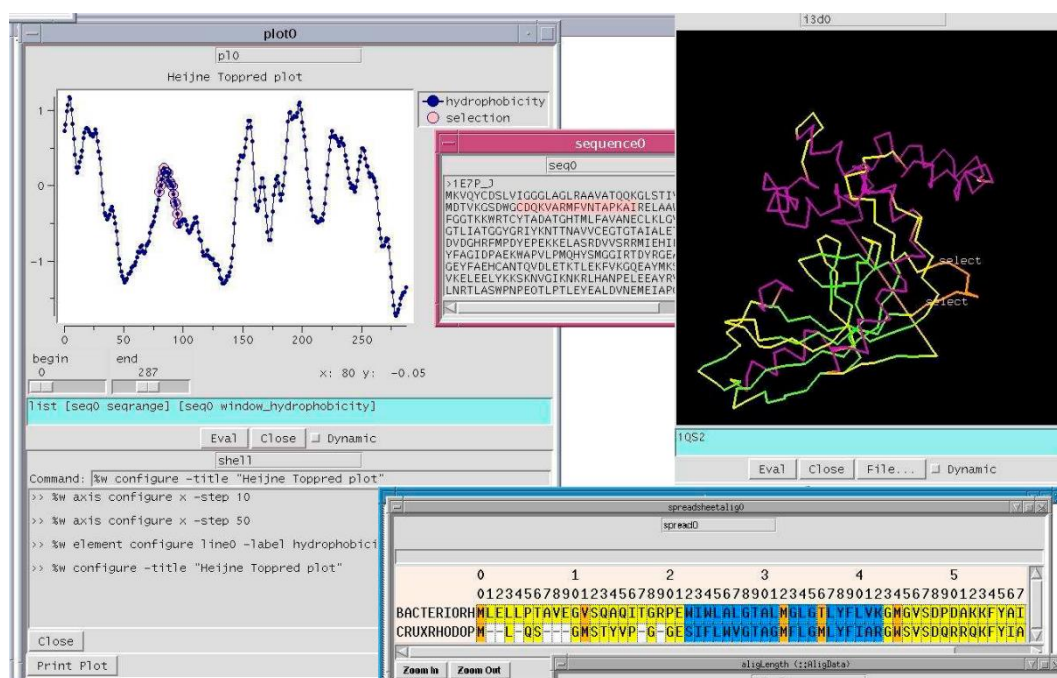


Figura 11 - Biok (Letondal & Zdun, 2003)

#### 2.2.4

#### Quanto às exigências da prática científica e profissional

A natureza da prática científica pode impor algumas condições ou obstáculos ao design de seus softwares. Embora o planejamento da interface em fases iniciais de projeto seja uma boa prática (Baxter et al., 2006), interfaces gráficas podem ter de se adaptar à *volatilidade dos requisitos*, às mudanças frequentes no código fonte (Lande, 2008) e ao crescimento contínuo do software. Esta necessidade de adaptação pode levar a soluções criativas como, por exemplo, o uso de menus expansíveis em um visualizador de bioeletricidade cardíaca que, para seu autor, “permitiram a rápida adição de submenus à medida que o código evoluiu” (McLeod et al., 1992, p.413, tradução do autor). A *complexidade dos dados*, outra característica, pode exigir a criação de métodos para a entrada e análise de um grande número de parâmetros (Fdez-Riverola et al., 2012). Também pode haver necessidade de métodos para entrada de dados com grande *precisão* (McLeod et al., 1992). Além disto, a produção de artefatos e material de design pode exigir a colaboração entre designers e cientistas (Chen et al., 2009).

Exigências profissionais também podem impactar o design. Diferentes graus de especialização dos usuários, por exemplo, podem exigir a criação de preferências de interface e modos de trabalho específicos para cada um (Gertz et al., 1994; Javahery et al., 2004), ou ainda a criação de uma barra de ferramentas para comandos frequentemente usados (Julvez et al., 2014). Neste sentido, o uso de inteligência artificial para criação de interfaces contextualizadas não parece ser uma prática usual, ainda que seja uma área de investigação há algum tempo (Brouwer-Janse, 1990). Há ainda de se respeitar hábitos de trabalho: estabelecer novas práticas pode ser um desafio devido à cultura de trabalho preexistentes (Morris & Segal, 2012). Neste caso, a adoção do software deve ser estimulada pela clareza imediata de seus benefícios, e usuários não deveriam sentir-se forçados a mudar a forma como trabalham (De Roure & Goble, 2009).

Dependendo da natureza do trabalho, a visualização de dados pode exigir uma análise crítica de dados com urgência (Aragon et al., 2008), ou providenciar indicações inequívocas de mau funcionamento de equipamentos e sistemas (Morais et al., 2014). A usabilidade parece percebida ser um fator crítico em softwares dedicados a respostas a desastres ambientais que, por sua natureza, exigem clareza



e agilidade de uso compatíveis com a urgência dos contextos de uso (Ferguson et al., 2016). Em alguns casos, a fim de preservar a familiaridade de usuários com equipamentos preexistentes, a interface gráfica pode emular a aparência física, e não apenas as funcionalidades daqueles (Foster, 1998).

### 2.2.5

#### Quanto às plataformas de uso e desenvolvimento

No que se refere ao *desenvolvimento*, a implementação de interfaces gráficas pode ser, em alguns casos, acelerada pelo uso de *frameworks* especializados – o que pode ajudar pequenos grupos de desenvolvedores a obter melhor usabilidade (Glez-Peña et al., 2010; Fdez-Riverola et al., 2012). Às vezes, softwares *open-source* permitem a personalização de interfaces preexistentes (Gorton et al., 2012) – de fato, permitir a customização e expansão das interfaces gráficas pode servir de estímulo para a adoção, e mesmo para a colaboração em desenvolvimento, por parte de outros desenvolvedores (La Rue et al., 2014).

Quanto ao *uso* do software científico, este ocorre mais em computadores pessoais (Hannay et al., 2009). Desta forma, métodos e componentes de interface são tipicamente baseados no paradigma *Windows-Icon-Mouse-Pointer* (WIMP), amplamente utilizados em PCs, permitindo o uso de várias janelas de trabalho simultâneas (McFaddin & Rice, 1992) e funcionalidades do tipo *drag-and-drop* (Manjunatha et al., 2011). Uma lista não-exaustiva de componentes típicos inclui: barras de ferramenta; mapas; menus *drop-down*; caixas de seleção (Rocha et al., 2001); menus; barras de *status*; sequenciadores numéricos (*numerical steppers*) (Lundstrom & Klimeck, 2006); paletas de ferramentas (Takatsuka & Gahegan, 2001); botões de rádio (*radio buttons*); janelas de diálogo; janelas de texto; menus não-persistentes; seções sanfona (*expandable stacks*) (Fine et al., 1998); planilhas (Willson et al., 1992); formulários (Shenoy et al., 2006); abas (Caroli et al., 2004); *sliders* (Hu & Lill, 2014); *thumbnails* de imagens; resultados numéricos (Maeda et al., 2010); janelas de hierarquia em árvore (Bitter et al., 2007).

A *web* também parece ser uma plataforma popular para software científico. Interfaces baseadas na web têm sido utilizadas há pelo menos duas décadas (Chen & Fu, 1996). No passado, a transição de um aplicativo de PC para a *web* significava comprometer a qualidade das interfaces e da usabilidade (Takatsuka & Gahegan, 2001). Hoje, navegadores *web* podem replicar ambientes tão diversos como linhas



de comando (Choi et al., 2006), softwares para *desktop* (Yamazaki et al., 2011), *websites* e plataformas *wiki* (Gorton et al., 2012) e *app stores* (Skidmore et al., 2011) – o que pode ajudar desenvolvedores a integrarem seus produtos em um ecossistema, como uma suíte de aplicativos (Turk, 2014). Além disto, interfaces *web* podem ser expandidas facilmente (Brookes et al., 2015); tornam mais fácil a adoção de interfaces gráficas padronizadas (Wolstencroft, 2015); e adaptam-se a telas de diferentes formatos, permitindo melhores experiências em *tablets* e *smartphones* (Yamazaki et al., 2011). A utilização de aplicativos *web* pode ainda eliminar problemas de instalação e compatibilidade, uma vez que o software é executado em um servidor, e não no computador pessoal do usuário. Em alguns casos, a interface gráfica de aplicativos *web* pode ser gerada e diagramada dinamicamente pelo próprio software (Wauer et al., 2004).

Cabe ressaltar que o uso de alguns softwares científicos pode exigir equipamentos especializados apropriados. Placas gráficas de alta qualidade podem ser necessárias, por exemplo, para legibilidade e visualização imersiva (Feibush et al., 2000; Kovalchuk et al., 2012). Em alguns casos, pode ser preciso permitir ao usuário ajustar configurações para atingir um ponto ótimo entre qualidade visual e performance (Fischer et al., 2010). Percebe-se ainda que a busca pela sensação de imersão não se restringe a cenários virtuais, mas tem se estendido à análise de dados quantitativos (Chandler et al., 2015; Dwyer et al., 2016). O uso de software científico pode exigir, ainda, o uso de recursos além dos tradicionais teclado, mouse e monitor, incluindo joysticks, controle por voz e plataformas de visualização 3D estereográfica (Feibush et al., 2000); telas multitoque, smartphones, equipamento para captura de movimento (Fourquet et al., 2007); câmeras de segurança (Gertz et al., 1994); *tablet PCs*, *trackballs*, controles com *force-feedback* (Keefe, 2010), e assim por diante – dependendo das necessidades de uso. Finalmente, deve-se considerar o uso combinado de diferentes plataformas e tecnologias quando necessário: uma mesma simulação poderia ser diagramada em aplicativos *web*, executada em recursos computacionais de alta performance e exibida em ambientes de realidade virtual (RV), tal como ilustrado na Figura 12 (Kovalchuk et al., 2012). Dados capturados por aparelhos médicos poderiam ser exibidos em *headsets* de realidade virtual (Dos Santos, 2016) ou extrapolados do ambiente digital, impressos em 3D (Werner et al., 2008). Interfaces híbridas, combinando PCs e ambientes

imersivos, poderiam permitir operações em modo 2D ou 3D, dependendo do contexto de uso (Carvalho et al., 2009).

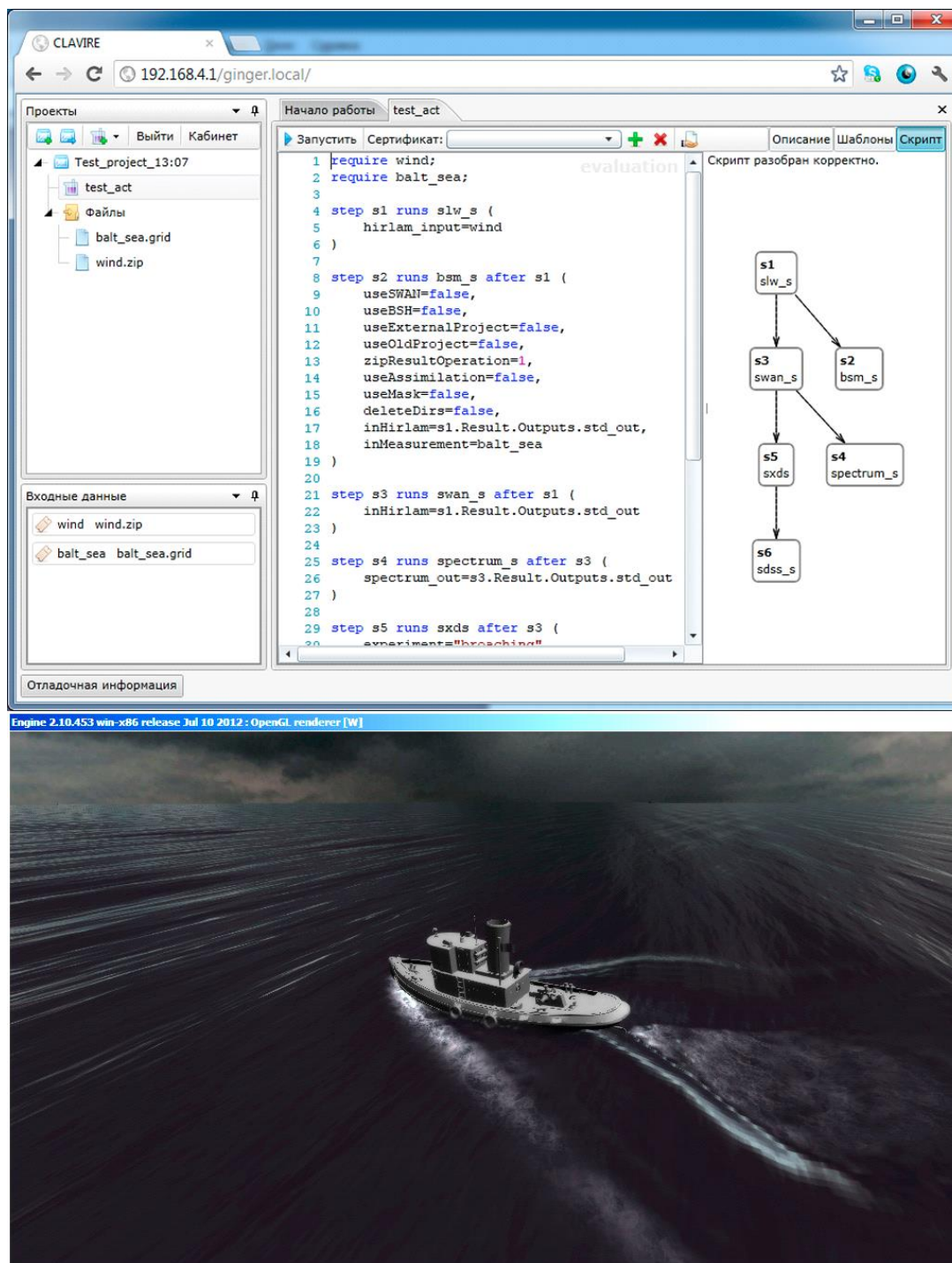


Figura 12 - Simulação em RV diagramada via web (Kovalchuk, 2012).

### 2.2.6

#### Avanços no design de interfaces para software científico

A conscientização da importância da usabilidade em software científico tem aumentado em algumas áreas (Eliceiri et al., 2012), um movimento que se percebe maior na última década. Novas abordagens à engenharia e ao desenvolvimento de software científico colocam o design de interfaces e a perspectiva do designer como questões prioritárias (Mohammad, 2010). É o caso, por exemplo, da metodologia de desenvolvimento *Butterfly*, que se preocupa em valorizar o design de interfaces desde estágios de desenvolvimento iniciais, buscando facilitar uso e aprendizado dos softwares através da compreensão da psicologia do usuário, das áreas científicas abordadas, do ambiente de trabalho, e de princípios e padrões de IHC (Ahmed et al., 2014). Esta preocupação com usabilidade se reflete no âmbito acadêmico: ao propor a criação de uma disciplina especializada no ensino de desenvolvimento de software científico, Wolff (2015) recomenda atenção à usabilidade através de pesquisas de estudo de caso e, também, da análise de similares. De acordo com Wolff, a qualidade do software reflete aspectos do desenvolvimento e do uso, incluindo a correta implementação de funcionalidades básicas, boas práticas de desenvolvimento, eficiência, eficácia, e prazer de uso. Para Venters et al. (2014), a usabilidade é considerada um dos atributos necessários à sustentabilidade do software, ao lado de extensibilidade, interoperabilidade, capacidade de manutenção, portabilidade, reusabilidade e escalabilidade.

Apesar da suposta falta de atenção à usabilidade atribuída ao desenvolvimento de software científico, pode-se encontrar um número significativo de estudos de caso e diretrizes. Tal produção, iniciada há mais de duas décadas, tem se intensificado nos últimos anos, sugerindo maturidade e maior qualidade na pesquisa e desenvolvimento de interfaces para software científico. Muitos destes estudos, alguns dos quais já mencionados neste capítulo, apresentam pesquisas detalhadas em design de interfaces, várias das quais – como veremos a seguir – sugerem o design centrado no usuário como abordagem mais propícia ao software científico – particularmente quando envolvendo métodos de design participativo e colaborativo.

Em *Improving the Usability of Numerical Software through User-Centered Design*, Cheri Pancake (1996) identifica quatro categorias para fatores de

usabilidade: facilidade de aprendizado; facilidade de uso; utilidade e eficiência. Pancake enfatiza a importância do design centrado no usuário, consistente, intuitivo e minimalista, capaz de prevenir e corrigir erros, agilizar operações e aumentar a produtividade. Para tal, propõe um processo em quatro etapas: (1) identificar requisitos de software baseados nas necessidades do usuário; (2) compreender como usuários operam em seus ambientes de trabalho; (3) projetar mudanças incrementais; (4) realizar testes com usuários para detecção de problemas.

Em *Integrating Visualization and Interaction Research to Improve Scientific Workflows*, Daniel Keefe destaca duas importantes necessidades de IHC em software científico: manuseio de dados complexos e "tarefas complexas para análises definidas por usuários motivados" (Keefe, 2010, p.8, tradução do autor). Keefe enfatiza ainda a importância da avaliação do software por parte de um *expert* no campo científico; a necessidade de priorizar-se precisão acima de velocidade de uso; e a existência de requisitos de interação específicos como seleção, navegação e criação de anotações.

Em *Developing sustainable software solutions for bioinformatics by the "Butterfly" paradigm*, Ahmed et al. (2014) argumentam que o design de interfaces deve ser guiado por princípios de IHC como experimentação, contextualização, iteração e medição empírica. O estudo propõe um processo abrangente de design, incluindo a produção de *mockups*, sessões de *brainstorming* e prototipagem informadas por padrões de design em IHC como, por exemplo, oferecer instruções passo-a-passo, desabilitar elementos irrelevantes e fornecer uma janela para cada tarefa. Também é enfatizada a importância de atenção ao domínio científico, à familiaridade com meio digital e, ainda, ao ambiente de trabalho do usuário.

Em *Ten Simple Rules for Developing Usable Software in Computational Biology*, List et al. (2017) discutem boas práticas de desenvolvimento para melhor usabilidade, dentre as quais se destacam: (1) identificar necessidades não atendidas por outros softwares; (2) coletar feedback de usuários em potencial; (3) planejar o software e interface para grande aumento de volume nos dados; (4) utilizar formatos de arquivo padrão (populares) para entrada e saída de dados; (5) a princípio, deixar usuários manipularem apenas os parâmetros obrigatórios; (6) partir do princípio de que usuários cometem erros; (7) registrar informações de uso (*logging*); (8) fazer com que usuários se iniciem rapidamente (através de instalações descomplicadas e dados de exemplo); (9) oferecer material para aprendizado; e (10) planejar o futuro:

distribuir seu código através de repositórios colaborativos e explicitar a licença de uso e modificação do software por terceiros. Além disto, recomenda-se responder ativamente a pedidos de suporte.

Questões levantadas por diretrizes como essas são, de modo geral, reforçadas pelos estudos de caso que, de forma unânime, enfatizam a necessidade de uma compreensão clara do trabalho científico e das tarefas envolvidas. Ao projetar a ferramenta de análise *MDC*, Springmeyer (1993) utilizou-se da metodologia "designer-como-aprendiz", incluindo como métodos a observação em campo e entrevistas com usuários, para entender como projetar a ferramenta em torno da forma como cientistas trabalhavam. A importância da pesquisa etnográfica também foi reafirmada pelos desenvolvedores de *Making Tea*, versão para *tablet PCs* dos cadernos de anotações de laboratórios (schraefel et al., 2004), cujo estudo reforça a importância de consulta a *experts* da área científica para maior compreensão das boas práticas de trabalho envolvidas. A avaliação do produto desenvolvido também ganha destaque, apontando para melhorias em performance, resultados e empoderamento dos usuários. Um outro estudo a respeito da adaptação de cadernos de laboratório para o meio eletrônico enfatiza a importância de funcionalidades para colaboração, criação de anotações e acesso a metadados (Talbot et al., 2005).

Em *Participatory Programming and the Scope of Mutual Responsibility: Balancing Scientific, Design and Software Commitment*, Letondal & Mackay relatam o desenvolvimento de *biok*, projetado e programado no Instituto Pasteur em colaboração com usuários, e cujo desenvolvimento se valeu de atividades de design centrado no usuário e, também, participativo, como pesquisas de opinião, observação de trabalho em campo, entrevistas, e oficinas de design participativo (Letondal & Mackay, 2004). Como resultado, *biok* permite aos próprios usuários que programem o software durante seu uso (Letondal, 2006).

Metodologias e princípios do design centrado no usuário são discutidos em *Beyond Power: Making Bioinformatics Tools User-Centered* (Javahery et al., 2004), que descreve o processo de design de ferramentas adaptáveis à experiência profissional de diferentes usuários, através de entrevistas, uso de *personas* e estudos de usabilidade.

De Roure & Goble (2009) identificaram seis princípios de design para estimular a adoção de ferramentas por usuários. Tais princípios baseiam-se, principalmente, no envolvimento direto dos usuários em um processo de design

iterativo, e podem ser sumarizados como (1) Adapte-se, não force mudanças; (2) incentive/recompense o esforço dos usuários, (3) Na hora certa e apenas o necessário, (4) aja localmente, pense globalmente, (5) deixe usuários agregarem valor, (6) projete para efeitos em rede.

Métodos de pesquisa em design foram amplamente documentados pelos criadores do software *OMERO* (Macaulay et al., 2009; Sloan et al., 2009; Loynton et al., 2009), incluindo sessões de teste, oficinas de design, demonstrações, questionários, avaliação de heurísticas e criação de material de treinamento. Os estudos enfatizam a necessidade de se compreender como cientistas trabalham e, também, como funcionam os laboratórios, além da importância do estabelecimento de prioridades e gerenciamento de expectativas dos usuários. A comunicação entre os envolvidos também recebe grande atenção através do estabelecimento de canais de troca de conhecimento entre usuários, desenvolvedores e demais participantes num processo colaborativo (Loynton et al., 2009).

O design centrado no usuário também é valorizado pelos responsáveis por *The Enzyme Portal* (De Matos et al., 2013), que utilizaram como métodos de design: levantamento de requisitos; criação de *personas*; entrevistas com usuários; análise de fluxos de trabalho; oficinas com *experts*; análises das oficinas; design e testes de protótipos de papel; especificações técnicas; protótipo interativo. Durante o processo, foram identificados os seguintes desafios: conciliar necessidades de cientistas computacionais com as de cientistas trabalhando em laboratórios tradicionais; medir níveis de *insight*; apresentar dados e metadados adequadamente; estabelecer padrões e encontrar indivíduos com conhecimento simultâneo nas áreas de IHC, computação e no domínio científico.

Em *UX DESIGN maximising the value of scientific software in life science R&D*, Cham & Costa (2017) discutem o benefício do software científico em setores de pesquisa e desenvolvimento ligados às ciências biológicas – campo para o qual o grupo *The Pistoia Alliance* desenvolve um conjunto abrangente de métodos e ferramentas para design de experiência do usuário.

Ramakrishnan & Gunter (2017) enfatizam, em seus dez princípios para usabilidade de software científico, a necessidade de equilíbrio entre o design da experiência do usuário e o processo de desenvolvimento do software. Outros princípios incluem a compreensão das motivações, métricas e contexto de uso do software pelo usuário. Além disto recomendam, como métodos para o design,

entrevistas, estudos de usabilidade e desenvolvimento de casos de uso (com a participação de usuários, se possível).

Nos últimos anos, o relato do uso de técnicas de design participativo tem aumentado em número e escopo – casos particularmente bem-sucedidos em compreender as formas de trabalho e as necessidades dos usuários, levantar requisitos e validar ideias de design. Pesquisadores da Universidade de Illinois, por exemplo, reuniram taxonomistas, desenvolvedores e cientistas da informação em um processo de design colaborativo formatado como uma *hackathon* – uma maratona de desenvolvimento onde participantes se envolvem coletivamente, por períodos de várias horas, na busca de soluções para problemas específicos ou em busca de maior criatividade. Neste caso específico, os *insights* gerados foram registrados como protótipos em papel (Thomer et al., 2016).

Baseados no processo de Macaulay et al. (2006), Rampersad et al. conduziram um processo de design participativo voltado para o desenvolvimento de uma interface para um software de visualização astronômica. No que se refere à eficácia dos protótipos gerados, observou-se que criações em papel resultaram em soluções criativas – ainda que tenham sido de difícil produção. Por outro lado, protótipos de alta fidelidade se mostraram mais eficientes para a avaliação das interfaces. (Rampersad et al., 2017)

A prototipagem em papel também foi utilizada por Luna et al. (2017), cujo processo de design participativo resultou na melhoria de interfaces para alertas de interação medicamentosa. Já Karamanis et al. (2018), ao projetarem um aplicativo *web* para pesquisa de medicamentos, organizaram oficinas de co-design, envolvendo usuários na prototipagem de interfaces.

Os casos e diretrizes identificados pela pesquisa indicam, portanto, que apesar da má reputação do software científico no que tange usabilidade e interfaces com usuário, há um esforço crescente e bem-sucedido no estabelecimento de processos de design – sobretudo colaborativos – capazes de atender às necessidades do software científico e seus usuários.

### 2.3. Gamificação, jogo e software científico

Aproximar o software científico dos jogos eletrônicos é uma ideia que remonta há aproximadamente duas décadas. Em 2000, Elias Houstis e John Rice previram que ambientes de solução de problemas (PSEs) se assemelhariam a jogos e simuladores ainda na década de 2010 (Houstis & Rice, 2000). De acordo com Rice, PSEs fariam uso de “mundos e espaços abstratos com novas regras e topologias” (Rice, 1996, p.5, tradução do autor). Além disso há, desde 2004, relatos sobre estudantes e jovens cientistas interessados em funcionalidades semelhantes às de video games em seus softwares científicos (Javahery et al., 2004).

É interessante notar afinidades entre o jogo eletrônico e o software científico – particularmente os PSEs, que há mais de duas décadas fazem uso de recursos multimídia, suporte a múltiplos usuários simultâneos e ferramentas interativas para visualização 3D (Anupam & Bajaj, 1993). Desde a década de 90, recursos semelhantes aos empregados jogos eletrônicos são celebrados e antecipados por usuários ávidos por ambientes virtuais que utilizem tecnologias “impulsionadas pelo grande mercado de jogos e entretenimento” (Kornbluh, 1993, p.74. Tradução do autor). Há, portanto, casos em que o software científico se utiliza de recursos de visualização e interação semelhantes àqueles presentes no jogo eletrônico. Há, também, casos, onde a semelhança está na forma de uso: a experimentação livre proporcionada pelo software científico – assunto ao qual retornaremos na seção 3.2 – o aproxima de um jogo. Esta associação pode ser reconhecida, a título de exemplo, em um software científico chamado de “*playground*” (Larkin et al., 2009), ou em ambientes virtuais de simulação apelidados de “*playboxes*” (Feibush et al., 2000, p.38). Neste último caso, observa-se que simulações para fins militares são frequentemente associadas a jogos (Savelli et al., 2010; Lyrio, 2011).

Apesar das semelhanças, há ao menos duas diferenças significativas entre jogo eletrônico e software científico. Em primeiro lugar, como observa Ian Bogost em *Unit operations: An approach to videogame criticism* (2006), o software científico é projetado para simulações *objetivas*, enquanto jogos buscam construir simulações *subjetivas*. No primeiro caso, valoriza-se precisão e exatidão para que se possa calcular e prever resultados que permitam compreender o sistema que está sendo simulado. No segundo caso, simulações podem ser estilizadas e distorcidas



para valorizar o design do jogo e sua fruição pelo jogador. A segunda diferença, que de certa forma é consequência da primeira, está na complexidade de desenvolvimento, configuração e uso comumente encontrada no software científico – assuntos debatidos na seção anterior – que torna mais difícil uniformizar a experiência do usuário e recriar esta mesma experiência em diferentes condições de uso (Rios, 2016). Acrescentamos ainda que, enquanto títulos individuais de jogos costumam bastar para proporcionar uma experiência completa ao jogador (à exceção de pacotes de expansão), o trabalho científico muitas vezes depende da integração ou do uso concomitante entre vários softwares.

A despeito dos desafios decorrentes das diferenças acima, há diversas iniciativas empenhadas na construção de experiências mais próximas às dos jogos. Este movimento não é exclusivo do software científico: o sucesso e o alcance do vídeo game enquanto produto da indústria cultural parecem justificar sua grande influência sobre o design de interfaces (Isbister, 2011). Ao mesmo tempo, o fato de o jogo eletrônico ter surgido há pouco mais de quarenta anos pode explicar seu iminente estabelecimento como paradigma em interatividade (Brown, 1996). Nesse contexto, há pouco mais de dez anos deu-se nome à tendência de se estruturar atividades diversas em formatos semelhantes aos jogos: a gamificação que, para Wolff (2015) poderia ser usada em software científico como forma de aperfeiçoamento da experiência do usuário. Ao longo das próximas subseções, serão discutidas definições, características e aspectos críticos desta tendência do design, particularmente quando aplicada à ciência e ao software científico.

### 2.3.1 Gamificação

A gamificação, tal como definida por Deterding et al., é “o uso de design de jogos em contextos externos a jogos” (2011, p.2., tradução do autor). Frequentemente, isso significa reestruturar uma atividade num formato semelhante ao de jogos – geralmente através da adoção de elementos estruturais de design de jogos, tais como sistemas de pontuação, *rankings* de liderança, fases, condições de vitória, etc. A essa abordagem se dá a alcunha de *gameful design* (Deterding et al., *ibid*), que se inspira no jogo enquanto atividade lúdica estruturada e voltada para a realização de objetivos preestabelecidos e medição de desempenho. O comportamento *gameful* –

termo cunhado por McGonigal (2011) – é, portanto, oposto e complementar ao *playful* que, por sua vez, descreve a brincadeira livre e desestruturada. As definições de *gameful* e *playful* assemelham-se àquelas propostas por Caillois (2001) na categorização de jogos e brincadeiras: *ludus* (estruturada, voltada a desempenho e sujeita a arbítrio) e *paidia* (livre, improvisada e despreocupada).

Ainda de acordo com Deterding et al., a gamificação é definida pelo emprego do *gameful design* – embora possa promover, também, comportamentos do tipo *playful*. (2011). Por outro lado, Popa propõe uma outra definição de gamificação: “o uso de referências oriundas de jogos para outros produtos (...) que possa prover experiências emocionais apropriadas aos usuários” (Popa, 2013, p. 17, tradução do autor). Tal abordagem parece privilegiar a plasticidade das formas de interação do jogo eletrônico – e é reforçada pela ideia de Deterding de que se deva “[traduzir] *insights* de design de jogos para design de interação” (Deterding, 2015, p.329, tradução do autor). Assim, a gamificação se vale não somente do uso de elementos motivacionais característicos de jogos, mas também de recursos visuais e interativos neles inspirados.

A título de esclarecimento, faz-se necessário destacar a diferença entre a gamificação e aquela que é outra aplicação do design de jogos em contextos além do entretenimento: os jogos sérios. Tal como a gamificação, os jogos sérios são frequentemente aplicados à educação, treinamento, ensino e saúde, entre outros campos. Deterding et al. (2011) esclarecem que, enquanto a gamificação se vale da aplicação elementos de design de jogos em outras atividades, jogos sérios são independentes de outras atividades, tratando-se de produtos fechados em si mesmo.

A gamificação emergiu rapidamente em meados da primeira década dos anos 2000, tornando-se uma tendência e um *buzzword*. Desde seu surgimento, foi considerada uma forma de se promover o entusiasmo, a diversão e o engajamento de participantes de atividades de diversas naturezas – desde questões em cidadania, educação e saúde, até organização do trabalho e fidelização de clientes. De acordo com Morschheuser et al. (2018), processos de gamificação – particularmente quando aplicados a software – podem ser divididos nas seguintes etapas: *preparação* (identificação de necessidades e requisitos do projeto); *análise contextual e de usuários* (avaliação de características do software e dos usuários); *ideação* (geração, discussão e consolidação de ideias); *design* (projeto, desenvolvimento e avaliação de protótipo, seguido de plano de desenvolvimento);

*implementação* (desenvolvimento, teste e geração de piloto); *avaliação* (avaliação e lançamento do produto); e, se necessário, *monitoramento* (para gerenciamento e identificação de melhorias). Existem diversas metodologias para a gamificação – algumas genéricas, outras dedicadas a setores específicos (Mora et al., 2015). De modo geral, o design de gamificação recorre a padrões de design de jogos (*game design patterns*) – tais como fases, pontos, turnos, medalhas, *power-ups*, etc. – para tornar as atividades-alvo mais interessantes para seus participantes (Deterding et al., 2011). Tais padrões de design podem servir a diversos propósitos: aumentar e regular a frequência com que usuários realizam uma atividade, estimular a competição e produtividade, tornar mais claro o progresso e os resultados dos participantes em determinadas tarefas, e assim por diante. Neste sentido, estes e outros *padrões em design de jogos*, tal como classificados e descritos por Bjork & Holopainen (2005), podem ser examinados isoladamente tendo em vista seus efeitos na experiência pretendida – um recurso frequentemente empregado em processos de gamificação.

De modo geral, a gamificação é considerada eficaz, ainda que se deva atentar para o contexto de sua aplicação e para que a pressão gerada por motivadores extrínsecos não prejudique a motivação intrínseca do participante (Hamari et al., 2014). Sua aplicação a ambientes de trabalho foi, por vezes, severamente criticada e acusada de ser uma forma de exploração (Bogost, 2015). Neste contexto, recomendou-se que profissionais trabalhando através de tais sistemas não substituíssem o real propósito de seus trabalhos pelos elementos motivacionais da gamificação (Kim, 2015). A este respeito, outro ponto da gamificação a ser alvo de críticas seria uma suposta redução do conceito à aplicação indiscriminada de sistemas baseados em pontos, medalhas e rankings (PBL, do inglês *Points, Badges and Leaderboards*), onde participantes têm sua performance em atividades medida, comparada e reforçada através destes elementos de design – o que favoreceria uma gratificação instantânea mas negligenciaria aspectos intrinsecamente divertidos e motivadores das atividades gamificadas. Em reação a esta tendência, pesquisadores e designers se posicionaram a favor da adoção de abordagens mais criativas e libertadoras, voltadas a experiências mais lúdicas e livres, e que se apoiassem nos valores e fatores de motivação intrínseca próprios das atividades sendo gamificadas (Deterding, 2014; Bateman, 2018) – um posicionamento ao qual esta tese se alinha. Neste caso, uma compreensão maior a respeito das atividades alvo, argumentam

Nacke & Deterding (2017), seria um dos requisitos para o amadurecimento do design de gamificação – outro requisito sendo um melhor entendimento e aplicação da natureza sistêmica dos jogos. Esta natureza é enfatizada por quadros teóricos como, por exemplo, o *Mechanics-Dynamics-Aesthetics* (MDA), de Hunicke et al. (2014), ou o *Skill Atoms* de Cook (2007). Em ambos os casos, destaca-se a relação entre as ações do jogador, as regras do jogo, e os resultados da interação entre as duas partes. Os métodos *The lens of intrinsic skill atoms* (Deterding, 2015) e *Gamification of online surveys* (Harms et al., 2014) buscam inspiração nos quadros teóricos de Cook e de Hunicke et al., respectivamente, enfatizando o caráter sistêmico dos jogos para propor maneiras de gamificação.

### 2.3.2

#### **Gamificação da ciência**

A ciência é um campo que tem se beneficiado do uso de gamificação em várias frentes. Educação e ensino de ciências têm feito uso de jogos através de sistemas gamificados de aprendizagem e, também, do crescente número de jogos de temática ligada às ciências. Acredita-se que jogos estimulem não somente a motivação em aprender, como também habilidades cognitivas (Morris et al., 2013). Em alguns casos, o jogo educacional pode ser interconectado a um software científico para a obtenção de dados necessários ao jogo (Garcia Esquirol, 2015).

Outra área de aplicação da gamificação na ciência é seu uso em redes sociais dedicadas ao *networking* entre pesquisadores e divulgação de trabalhos, cujo exemplo mais apropriado, dado ao uso explícito de elementos de *game design*, é a rede ResearchGate. Esta plataforma conta com um sistema de “conquistas” – através do qual usuários são parabenizados na medida em que seus trabalhos obtêm mais visualizações (Figura 13). Além disto, inclui um sistema de pontuação de usuários baseado no número de trabalhos enviados, perguntas e respostas feitas, entre outros critérios. No entanto, tal sistema é alvo de críticas, sendo acusada de falhar em refletir critérios que meçam a reputação acadêmica dos usuários de modo substancial (Kraker & Lex, 2015). Trata-se de uma crítica condizente com visões contrárias ao uso exagerado e arbitrário de elementos associados a motivações extrínsecas, tais como sistemas de pontos e rankings, cuja aplicação é corriqueira em sistemas arbitrários dados à exploração dos participantes.

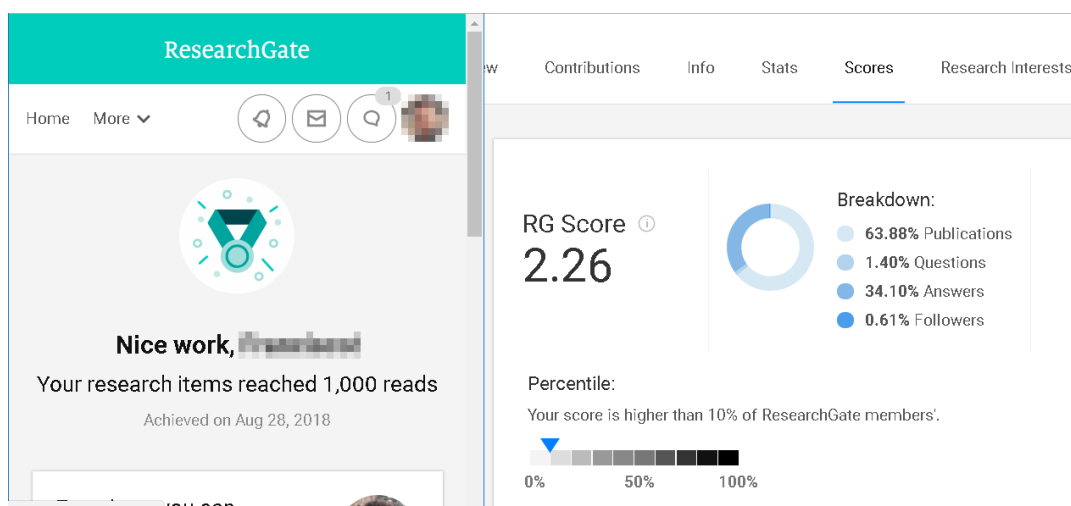


Figura 13 - Conquistas e sistemas de pontos do ResearchGate

Uma área de enorme êxito na aplicação da gamificação é a *ciência cidadã* – parte de uma cultura de colaboração em massa onde tarefas necessárias à pesquisa científica são distribuídas a um número de colaboradores massivo (podendo chegar a centenas de milhares). Estas tarefas – que podem exigir desde habilidades simples até, em alguns casos, conhecimento especializado – muitas vezes são tornadas mais atraentes através da gamificação (Franzoni & Sauermann, 2014). Um exemplo de ciência cidadã bem-sucedida é o jogo *Foldit*, projetado pela Universidade de Washington, através do qual centenas de milhares de jogadores contribuíram na solução de um problema complexo envolvendo o comportamento de proteínas (Fig.14). Para que se chegasse à solução, jogadores interagiram com modelos virtuais da proteína e, como num *puzzle*, pesquisaram formas de dobrar suas estruturas.

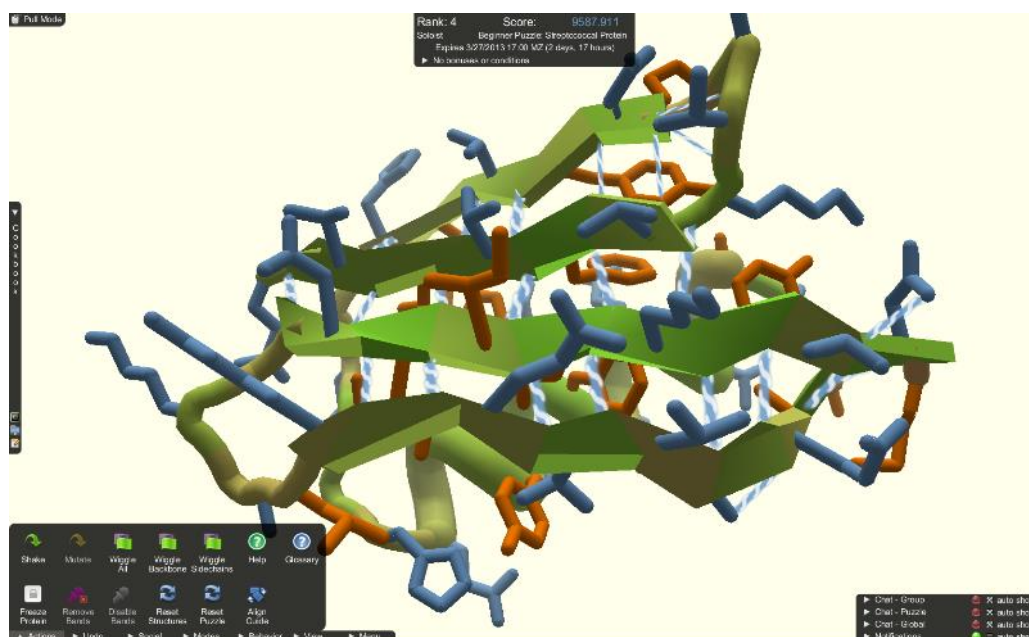


Figura 14 – Foldit, Gamificação de ciência cidadã

A ciência cidadã pode ser usada para educar, informar e coletar dados através do público geral (MacDonald et al., 2015) ou mesmo de especialistas (Good et al., 2012). Quando problemas são excessivamente complexos, a soma das habilidades dos participantes para solução de problemas pode ser mais eficaz que recursos computacionais (Cooper et al., 2010), podendo ser utilizada para melhorar a precisão de software (Mason et al., 2012). Neste sentido, como colocado por Cooper, jogos contribuem com a ciência cidadã “combinando aquilo em que humanos são bons com aquilo em que computadores são bons” (2015, p.490, tradução do autor). Ao jogar, participantes aprendem sobre tópicos avançados ao mesmo tempo em que contribuem para o avanço deste conhecimento (Devlin et al., 2014). A ciência cidadã pode estimular a motivação de seus participantes através de elementos de design de jogos; socialização (Bowser et al., 2013); cenários e situações fictícias (Prestopnik & Tang, 2015); diversão (Greenhill et al., 2014); educação e descoberta (Bowser et al., 2014); altruísmo (Schrope, 2013) e interesse prévio em ciência (Iacovides et al., 2013). Ao seu modo, a ciência cidadã parece ser o modelo mais bem-acabado do conceito, proposto por Malina, de *ciência íntima* (Malina, 2009), aproximando o público geral da atividade científica. O sucesso de iniciativas nesta linha pode ser medido por sua contribuição acadêmica (através de publicações) e, também, por seu alcance junto ao público (Simmons et al., 2015). Tal como em outras atividades gamificadas, o emprego de sistemas baseados em pontos em ciência cidadã é controverso: estima-se que sirvam como motivação para

alguns usuários (Bowser et al., 2013), mas pode ter efeito desmotivador em outros (Eveleigh et al., 2014). Neste caso, a motivação pode ser alcançada através de cenários fictícios (Prestopnik & Tang, 2015), estética bem trabalhada (Kappen et al., 2013), ou abordagens do tipo *playful*, de jogabilidade livre e exploratória (Ponti et al., 2015).

Apesar de voltada ao público geral – e, portanto, fora da definição de software científico adotada pela tese – a ciência cidadã pode ser uma boa fonte de inspiração para processos de gamificação para aquele tipo de software. Em artigo sobre a elaboração de *Foldit*, Cooper et al. (2010) discutem como os designers responsáveis pelo jogo eram simultaneamente informados pelos jogadores (a respeito da jogabilidade) e por cientistas (a respeito da ciência subjacente ao jogo). Além disto, apresentam desafios de design em interação e visualização que podem ser pertinentes também ao software científico: *Visualizações* devem tornar claras as regras dos sistemas sendo simulados; lidar com a complexidade dos modelos científicos; tornar o jogo acessível ao público leigo (mas personalizável por *experts*); *Interações* devem ser intuitivas e divertidas; respeitar as limitações do modelo científico e permitir a exploração do mesmo. *Sistemas de pontuação* devem fornecer *feedback* ao jogador e indicar quando este esteja no caminho certo. Finalmente, *fases introdutórias* devem ensinar como o jogo funciona (Cooper et al., 2010). O desafio de se desenvolver, em ciência cidadã, “interfaces acessíveis para estruturas e problemas complexos” (Cooper et al., 2013, p.1483. Tradução do autor) parece ser comum ao software científico, assim como outros pontos levantados por diferentes autores: Bowser et al. alertam para a necessidade de se preservar a qualidade dos dados (2013). Jennet & Cox destacam atributos desejáveis em ciência cidadã, tais como: clareza; material de suporte ao aprendizado; funcionalidades que ajudem usuários a completar tarefas; atenção à diversidade de níveis de especialização dos usuários (2014).

### 2.3.3 Gamificação de software científico

Elementos estruturais de design de jogos têm sido aplicados na construção de comunidades de desenvolvimento e uso de software científico (Katz, 2015). *BioStar* (Blankenberg et al., 2015, Afgan et al., 2016), por exemplo, é uma plataforma para

*forums* gamificados de perguntas e respostas no qual participantes ganham pontos de reputação na medida em que contribuem com informações e respondem a dúvidas de outros usuários (Fig.15) – um formato inspirado no popular fórum de desenvolvimento e programação *StackOverflow*.

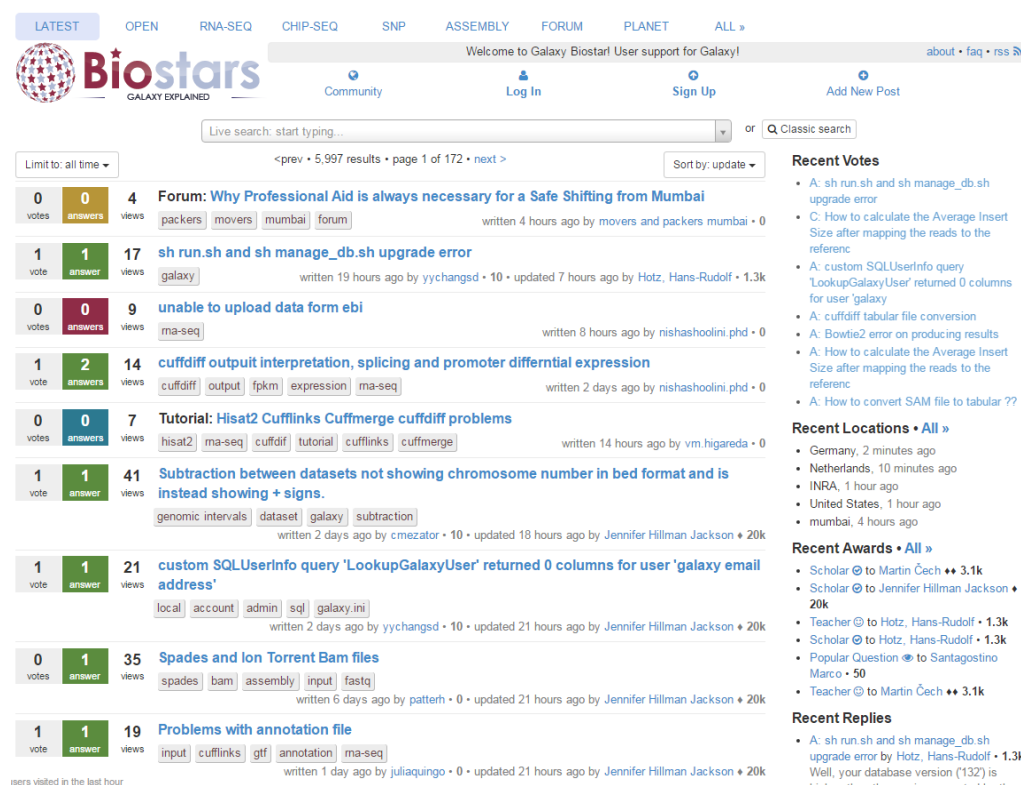


Figura 15 - BioStar (biostar.usegalaxy.org)

Já a o projeto *BioDynamo*, que busca desenvolver uma plataforma para simulação biológica, promoveu uma competição entre estudantes, criando um ranking de soluções em otimização de código e premiando os autores das melhores com um estágio na Organização Europeia para a Pesquisa Nuclear (CERN) (Breitwieser et al., 2016). De fato, sugere-se que a gamificação poderia auxiliar outros aspectos – além da otimização – do desenvolvimento de software, tais como processos de documentação (Pawlik et al., 2015), teste e verificação (Mao et al., 2017), e levantamento de requisitos (Fernandes et al., 2012) e medição de sustentabilidade (Druskat, 2016). No entanto, o presente trabalho se concentra, para efeitos práticos, em questões ligadas ao uso dos softwares produzidos – e não processos de desenvolvimento – ainda que reconheça a proximidade entre desenvolvimento e uso de software científico.



No que se refere ao *uso* do software, dentro do espectro de aplicações profissionais e acadêmicas nos campos de ciência, tecnologia, engenharia e matemática (STEM), o setor de engenharia parece ser aquele que assimilou a gamificação em seus produtos em maior velocidade e escala. Em alguns destes casos, o uso da gamificação é voltado ao treinamento e educação. Uma equipe de pesquisa da Autodesk, por exemplo, desenvolveu o *GamiCAD* (Fig. 16), tutorial gamificado para *AutoCAD* que faz uso de missões, narrativa, elementos visuais e indicadores de progresso (Li et al., 2012).

**Mission 1**  
Saturn V is the tallest, heaviest and most powerful multistage liquid-fueled launch rocket in the human history. You are now in charge of designing several important parts using AutoCAD LINE command.

Task	Best Score
1. Draw a line.	231
2. Draw a triangle.	228
3. Draw a line (2 points).	194
4. Draw a rectangle.	225
5. Draw a fin.	459
6. Draw a burner.	300
7. Connect circles	722

Launch escape motor  
Command module  
2nd stage motor  
1st stage motor  
Fin  
Burner cups  
Frame

← Start Page Bonus Mission →

Figura 16 - Missão em *GamiCAD* (Li et al., 2012)

Em outra experiência relacionada a educação, pesquisadores da Universidade de Calgary investigam como jogos podem ser utilizados no ensino de engenharia, propondo a alunos que criem jogos sobre técnicas de design de placas de circuito (Marasco et al., 2015).

Aplicações de CAD (*Computer-Aided Design*) e BIM (*Building Information Modeling*) aplicam tecnologias, mecânicas e estética de jogos para melhorias em usabilidade (Kosmadoudi et al., 2013). Pesquisadores da Universidade Chinesa de Hong Kong desenvolveram *ModRule*, software que explora aspectos colaborativos

da gamificação para promover o diálogo entre arquitetos e potenciais compradores no design de unidades de habitação (Lo et al., 2014). No entanto, a maior influência dos jogos eletrônicos sobre aplicações em engenharia parece ser a criação de experiências mais intuitivas e atraentes do que as proporcionadas por ferramentas convencionais (Boeykens, 2011). Tais experiências tiram proveito das possibilidades estéticas e imersivas proporcionadas pelas tecnologias e design de jogos (Aydin & Schnabel, 2014; Moloney, 2015). Neste contexto, através de softwares de produção de jogos (*engines*), modelos de arquitetura e engenharia podem ser transformados em ambientes imersivos, com suporte a realidade virtual e aumentada (Keenaghan & Horvath, 2014).

Tais qualidades imersivas, alcançada através da adoção de tecnologias e padrões de interatividade típicos dos *video games*, se mostram como a alternativa mais prolífica em gamificação de software científico. Um bom exemplo desta abordagem é o *Virtual Astronaut* (VA), ambiente tridimensional projetado para análise geovisual virtual da superfície de Marte (Fig.17), desenvolvido através de uma *engine* – software especializado para construção de jogos (Wang et al., 2012).



Figura 17 - Virtual Astronaut (Wang et al., 2012)

Uma *engine* foi utilizada, também, na construção de ambientes de realidade virtual em estudos de entomologia e diagnóstico por imagem (Bergmann et al.,

2017), providenciando um ambiente imersivo para estas áreas de pesquisa – apesar de limitações em interatividade. Tecnologias em realidade virtual – apoiadas por técnicas tradicionais em diagnóstico por imagem – foram aplicadas também à medicina, possibilitando a exploração de modelos tridimensionais de fetos (Dos Santos et al., 2016). Por sua vez, Hutson et al. (2016) relatam uma experiência em design colaborativo para uma nova versão da interface de *Aquaria*, ferramenta web dedicada à bioinformática. Tal experiência resultou em *insights* para funcionalidades inspiradas em jogos que, apesar de não terem sido implementadas, facilitariam aspectos da visualização e colaboração do software.

Por outro lado, apesar do entusiasmo inicial em torno da gamificação, a gamificação de software – científico ou não – através da reestruturação de suas atividades em formatos semelhantes a jogos não parece ter vingado em grande (ou média) escala: Darejeh & Salim (2016) realizaram uma revisão sistêmica acerca da gamificação de software e seus efeitos no engajamento de usuários. O estudo em questão identificou somente dois artigos sobre iniciativas de gamificação de software de produtividade, ou seja, voltado ao trabalho (categoria na qual o software científico, voltado à pesquisa ou tecnologia, se encaixaria) – um destes artigos, aliás, se reportava ao já mencionado *GamiCAD* (Li et al., 2012). Há, porém, um aspecto ainda mais restritivo: ambos os softwares identificados fazem uso da gamificação no aprendizado do software, e não seu uso contínuo – o que indica uma dificuldade na gamificação do trabalho realizado em software de modo geral.

## 2.4

### Conclusão do capítulo

Ao longo deste capítulo, buscou-se reconhecer e descrever peculiaridades do desenvolvimento e uso de software científico, a fim de se identificar os processos de design mais adequados àquelas características. Buscou-se também apresentar o estado da arte em design de usabilidade e gamificação para este tipo de software. As próximas subseções apresentam um sumário das informações obtidas para, ao final, sugerir uma abordagem à gamificação do software científico que se mostre harmônica às suas idiossincrasias de uso e desenvolvimento, aos métodos de design mais adequados e bem-sucedidos e, também, a uma visão sistêmica dos jogos e o aproveitamento de características lúdicas do trabalho em ciências.

### 2.4.1

#### Características de uso e desenvolvimento

Foram identificadas particularidades no uso e desenvolvimento do software científico capazes de impactar seu design de interfaces e gamificação – e em alguns casos impulsionando a geração de soluções criativas. Esta relação de características será determinante na construção das ferramentas do método de gamificação proposto pela tese, informando o processo de design.

Tais particularidades podem ser organizadas dentro das seguintes categorias: desenvolvimento; usuários; aspectos profissionais; aspectos científicos e acadêmicos; aspectos de software; interface com usuário; etapas do trabalho científico.

Questões de *desenvolvimento* incluem: tamanho da equipe de desenvolvimento; colaboração entre membros das equipes; metodologias de desenvolvimento; prazos; vida útil do software; tecnologias de desenvolvimento; escopo e necessidade de expansão do projeto. Isso sugere que processos de design e gamificação devem ser flexíveis e considerar criteriosamente custos de implementação, adequando-se ao tamanho e recursos das equipes, além de serem planejados para desenvolvimento iterativo, priorizando necessidades atuais do software.

Questões relativas aos *usuários* incluem: tamanho e necessidades da base de usuários; diversidade de tipos e níveis de especialização; familiaridade com ambientes digitais; expectativas dos usuários; empoderamento através de participação ativa e colaborativa no desenvolvimento; construção de comunidades. Assim, deve-se buscar contemplar diversos perfis de usuário – possivelmente buscando diversidade nas formas de gamificação. Pode ser interessante, ainda, tornar o próprio processo de design colaborativo.

*Aspectos profissionais* envolvem práticas, condições, cultura, normas, convenções, e ética no trabalho – todas devem ser respeitadas por processos de design. Sugerem ainda a necessidade de consulta e avaliação de *expert* da área científica.

*Aspectos científicos e acadêmicos* a serem observados incluem: motivação; rigor e precisão; reprodutibilidade; complexidade e qualidade de dados; dificuldade

na obtenção de requisitos. Enquanto o primeiro aspecto aponta para possíveis abordagens aos objetivos propostos pela gamificação, os seguintes apontam para desafios projetuais. O último aspecto listado sugere possíveis dificuldades no próprio processo de *design*.

Questões relativas a *aspectos do software* sugerem dificuldades no processo de desenvolvimento e acabamento de recursos visuais e de interatividade complexos, e incluem: necessidade de expansão; desenvolvimento incremental; portabilidade; performance; facilidade de instalação e configuração; automação; adequação aos requisitos de sistemas onde será instalado.

Em relação à *interface com o usuário*, destacam-se as seguintes questões: facilidade de uso e aprendizado; adequação à plataforma de uso; possibilidade de customização; flexibilidade; consistência; minimalismo; mudanças incrementais; entrada, saída e monitoramento de dados complexos; precisão; criação de anotações; acesso à métodos para programação; acesso a bases de conhecimento; prevenção e correção de erros; repetição e registro de ações; geração de material para relatórios; acesso a metadados; modos de operação sem interface gráfica.

Finalmente, em relação às *etapas do trabalho científico*, destacam-se as seguintes necessidades: produtividade na composição de *workflows* e modelos; geração de *insights* através da análise de resultados apoiada na visualização, navegação e manipulação de dados em várias dimensões; contextualização e integração adequadas entre etapas de modelagem, simulação e análise de resultados.

## 2.4.2

### Design de usabilidade e gamificação

Apesar das críticas ao estado geral da usabilidade em software científico, identificou-se um número significativo de projetos e pesquisas em design de interfaces muito bem realizados e fundamentados. De fato, o design de interfaces neste campo parece, em alguns casos, ser mais incompreendido do que necessariamente deficitário. Parece haver, ainda, uma tendência no aumento de projetos e pesquisas voltados a melhorias no design de software científico à medida que metodologias de desenvolvimento voltadas ao campo se tornam mais robustas. Parece haver, também, uma crescente conscientização da necessidade de

investimento em usabilidade. De modo geral, com base no material investigado, pode-se afirmar que os estudos de caso e diretrizes apontam para uma mesma direção: a necessidade de um design voltado para a usabilidade que seja informado pelas necessidades inerentes ao desenvolvimento e uso do software científico em ambiente profissional, e apoiados por processos cuidadosos de design – particularmente aqueles que buscam uma abordagem centrada no usuário, envolvendo-os ao longo do desenvolvimento através de experiências em design participativo e co-design. Este envolvimento se mostra essencial ao observarmos o grau de especialização e especificidade a que pode chegar o software científico.

Quanto à gamificação, percebe-se duas possíveis abordagens à sua aplicação: (1) o uso de elemento de design estruturais que confirmam uma estrutura de jogo à atividade (através de missões, pontos, condições de vitória, etc.); e (2) uso de tecnologias, estéticas, recursos de interatividade e visualização semelhantes àqueles proporcionados por jogos. A revisão de literatura indica que a segunda abordagem tem se mostrado mais produtiva no campo do software científico – embora a primeira se mostre eficaz no desenvolvimento de aplicativos para ciência cidadã, cujos produtos adotam formatos estruturalmente semelhantes aos de jogos em sua clareza de objetivos, divisão de tarefas, indicações de progresso e performance, e até mesmo, em uma apresentação mais polida. Um motivo possível para a dificuldade na aplicação de uma abordagem estrutural pode ser a dificuldade em se conciliar o processo de desenvolvimento do software (que, como vimos, pode ser errático) com o design e implementação de um sistema gamificado que exige, por si só, diferentes etapas de desenvolvimento. De fato, a escassez de exemplos de software para trabalho e produtividade gamificados sugere que, passada mais de uma década do *boom* da gamificação, estimativas iniciais quanto à sua disseminação parecem, em retrospecto, exageradas. Tal escassez poderia reforçar, também, os argumentos que relacionam a gamificação à exploração dos participantes: se, por um lado, sistemas gamificados são utilizados em contextos profissionais para medir e estimular a produtividade, por outro, parecem não ser incorporados às ferramentas através das quais o trabalho é realizado para torná-las mais fáceis, eficazes ou agradáveis.

### 2.4.3

#### Diretrizes para a elaboração do método: do software e seu desenvolvimento

A partir da análise das informações apresentadas, pode-se vislumbrar diretrizes para a construção de um método para a gamificação adequada ao software científico. Em princípio, tal método deve: (1) ser adaptável ao *processo de design de interfaces* e usabilidade para esse tipo de software; (2) considerar *características peculiares* de uso e desenvolvimento; (3) atentar para a *natureza sistêmica* dos jogos; (4) considerar os *aspectos lúdicos da pesquisa* e trabalho científico. Em uma comparação entre o método a ser construído e um jogo, as diretrizes 1 e 3 seriam análogas às regras, definindo sua dinâmica e funcionamento. Por outro lado, as diretrizes 2 e 4 seriam comparáveis às peças, que devem ser manipuladas para que se atinjam os objetivos.

Questões referentes às diretrizes 1 e 2 foram averiguadas ao longo das seções anteriores, através de um levantamento das características de uso e desenvolvimento, onde tomou-se o cuidado de identificar aspectos potencialmente problemáticos e, também, destacar particularidades importantes. Parece claro, portanto, que *processos de design para software científico devem manter em evidência as características deste tipo de software*, para que usuários, designers e desenvolvedores estejam alertas às oportunidades e desafios.

Por outro lado, em relação aos processos de design de interfaces, reforçou-se a importância e eficácia dos métodos colaborativos e participativos em atender às necessidades do software científico em processos de desenvolvimento iterativos. É através da ação pontual e local às necessidades emergentes do software e da pesquisa – mais do que de um extenso planejamento de implementação – que melhorias podem ser incorporadas. Por esse motivo, esta tese defende a hipótese de que *o processo de design de gamificação adequado ao software científico deve adotar a forma de um método de design participativo e centrado no usuário*.

As diretrizes 3 e 4, no entanto, permanecem em aberto e implicam duas questões: quais são os aspectos lúdicos do trabalho científico sobre os quais a gamificação deve se apoiar, e como encará-los de maneira sistemática? No próximo capítulo, estas questões serão tratadas através de uma discussão sobre paralelos entre jogo e ciência e, também, de uma interpretação sistemática para elementos fundamentais do jogo.

### 3

## Jogo e ciência

Em 2010, Andre Geim e Konstantin Novoselov receberam o prêmio Nobel de Física por sua pesquisa sobre o grafeno, material inovador que, além de ser o mais forte já identificado, caracteriza-se por ótima flexibilidade, transparência, condução de calor e eletricidade. Possuindo apenas um átomo de espessura, o grafeno é considerado, também, o mais leve e fino dos materiais. As qualidades e potencial das aplicações do grafeno na indústria renderam-lhe o apelido de ‘material milagroso’ (Novoselov et al., 2012). A respeito de Geim e Novoselov, a Academia Real de Ciências da Suécia publicou:

“Uma das marcas registradas [de Andre Geim e Konstantin Novoselov] é o espírito lúdico: com ele, sempre se aprende algo com o processo. Às vezes, pode-se até tirar a sorte grande. Como agora quando, com o grafeno, inscrevem-se nos anais da ciência.” (Academia Real de Ciências da Suécia, 2010, p.1, tradução do autor)

A afirmação reproduzida acima serve como uma boa ilustração do argumento que será apresentado neste capítulo: vários aspectos – tais como o espírito lúdico, a experimentação e a busca por recompensas – aproximam o ato de jogar ao de fazer ciência. Propomos, portanto, a exploração dessa afinidade como um caminho natural e adequado para a gamificação do trabalho científico e, consequentemente, do software científico. Nossa argumentação será construída, ao longo das próximas seções, da seguinte forma: *Usos prévios do jogo como modelo para ciência* discute obras que comparam extensivamente estas duas atividades, utilizando jogos como modelo para a descrição da prática científica. Em seguida, *Uma definição clássica como base para o modelo* apresenta a definição de jogos que servirá de base para esta análise comparativa e, também, para a visão sistêmica à qual o método a ser proposto será orientado. As três seções seguintes, *Regras e realidades*, *Medidas e valores* e *Esforço e recompensa* apresentam, efetivamente, as questões centrais do capítulo, discutindo a ciência nos termos apresentados pela definição formal de jogo, a partir das obras de referência. Finalmente, a *Conclusão do capítulo* oferece



uma síntese e reflexão sobre o conteúdo apresentado, e seus impactos sobre processos de gamificação baseados em uma visão sistêmica de jogos.

Este capítulo é baseado em pesquisa bibliográfica a obras e autores selecionados. Tal pesquisa buscou recolher informações sobre a natureza e condições do trabalho científico a partir da análise de obras das da filosofia da ciência, história da ciência, antropologia e psicologia. Através desta pesquisa bibliográfica, buscamos uma visão mais abrangente e aprofundada da atividade científica – seus aspectos profissionais, ontológicos e epistêmicos – do que seria possível alcançar através somente da revisão de literatura e do estudo de campo.

O estudo aborda obras publicadas a partir do início do século XX até o ano de 2014, selecionadas por suas reconhecidas contribuições às respectivas áreas de estudo e, também, por afinidade temática com o tema de pesquisa. A relação de obras escolhidas, listadas em ordem cronológica de primeira publicação, está reproduzida a seguir (obras com edições em português estão listadas naquele idioma, independentemente da versão consultada).

- *Ciência e Hipótese*, de Henri Poincaré (1902).
- *O Valor da Ciência*, de Henri Poincaré (1905).
- *Ciência e Método*, de Henri Poincaré (1908).
- *O Novo Espírito Científico*, de Gastón Bachelard (1934).
- *A Lógica da Pesquisa Científica*, de Karl R. Popper (1934).
- *A Formação do Espírito Científico*, de Gastón Bachelard (1938).
- *A Filosofia do Não*, de Gastón Bachelard (1940).
- *A Estrutura das Revoluções Científicas*, de Thomas S. Kuhn (1962).
- *Conjecturas e Refutações*, de Karl R. Popper (1963).
- *A Dupla Hélice – Como Descobri a Estrutura do DNA*, de James D. Watson (1968).
- *Contra o Método*, de Paul Feyerabend (1975).
- *A Tensão Essencial*, de Thomas S. Kuhn, (ensaios produzidos entre 1959 e 1976. Publicado em 1977)
- *A Ciência em uma Sociedade Livre*, de Paul Feyerabend (1978).
- *A Vida de Laboratório – A Produção dos Fatos Científicos*, de Bruno Latour e Steve Woolgar (1979).

- *The Manufacture of Knowledge: An Essay on the Constructivist and Contextual Nature of Science*, de Karin Knorr Cetina (1981).
- *How the Laws of Physics Lie*, de Nancy Cartwright (1983).
- *Representar e Intervir: Tópicos Introdutórios de Filosofia da Ciência Natural*, de Ian Hacking (1983).
- *Reflections on Gender and Science*, de Evelyn Fox Keller (1985).
- *Adeus à Razão*, de Paul Feyerabend (1987).
- *How Experiments End*, de Peter Galison (1987).
- *A Ciência em Ação: Como Seguir Cientistas e Engenheiros Sociedade Afora*, de Bruno Latour (1987)
- *All Life is Problem Solving*, de Karl R. Popper (ensaio produzido entre 1958 e 1994. Publicado em 1994).
- *The Second Self: Computers and the Human Spirit*, de Sherry Turkle (1995).
- *A Descoberta do Fluxo: Psicologia do Envolvimento Com a Vida Cotidiana*, de Mihaly Csikszentmihalyi (1996).
- *Creativity: Flow and the Psychology of Discovery and Invention*, de Mihaly Csikszentmihalyi (1996).
- *Epistemic Cultures: How the Sciences Make Knowledge*, de Karin Knorr Cetina. (1999).
- *O Caminho desde a Estrutura*, de Thomas S. Kuhn (ensaio produzido entre 1970 e 1993. Publicado em 2000).
- *Good Work – When Excellence and Ethics Meet*, de Howard Gardner. Mihaly Csikszentmihalyi e William Damon (2001).
- *Making Sense of Life: Explaining Biological Development with Models, Metaphors, and Machines*, de Evelyn Fox Keller (2003).
- *Objectivity*, de Lorraine Daston e Peter Galison (2007).
- *Creating Scientific Concepts*, de Nancy J. Nersessian (2008).
- *Simulation and its Discontents*, editado por Sherry Turkle (2009).
- *Cogitamus, Seis cartas sobre as humanidades científicas*, de Bruno Latour (2010)
- *Representation in Scientific Practice Revisited*, editado por Catelijne Coopmans, Janet Vertesi, Michael Lynch e Steve Woolgar (2014).

É objetivo deste capítulo, portanto, a identificação de paralelos entre jogo e ciência. Ao longo do processo de elaboração do texto, não se buscou tecer uma comparação histórica que avaliasse ou registrasse uma evolução cronológica dos temas pesquisados, mas sim a identificação de aspectos relevantes à prática científica os quais pudessem ser discutidos sob perspectivas teóricas do design de jogos, gamificação e *game studies*. O resultado da argumentação, portanto, não é uma análise comparativa entre o pensamento de diferentes autores, mas a exposição e o mapeamento de conceitos como insumo para a produção de um método de design. Assim, o que buscamos inicialmente não é uma síntese, mas um mosaico – uma coleção de visões que – ainda que contraditórias umas em relação às outras – permitam uma visão mais ampla do que a ciência pode ser ou significar. Entretanto, esperamos que a discussão que fecha este capítulo deixe claro que a visão sistêmica a ser proposta pela tese como base para método de gamificação é mais compatível com determinadas perspectivas do que com outras.

### 3.1

#### Usos prévios do jogo como modelo para a ciência

Originalmente escrito em 1938 pelo historiador holandês Johan Huizinga, *Homo Ludens - o jogo como elemento da cultura* é uma obra seminal na discussão da importância e papel dos jogos no desenvolvimento da civilização (Huizinga, 2004). De acordo com Huizinga, é a partir do espírito lúdico que se desenvolvem e se organizam atividades arquetípicas humanas como a guerra, o direito e a poesia. Apesar de conceder a possibilidade de um elemento lúdico na sistematização da ciência moderna, Huizinga argumenta que "a afirmação de que a ciência não passa de um jogo é uma afirmação gratuita, demasiado fácil, que se impõe descartar provisoriamente" (2004, p.226). Para ele, há duas questões que afastam a ciência do jogo: em primeiro lugar, a forte conexão entre ciência e a realidade à qual tenta ser útil ou explicar. Em segundo lugar, o caráter provisório das regras da ciência. É revelador, no entanto, a frequência em que a ciência é comparada ao jogo – às vezes de forma explícita e extensa – pelos autores que servem de base à presente pesquisa.

O objetivo desta discussão, no entanto, não é afirmar categoricamente que a ciência seja um jogo — nem estabelecer que a ciência moderna tenha se desenvolvido a partir do espírito lúdico. Tais questões são pertinentes e passíveis

de debate, e embora possam estar sugeridas neste trabalho, não buscaremos comprová-las. O que buscaremos é reunir características do trabalho científico que estejam presentes ou encontrem equivalentes no jogo e na brincadeira e que, assim, o jogo sirva como modelo para o design voltado à ciência. Como ponto de partida, analisaremos o que define o jogo e, a partir de suas características definidoras, discutiremos elementos semelhantes ou aproximados encontrados no trabalho científico ou, ainda, no software científico, tais como apontados pelos autores de referência. Ao final da discussão, esperamos encontrar elementos do trabalho científico que possam servir de base à gamificação do software científico.

O uso do jogo como modelo ou metáfora para ciência não é inédito e, neste sentido, duas obras merecem destaque. A primeira delas é *The Game of Science*, de Garvin McCain e Erwin M. Segal (1989), cuja primeira edição foi publicada em 1969. Trata-se de um texto introdutório à ciência para o público leigo, apresentando o tema a partir de conceitos oriundos de jogos, como regras (o método científico), jogadores (cientistas), atividades (produção de ideias, teorias, experimentos), motivações (curiosidade, desafio, deleite, exercício intelectual) e resultados (efeitos na sociedade, relação com tecnologia, relação com fontes de financiamento). Algumas das analogias apresentadas se assemelham às desta tese, particularmente as comparações entre regras e método científico, a produção de teorias e experimentos como atividades, e o deleite e desafio como fatores motivacionais.

A segunda obra merecedora de destaque é o ensaio *Getting the game right: Some plain words on the identity and invention of science*, de Andrew Cunningham (1988), cujo objetivo é estabelecer critérios através dos quais historiadores possam ter certeza de que manifestações passadas de prática científica se classifiquem, de fato, como ciência (e não, por exemplo, como filosofia da natureza, predecessor espiritual da ciência). É central, na argumentação de Cunningham, a importância das regras que compõem a atividade estruturada da prática científica, e às quais cientistas devem obedecer de forma consciente e intencional para cumprir objetivos através de procedimentos. Destaca-se também a natureza dos jogos e da ciência como atividades criadas e realizadas por seres humanos.

Tanto Cunningham quanto McCain & Segal defendem a caracterização da ciência como jogo. No entanto, é curioso que, em ambos os casos, tais caracterizações prescindam da apresentação de uma definição formal do termo

‘jogo’ a qual possa ser comparada com os elementos da prática científica. Para Cunningham,

“Os extensivos paralelos entre jogos e a prática científica mostram que a prática científica pode ser legitimamente considerada possuidora do mesmo tipo de natureza, características e estrutura de um jogo” (1989, p.377, tradução do autor).

McCain & Segal, por outro lado, afirmam que “por causa das similaridades entre as atrações da ciência e aquelas apresentadas por vários jogos, podemos considerar a ciência como um jogo” (1989, p.4, tradução do autor). Porém, tanto os paralelos apresentados por Cunningham quanto as atrações listadas por McCain & Segal parecem, ainda que abrangentes e inclusivos, arbitrários – uma deficiência que o presente estudo se propõe a contornar.

### 3.2

#### Uma definição clássica como base para o modelo

Em busca de maior objetividade, esta tese adota como fio condutor de sua argumentação a definição clássica proposta por Jesper Juul, teórico e pesquisador de jogos, em seu livro *Half Real: Video Games between Real Rules and Fictional Worlds* (2005). Tal escolha se deve, principalmente, a dois motivos: (1) trata-se de uma definição relativamente recente que, embora não seja limitada a jogos eletrônicos, contempla aspectos do *video game* que poderiam ser ignorados por outros quadros teóricos; (2) é uma definição construída a partir de sete definições anteriores, propostas por diversos autores, o que a torna mais completa e abrangente. De acordo com Juul,

"o jogo é um sistema baseado em regras, de resultados variáveis e quantificáveis, onde diferentes valores são atribuídos a diferentes resultados, o jogador se esforça para influenciar o resultado, o jogador se sente emocionalmente ligado ao resultado e as consequências da atividade são negociáveis" (Juul, 2005, p.36. Tradução do autor).

Há, na definição de Juul, uma limitação que deve ser contornada: a definição escolhida exclui atividades lúdicas que contradigam a terceira característica, ou seja: não atribuem valores aos resultados, promovendo assim comportamentos do tipo *playful*, e não do tipo *gameful*. É preciso reconhecer, no entanto, que há produtos e gêneros, dentro dos *video games*, que abraçam esta característica, como simuladores livres dos quais o mais emblemático exemplo talvez seja *Sim City*, definido por seu criador como um "brinquedo de software" (Rouse III, 2001), e que

permite ao jogador criar e desenvolver cidades virtuais sem se preocupar com condições de vitória. Juul reconhece tais produtos e gêneros como casos-limítrofes entre o que pode ou não ser considerado jogo (Juul, 2005). Neste caso, é importante lembrar que, tal como discutimos anteriormente, atividades do tipo *gameful* podem suscitar comportamentos do tipo *playful* (Deterding et al, 2011), os quais também serão abordados ao longo desta análise.

No que se refere à qualidade sistemática dos jogos, a definição de Juul poderia ser descrita como um sistema onde jogadores aplicam esforço sobre um sistema que calcula, baseado em regras, os resultados deste esforço – resultados estes que informam ao jogador sobre seu desempenho e orientam seus próximos passos. Para uma melhor visualização de seus aspectos sistêmicos, produzimos uma ilustração diagramática, que pode ser vista na Figura 18.

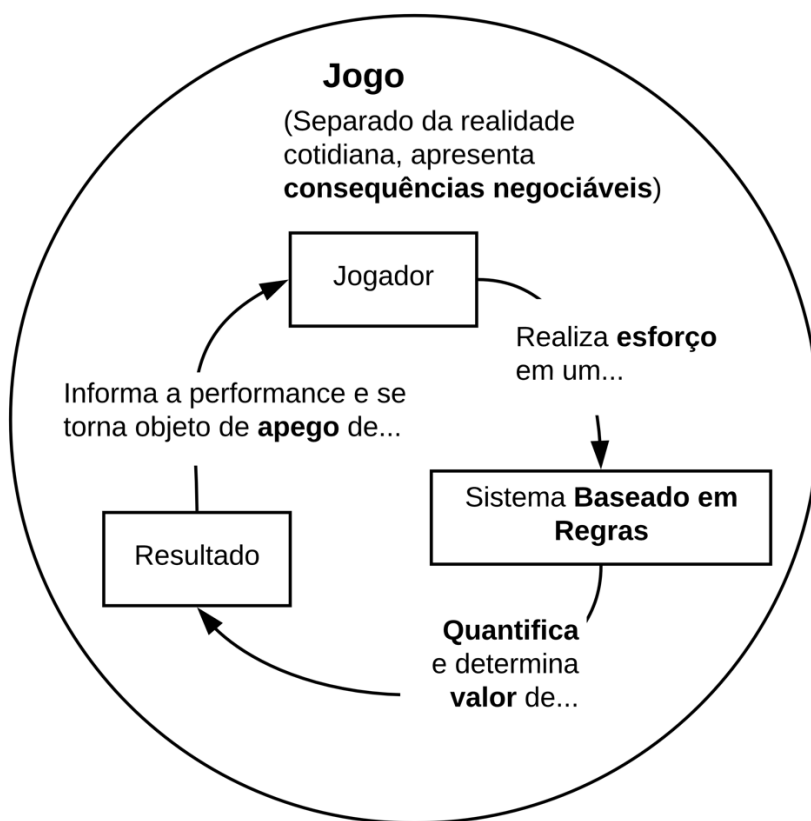


Figura 18 - Visão sistêmica das definições de Juul (Queiroz et al., submetido)

Assim, nossa argumentação será estruturada em torno das seis características que, de acordo com Juul, compõem esta definição: (1) *regras*; (2) *resultados variáveis e quantificáveis*; (3) *valorização do resultado*; (4) *esforço do jogador*; (5) *apego do jogador ao resultado*; (6) *consequências negociáveis*.

Entretanto, para maior fluidez do texto e adequação ao material pesquisado, estas características serão agrupadas e reordenadas da seguinte maneira: (1) regras e consequências negociáveis (apresentadas na seção *Regras e realidades*); (2) resultados variáveis, quantificáveis e valorizados (*Medidas e valores*); (3) o jogador: esforço e apego (*Esforço e recompensa*).

### 3.3

#### Regras e realidades

Regras determinam o funcionamento de um jogo, estabelecendo aquilo que é ou não permitido (Huizinga, 2004). São conjuntos de instruções claras e explícitas a respeito do funcionamento do jogo e das ações do jogador (Salen e Zimmerman, 2004). Quando aplicada ao trabalho em ciências e, particularmente, em software científico, a noção de *regra* pode significar pelo menos duas coisas: em primeiro lugar, *o método científico*, que prevê a elaboração de hipóteses, a realização de experimentos, o estabelecimento de metodologias que devem governar a pesquisa científica, garantindo sua testabilidade, reprodutibilidade e rigor, visando a observação e análise dos resultados. Em segundo lugar, regras podem se referir às *teorias* que são formuladas e, eventualmente, implementadas através de modelos matemáticos para simular os fenômenos estudados. De modo geral, as primeiras devem guiar as ações do cientista, enquanto os segundos devem governar o funcionamento da simulação. Em *The Structure of Scientific Revolutions*, Thomas S. Kuhn compara extensivamente a prática científica a jogos de quebra-cabeça. Para Kuhn, ambas as atividades são sustentadas por regras que estabelecem limites sobre “a natureza das soluções aceitáveis e as etapas através das quais são obtidas” (Kuhn, 2012 p.38, tradução do autor). A frase de Kuhn reflete esse duplo sistema de regras englobando as leis e paradigmas da ciência (‘soluções aceitáveis’) e os métodos da pesquisa científica (‘etapas através das quais são obtidas’). Em ambos os casos, deve-se reforçar, preza-se pela *objetividade* que – como lembraram Huizinga e Bogost – distingue a ciência e o software científico dos jogos e *video games*.

O *método científico* busca assegurar a integridade da ciência na construção do conhecimento. Para Hacking (1983), é característica do método científico a união do raciocínio (teórico) à prática (experimental) através da articulação e do cálculo. Podemos argumentar que regras metodológicas funcionam como formas de limitar e direcionar as ações do cientista – assim como ocorre com o jogador no

jogo. Tais regras são, de acordo com Karl Popper, “as regras do jogo da ciência empírica” (Popper, 2005, p.32, tradução do autor). Para ele, o método científico surge da necessidade de solução de problemas sobre os quais elaboram-se hipóteses a serem criticamente discutidas e testadas (Popper, 2010). A partir dos resultados observados, hipóteses são eliminadas ou consideradas satisfatórias – mas nunca confirmadas em definitivo (Popper, 2010). Ainda de acordo com Popper (2005), teorias devem ser testadas através da *dedução*, através da verificação de suas consistências lógicas internas; da comparação com outras teorias, e de testes empíricos para avaliação de aplicações e experimentações no mundo real. É através deste caminho, afirma, que teorias podem ser corroboradas ou refutadas. Este, aliás, é o ponto central de *A Lógica da Pesquisa Científica*, no qual Popper observa que teorias científicas devem ser *falseáveis*, ou seja, abertas a teste e refutação. Desta forma, “o jogo da ciência, em princípio, não tem fim” (Popper, 2005, p.32, tradução do autor). Testes devem ser, portanto, conduzidos através de experiências rigorosas, reproduzíveis e falseáveis, que corroborem os enunciados científicos propostos de forma lógica e sólida. Assim, “aqueles que não estiverem dispostos a expor suas ideias ao risco da refutação não participam do jogo científico” (Popper, 2005, p.280, tradução do autor). De fato, observa-se a extrema importância, no trabalho científico, da elaboração, detalhamento e descrição dos procedimentos metodológicos referentes aos experimentos – particularmente quando direcionados à publicação. Trata-se de apresentar formas para reprodução de experimentos, colocando à prova o valor científico do conhecimento gerado.

Existem ainda outras maneiras de se registrar as regras e procedimentos em diferentes contextos: em *Epistemic Cultures – How the Sciences Make Knowledge*, a socióloga Karin Knorr Cetina enfatiza a importância da narrativa e visualidade de guias visuais e desenhos esquemáticos que descrevem procedimentos passo-a-passo (frequentemente organizados, como alguns jogos, em etapas e sub-etapas) de trabalho experimental em biologia molecular, muitas vezes ilustrando elementos e materiais invisíveis ao olho. Knorr Cetina Também destaca a importância de histórias contadas sobre os procedimentos anteriores – bem-sucedidos ou não – que mantêm o conhecimento gerado circulando sem reduzi-lo a “regras ou instruções abstratas” (Knorr Cetina, 2009, p.106, tradução do autor).



A deferência ao rigor do método científico é contestada pelo filósofo Paul Feyerabend, que propõe, em *Contra o Método*, que é justamente ao ir contra práticas metodológicas rigorosas que a pesquisa científica avança. Enquanto Kuhn fala de quebra-cabeças, Feyerabend compara a construção do conhecimento à manipulação e funcionamento de blocos de montar, propondo uma atitude – em termos que discutimos anteriormente – *playful*, e não *gameful*. Para ele, é através da brincadeira, e não da identificação de problemas, que soluções, ideias e práticas são desenvolvidas. Brincar, portanto, seria um caminho natural para o conhecimento. De acordo com ele,

“Primeiro, temos uma ideia ou um problema, e depois agimos (...) essa não é a forma como crianças se desenvolvem. Elas usam as palavras, as combinam, brincam com elas até compreender o significado (...) a brincadeira inicial é parte essencial do ato final de compreensão. Não há porque esse mecanismo deixe de funcionar no adulto” (Feyerabend, 1993, p.17, tradução do autor).

Feyerabend propõe, também, uma abordagem interacionista onde o cientista é simultaneamente observador e participante. Sua visão do trabalho científico se afasta da visão tradicional. Em *Objectivity*, Lorraine Daston e Peter Galison discutem o estabelecimento da objetividade, a partir de meados do século 19, como valor epistêmico fundamental da ciência. Em prol da objetividade, cientistas passaram a buscar observar fenômenos de forma neutra, esforçando-se para domar a própria subjetividade na interpretação e construção de fatos científicos (Daston & Galison, 2010). É a essa abordagem que Feyerabend parece se opor. Como defende em *A Ciência em Uma Sociedade Livre*, “[um] cientista, (...) não é como uma criança que precisa da metodologia do pai e da racionalidade da mãe para lhe dar segurança e direção; ele pode cuidar de si mesmo” (2011, p.41). Para Feyerabend, é ingenuidade acreditar que problemas de pesquisa sejam resolvidos a partir da aplicação de regras simples. Cientistas, como afirma em *Adeus à Razão*, “sempre adaptaram seus procedimentos à tarefa a ser realizada. Não há um único “método científico”, mas sim uma grande quantidade de oportunismo” (Feyerabend, 2010, p.47). Assim, ao invés de tentarem se adequar a um conjunto de regras pré-definidas, cientistas as definem enquanto desenvolvem seus trabalhos (Feyerabend, 2011). Trata-se de uma afirmação compatível com a de Galison: “apenas artificialmente experimentos podem ser reduzidos a um protocolo” (1987 p.241, tradução do autor). Ou seja: em alguns casos, nem mesmo a reprodução bem-sucedida de um experimento pode ser garantida por mera obediência a regras. A

ideia de reprodutibilidade parece enganosa para Hacking, que afirma que experimentos “não são repetidos, mas melhorados até que o fenômeno possa ser obtido com regularidade” (1983, p.xiii, tradução do autor).

Por sua vez, *Leis e teorias* científicas se caracterizam por regras através das quais cientistas descrevem o que acreditam ser as leis da natureza. A este respeito, Richard Feynman afirmou:

“Não conhecemos as regras do jogo; podemos apenas *assistir* ao jogo em andamento. É claro que, se o assistirmos por bastante tempo, em algum momento poderemos entender algumas das regras. *As regras do jogo* é o que queremos dizer com os *fundamentos da física*” (Feynman et al., 1966, p2-1, tradução do autor).

O progresso da ciência, no entanto, não se dá pelo contínuo acúmulo de teorias, mas pela substituição de teorias por outras, mais abrangentes e apropriadas para descrever os fenômenos que observam. Em *O Valor da Ciência* (1995), o filósofo e matemático Henri Poincaré destaca a gradual substituição de antigas leis por outras novas, mais próximas da verdade. A evolução das ciências, para ele, se dá através da transformação gradual dos antigos saberes em novos – um processo que compara à evolução das espécies – onde fatos e teorias mais próximos da verdade substituem seus antecessores, preservando traços e atualizando seus conhecimentos. Este pensamento também se mostra presente em *Ciência e Método* (Poincaré, 1921), onde Poincaré afirma que, estando uma lei científica firmemente estabelecida, torna-se mais importante procurar exceções à regra e observar onde ela falha. Estar atento e aberto à possibilidade de mudanças nas regras é, portanto, crucial no avanço da ciência. Este pensamento é semelhante ao de Kuhn (2012), que afirma que mudanças de paradigmas nas ciências têm suas origens em anomalias que não podem ser explicadas por teorias predominantes de suas épocas – exigindo novas teorias e práticas que possam explicá-las. São essas revoluções que, para Kuhn, fazem da ciência um jogo que vale a pena ser jogado (Kuhn, 2000). Desta forma, antigos conjuntos de regras são trocados por outros, que se tornam o paradigma vigente de seus domínios científicos até que sejam depostos por seus sucessores. No entanto, é importante reforçar que novas teorias emergem dos problemas encontrados por suas antecessoras. Kuhn ilustra esse ponto em outra comparação à evolução biológica: “o desenvolvimento científico deve ser visto como um processo movido pelo que já passou, e não puxado pelo que virá – é uma evolução a partir de um ponto, e não em direção a outro” (2000, p.96, tradução do

autor). A respeito das substituições de teorias por outras, Galison (1987) faz ressalvas à ideia de um rompimento radical com antigas práticas e teorias, afirmando que, em alguns casos, cientistas experimentais são capazes de abandonar aspectos mais abstratos, mas aderir a outros, mais objetivos e concretos.

Popper descreveu teorias científicas como “sistemas de signos ou símbolos [utilizados] para racionalizar, explicar e dominar [o mundo]” (2005, p.37-38, tradução do autor). Deve-se enfatizar, nesse caso, a importância do registro e codificação das teorias. Para Popper, é a linguagem que torna possível tornarmos a ciência objetiva, materializando teorias para que sejam examinadas (2010). De acordo com o filósofo Gaston Bachelard (1978b), a matemática é a linguagem que melhor se presta a esse papel, por sua inventividade e estabilidade. Portanto, é interessante perceber que, tal como as regras de jogos eletrônicos, leis científicas podem ser codificadas e implementadas em objetos. Em *O Novo Espírito Científico*, Bachelard descreve instrumentos científicos como “teorias materializadas” (Bachelard, 1978), propondo que os fenômenos científicos podem ocorrer não na natureza observável, mas sim naqueles instrumentos (a *fenomenotécnica*). Esta ideia pode ser transportada tanto para o software científico – que codifica regras em modelos matemáticos e simulações – quanto para jogos, particularmente os digitais, que registram e automatizam suas regras de funcionamento em seus códigos e interfaces. De forma semelhante, aparatos científicos “fornecem as regras do jogo aos cientistas” (Kuhn, 2012, p.41, tradução do autor) – uma relação entre instrumentos, práticas e teorias que, novamente, é relativizada por Galison (1987), para quem a associação é flexível, permitindo que técnicas e instrumentos sejam aplicados a diferentes teorias. Outro conceito importante relacionado à codificação de regras e leis científicas são os *modelos científicos* – representações da estrutura e funcionamento de fenômenos pesquisados – na constituição de teorias e sua tradução em experimentos. Em *Creating Scientific Concepts*, Nancy J. Nersessian discute a construção de modelos como caminho para a solução de problemas e o desenvolvimento de novos conceitos científicos. Para ela, modelos são capazes de propor as regras de funcionamento dos sistemas pesquisados ou, também, representar demonstrações do sistema proposto. Em alguns casos – particularmente no uso de modelos matemáticos – ambas as propriedades (proposicional e demonstrativa) podem estar presentes. Modelos científicos podem assumir diversas

formas: diagramas, modelos tridimensionais (físicos ou não), fórmulas matemáticas, etc. (Nersessian, 2008). Peter Galison (1987) destacou a importância do uso de modelos como elos essenciais entre a teoria e os experimentos em física de altas energias, e James D. Watson (1980) enfatizou o papel de modelos (físicos, no caso) na descoberta da estrutura do DNA, realizada por ele e Francis Crick. De acordo com Nersessian, a construção de modelos – e processos de simulação de novos estados – requer “conhecimento tácito e explícito sobre o domínio das entidades, comportamentos e processos, especialmente conhecimento sobre as suas causas” (Nersessian, 2008, p.105, tradução do autor). No caso da ciência computacional, modelos podem ser criados e manipulados em diferentes estágios da pesquisa, seja na codificação das teorias e leis científicas em algoritmos e modelos matemáticos ou na representação visual – e possivelmente a manipulação interativa – dos fenômenos pesquisados. Modelos, observou Evelyn Fox Keller, tornaram-se demasiadamente intrincados e complexos: “Embora [o modelo computacional] seja tradicional em seu uso de equações diferenciais para representar o sistema, ele passa longe da simplicidade e elegância as quais buscavam modelos das ciências físicas no passado. De fato, tal modelo seria inimaginável sem o poder inimaginável de devorar números dos computadores modernos. É pesado, bagunçado e, isoladamente, opaco a qualquer tipo de intuição (matemática ou não).” (Keller, 2003, 251, tradução do autor).

Há, portanto, uma relação entre diferentes conjuntos de regras: de um lado, o método científico, de outro, leis e teorias. Tal relação torna a ciência um jogo bastante particular e intrincado. Neste jogo, cientistas devem identificar e estabelecer as regras mais apropriadas para seus experimentos e, a partir dos resultados, contribuir para o aperfeiçoamento de leis e teorias. Do ponto de vista experimental, a necessidade dos cientistas de redefinir regras – sejam essas teóricas ou metodológicas – pode tornar desejável o desenvolvimento de experimentos que possam ser reconfigurados com relativa facilidade e custo, enquanto a necessidade de reproduzi-los pode exigir formas de torná-los portáteis (Galison, 1987).

A codificação das regras – e da ciência – é um aspecto da segunda característica dos jogos em discussão: *consequências negociáveis*. Para Juul, exceto em casos limítrofes como jogos de azar jogados por dinheiro, os resultados do jogo não oferecem consequências à “vida real” de seus jogadores. Jogos costumam ser

considerados atividades livres, não-produtivas, separadas da vida real (Caillois, 2001; Huizinga, 2004). Huizinga ilustrou essa ideia através do *círculo mágico* onde a brincadeira ocorre isolada e protegida do mundo real. Embora possa-se argumentar que este círculo não é impermeável, e que jogos e mundo real estabeleçam trocas e diálogos (Salen e Zimmerman, 2004), é seguro afirmar que jogos obedecem às suas próprias regras. Embora os resultados da prática científica tenham impacto na vida do cientista (sobretudo em aspectos profissionais), podemos argumentar que a ciência é muitas vezes descrita em termos que sugerem sua separação do mundo real.

Em *Laboratory Life: The Construction of Scientific Facts*, o filósofo Bruno Latour e o sociólogo Steve Woolgar apontam que práticas em laboratório são mediadas por instrumentos e equipamentos. Os conceitos de fenomenotécnica e teorias materializadas de Bachelard são usados para comunicar a “realidade artificial” onde, de acordo com eles, os experimentos ocorrem. Para eles, fatos científicos não são descobertos, mas cuidadosamente e artisticamente construídos: “‘realidade’ não pode ser usado para explicar por que enunciados [científicos] se tornam fatos, pois é só após tornar-se fato que o fenômeno é realmente obtido” (Latour & Woolgar, 1986, p.180, tradução do autor). Trata-se de uma opinião semelhante à de Karin Knorr Cetina, cujo livro *The Manufacture of Knowledge: An Essay on the Constructivist and Contextual Nature of Science* (1981) analisa a forma como fatos são fabricados no laboratório, em uma realidade previamente construída com matérias-primas devidamente selecionadas, distinta da natureza que a ciência se dispõe a explicar. Mais ainda, o trabalho científico realizado é bem-sucedido por atender às necessidades contextuais do ambiente especial onde é realizado. Bachelard, que exaltava a matematização e pensamento como produtores de novos fenômenos, fazia afirmações semelhantes quatro décadas antes: “É então que se percebe que a ciência constrói seus objetos, que nunca ela os encontra prontos” (1996, p.77).

Na filosofia da ciência, o limite entre ciência e realidade é o foco do debate entre realistas e antirrealistas. A natureza deste debate foi sintetizada da seguinte maneira por Hacking:

“O realismo sobre teorias diz que eles buscam a verdade, e às vezes se aproximam dela. Realismo sobre entidades dizem que os objetos mencionados em teorias devem

realmente existir. O antirrealismo sobre teorias diz que não se deve acreditar literalmente em nossas teorias que são, no melhor dos casos, úteis, aplicáveis, e boas para predições. O antirrealismo sobre entidades diz que entidades postuladas por teorias são, no melhor dos casos, ficções intelectuais úteis” (Hacking, 1983, p.x, tradução do autor).

Popper, um realista, pressupunha a existência de uma realidade a ser descrita por teorias científicas que – apesar de insuficientes – aproximam-se da verdade (Popper, 2010). Instrumentalistas, no entanto, discordam. Popper exemplifica a abordagem instrumentalista a partir do posicionamento de Moritz Schlick, o qual teria dito que “teorias não são propriamente afirmações, mas ‘uma regra ou conjunto de instruções para a derivação de afirmações a partir de outras afirmações’” (apud Popper, 2002, p.145, tradução do autor). Assim, mais do que descrever a realidade, leis científicas seriam ferramentas matemáticas para a obtenção de resultados. Nota-se, portanto, uma relação direta entre sistemas de regras e a separação com a realidade, através dos quais jogos se definem. Vale lembrar, como faz Popper, da possibilidade de aplicação e uso de teorias mesmo após estas serem refutadas (caso da física newtoniana, por exemplo): “na busca por instrumentos poderosos, estamos frequentemente bem servidos por teorias que sabemos serem falsas” (Popper, 2001, p.306, tradução do autor).

Poincaré, no entanto, se opunha à classificação de teorias como meros conjuntos de regras, afirmando:

“As regras do jogo são convenções arbitrárias, e poderíamos ter adotado a convenção contrária, que não teria sido menos boa. Ao contrário, a ciência é uma regra de ação que funciona, pelo menos de maneira geral, e digo mais – ao passo que a regra contrária não teria funcionado (...) Se essas “receitas” científicas têm um valor, como regra de ação, é porque sabemos que elas funcionam” (Poincaré, 1995, p140).

Poincaré opõe-se à visão de que as leis e fatos científicos sejam artificialmente construídos pelo cientista: crê em uma realidade objetiva, constituída por fatos concretos que podem ser traduzidos em fatos científicos e termos matemáticos. Esta tradução, no entanto, seria incapaz de levar em consideração todos os detalhes do universo exterior que busca representar (no que se aproxima das simulações). Assim, a ciência lança mão de regras elaboradas e escolhidas não por serem, necessariamente, mais verdadeiras, mas também por critérios de eficiência e simplicidade. Poincaré admite que o cientista cria, de fato, a linguagem na qual elabora as leis científicas. Admite também a limitação de tais leis, aproximadas e imperfeitas, na representação de um universo impossível de ser

reproduzido em sua completude e complexidade. Considera ainda que o cientista tem papel ativo nessa aproximação ao excluir, por arbitrariedade ou perspicácia, variáveis que julga descartáveis nas leis que enuncia. Porém, considera que teorias não são “meras receitas práticas” (1921b, p.140, tradução do autor), uma vez que expressam relações verdadeiras.

Uma visão antirrealista é defendida pela filósofa Nancy Cartwright em *How the Laws of Physics Lie* (1983), obra na qual argumenta que leis fundamentais da física só são verdadeiras somente quando aplicadas aos modelos associados às teorias, mas não ao mundo real. Para ela, tais leis funcionam como simulacros que apenas se assemelham à realidade, tentando adaptar os fenômenos da natureza a modelos que sejam adequados à teoria. A título de exemplo, Cartwright observa que a lei da gravidade – apesar de seu gigantesco reconhecimento – falha em descrever o verdadeiro comportamento de corpos físicos em condições que não sejam ideais. Assim, as leis da física não regeriam a natureza real, mas sim seu simulacro. Modelos científicos seriam, portanto, trabalhos de ficção onde propriedades reais dos objetos se misturam a outras, adotadas por conveniência.

Hacking credits sua conversão ao realismo ao testemunho de uma experiência onde um cientista borrifava pósitrons sobre uma amostra de nióbio. Para o filósofo, “se você pode borrifá-los, eles são reais” (1983, p.23, tradução do autor). Bachelard, muito antes de Hacking, teceu observação semelhante:

“Mesmo numa ciência muito avançada, as condutas realistas subsistem. (...) Essas condutas realistas reinstalam-se porque o teórico racionalista tem necessidade de ser compreendido por simples experimentadores (...) porque na sua vida comum ele é efetivamente realista.” (Bachelard, 1978b, p.15)

A aplicação do conceito de consequências negociáveis à ciência não é necessariamente dependente de uma visão antirrealista, e pode ser interpretado de diferentes maneiras. Ele é compatível, por exemplo, com o paralelo entre os pensamentos pré-científico e científico estabelecido por Popper, para quem ambas as modalidades se fundamentam no modelo: (1) problema, (2) tentativa de solução, (3) eliminação. No entanto, Popper argumenta, durante o desenvolvimento pré-científico, as consequências de tentativas fracassadas eram sofridas por aqueles que as realizavam. Por sua vez, o método crítico do pensamento científico – de acordo com o qual problemas são sujeitos a elaboração e experimentação – evita que

aqueles que tentam solucionar o problema sejam, efetivamente, vítimas passivas da eliminação de soluções propostas via tentativa e erro, tornando-se agentes do processo de eliminação. Dessa forma, “o cientista pode destruir sua hipótese sem perecer junto a ela” (Popper, 2010, p.10, tradução do autor). De fato, embora a objetividade seja mais frequentemente associada à não-contaminação da pesquisa pelo sujeito que a investiga, o cientista é, também, protegido por essa separação. Segundo Bachelard em *A Formação do Espírito Científico*, o fracasso de uma experiência alquímica – precursora da química – levantaria suspeitas quanto à pureza moral de seu realizador. “Não conseguir produzir o fenômeno inspirado com o apoio dos símbolos corretos não é um simples fracasso, é um déficit psicológico e uma falta moral.” (1996, p.64). Mais adiante, complementa: “Parece que, hoje, o pesquisador de laboratório consegue se afastar com mais facilidade da sua função. Ele não confunde a vida sentimental com a vida científica” (1996, p.65).

Abordamos, no capítulo dois, como o software científico permite aos usuários gerar experimentos que, por impraticabilidade ou riscos, não poderiam ser realizados no mundo real. No entanto, observamos que esta noção de separação com o real precede o surgimento da ciência computacional, podendo ser relacionada à matematização da ciência – um fenômeno que, de acordo com Kuhn (1977), se expandiu rapidamente no início do século XIX. Em *Flow: The Psychology of Optimal Experience*, Csikszentmihalyi afirma que, enquanto sistema simbólico, a matemática “é como um jogo, provendo uma realidade separada, um mundo próprio onde se pode realizar ações permitidas naquele mundo” (2008, p.118, tradução do autor). Décadas antes do advento da computação, Bachelard teceu elaborações a respeito da matematização da ciência, afirma que, daquele momento em diante, “o estudo do fenômeno depende duma atividade puramente numenal; é a matemática que abre os novos caminhos para a experiência ” (Bachelard, 1978, p.119). Esta matematização apoia o que Bachelard identifica como a ciência se distanciando da observação imediata dos fenômenos e objetos do mundo real, e se aproximando mais intensamente do pensamento racional e do plano das ideias. Neste sentido, Poincaré já havia observado, em *Ciência e Hipótese*, a deficiência da coleta de dados puramente através de métodos sensoriais (1921b). A observação científica, observa Bachelard, “reconstrói o real depois de ter reconstruído seus esquemas” (1978, p.96). Este ponto é particularmente relevante para a presente discussão, pois sugere que o pensamento científico tenta reconstruir a realidade através de “um



mundo à imagem da razão” (Bachelard, 1978, p.96). Nesse caso, a realidade se torna um caso particular dentro de todas as possibilidades. Esse novo espírito científico, capaz de romper com a realidade imediata para avançar no campo da representação, seria capaz de avançar teorias de formas antes impensáveis, como afirma Bachelard em *A Filosofia do Não*: “Para o cientista do século XIX, o conceito de uma massa negativa teria sido um conceito monstruoso. Teria sido, para a teoria que o produziu, um erro fundamental” (1978b, p.21). Mais ainda, ao retirar-se a ênfase da observação, substituindo-a pelo racional e pelo matemático, facilitou-se o estabelecimento de regras e teorias – ponto ilustrado, por Bachelard, com a tabela periódica de Mendeleiev, através da qual as propriedades de elementos ainda não descobertos poderiam ser previstas (1978b). Neste caso, a observação confirma a descoberta, mas não é seu ponto de partida. Alguns domínios científicos, no entanto, oferecem resistência à matematização. Em *Making Sense of Life: Explaining Biological Development with Models, Metaphors, and Machines*, Evelyn Fox Keller discute, entre outros assuntos, como a biologia demorou, em comparação a outras disciplinas, a ganhar um campo de especialização baseado na matemática. Mais ainda, Keller aponta para uma resistência histórica em se considerar este campo um ramo da biologia propriamente dita, uma vez que seus modelos são “claramente considerados fictícios, não se pretendendo realistas” (Keller, 2003, p.12, tradução do autor), sendo a matemática melhor aceita no estudo de populações do que em indivíduos. Na visão de Keller, foi graças à genética experimental, e não à matematização, que se compreendeu melhor o desenvolvimento biológico. De fato, indica Keller, enquanto o termo “modelos” são amplamente utilizados em vários domínios para descrever sistemas artificialmente criados, em biologia ele costuma ser empregado para descrever organismos modelos, apanhados na natureza, como moscas e camundongos normalmente utilizados em experimentos. No entanto, Keller observa, a computação serviu como metáfora na construção de teorias: o conceito de “programa genético”, por exemplo, se vale de sua semelhança com programas de computador para maior poder explicativo e força narrativa – ainda que a ideia de “programação” não deva ser levada ao pé da letra. Mesmo o uso de novas tecnologias computacionais – como a possibilidade de se assistir a processos celulares em tempo real – pode reforçar a conexão das ciências biológicas com a ‘realidade’: “Ironicamente, a eficácia destas tecnologias contribuiu para o reavivamento de uma tradição duradoura entre

estudantes de biologia – a tradição na qual ver (e, ainda mais, assistir – ou ver acontecer) é considerada a fonte de conhecimento mais confiável e a base indispensável da compreensão” (Keller, 2003, p.201, tradução do autor). Por outro lado, Keller reconhece que, tendo a simulação computacional se popularizado nas ciências biológicas durante a década de 1990, o computador tornou-se indispensável à pesquisa nos dias atuais – quando modelos computacionais fortemente baseados em dados experimentais se tornam cada vez mais indispensáveis e presentes na representação, análise e interpretação de dados por cientistas – mesmo aqueles de pouca familiaridade com matemática. A aplicação de modelos computacionais na biologia, afirma Keller, exemplificam “práticas teóricas onde nem o pensamento nem teoria são puros, e onde a mistura de trabalho teórico e prático, conceitual e material, é visível e inescapável” (2003, p.238, tradução do autor). Por outro lado, a computação possibilitou, também, o estudo de mundos sintéticos formas artificiais de vida, como os autômatos celulares.

Em *Ciência e Hipótese*, Poincaré reforça o distanciamento entre o pensamento matemático e o mundo real ao afirmar que “Matemáticos não estudam objetos, mas relações entre objetos (...) e, para eles, a matéria não é importante, apenas a forma os interessa” (1921b, p.44, tradução do autor). A ideia de se construir um mundo racional e abstrato, através do qual compreender e refletir o mundo real, se aplica também às simulações presentes na ciência computacional – e a separação entre este mundo racional e a realidade imediata remete ao círculo mágico dos jogos. Para Feyerabend, a matemática foi a primeira disciplina a promover a ideia de um conhecimento “verdadeiro, puro e objetivo” (2010, p.90), e carrega o maior grau de liberdade dentre as disciplinas.

Poincaré defende que se cultive a matemática por si mesma, independente de aplicações ou motivos utilitaristas, de maneira próxima à arte. A abstração característica da análise matemática faz com que a ciência se afaste do mundo real, e Poincaré expressa este distanciamento quando afirma: “Para compreender a natureza é preciso sair de si mesmo, por assim dizer, e contemplá-la de vários pontos de vista diferentes” (Poincaré, 1995, p.105). Este afastamento do real se assemelha àquele que ocorre no jogo. De fato, Poincaré aproxima a ciência ao jogo, atribuindo à primeira o caráter desinteressado que Huizinga atribui ao segundo:

“[Admiro] as conquistas da indústria (...) um dia elas darão a todos *o lazer de contemplar a natureza*. Não digo que a ciência é útil porque nos ensina a construir máquinas; digo que as máquinas são úteis porque, *ao trabalhar para nós, um dia nos deixarão mais tempo livre para fazer ciência* (...) *tendo o homem perseguido um objetivo desinteressado*, todo o resto lhe veio por acréscimo” (Poincaré, 1995, p.106, grifo do autor).

Retornamos, assim, ao conceito de círculo mágico, originalmente proposto por Huizinga. Em *Rules of Play: Game Design Fundamentals*, Katie Salen e Eric Zimmerman descrevem-no como “um lugar especial no tempo e espaço criado pelo jogo (...) fechado e separado do mundo real” (2004, p.95, tradução do autor). Tal definição pode ser transportada para outro espaço, dentro da prática científica: o *laboratório*. Em *Creativity: Flow and the Psychology of Discovery and Invention*, o psicólogo Mihaly Csikszentmihalyi investiga o fenômeno da criatividade em vários domínios profissionais – inclusive na ciência. Csikszentmihalyi descreve laboratórios como lugares que podem ser projetados para favorecer foco e o trabalho criativo, servindo como um local seguro para se trabalhar ininterruptamente, “sem as distrações da “vida real”” (2013, p.145, tradução do autor). Mais recentemente, esta relação entre jogo e ciência foi explicitada pelo neurocientista Stefan Glasauer, para quem a necessidade de ambientes controlados e a repetição de experimentos científicos os tornam mais próximos do jogo do que da vida real (Glasauer e Jahrmann, 2016). Kuhn observa que um aspecto da transformação da ciência no século XXVII foi, justamente, a criação de condições experimentais que não seriam possibilitadas pela natureza, mas resultado da intervenção de cientistas (Kuhn, 1977). Kuhn afirma, também, que a autossuficiência do meio científico para julgar e apreciar o conhecimento gerado contribui para a separação da prática científica da sua vida cotidiana (Kuhn, 1977).

Uma extensa caracterização do laboratório como espaço separado da vida real é realizada por Knorr Cetina, que o descreve como local para a exploração de realidades alternativas: “o laboratório é um espaço virtual (...) como um palco onde peças são realizadas de tempos em tempos” (2009, p.34, tradução do autor). Neste ambiente, fenômenos naturais podem ser estudados desconectados de quando, onde e como ocorrem no mundo real e na natureza. Knorr Cetina apresenta uma distinção entre três tipos de objetos de estudo em pesquisa laboratorial: (1) versões encenadas de fenômenos do mundo real; (2) versões parciais e processadas de fenômenos do mundo real; e (3) signos produzidos por fenômenos do mundo real. O primeiro tipo

é comparado a jogos de guerra utilizados no século XVIII para o planejamento de batalhas – modelo do qual, de acordo com a socióloga, a simulação de computador é o equivalente moderno. Neste primeiro caso – obviamente relevante ao presente estudo – a similaridade e fidelidade entre o modelo de laboratório e a realidade são de fundamental importância. O segundo tipo é comparado ao método de construção de catedrais na era medieval, onde construtores pesquisavam elementos de arquitetura de catedrais previamente construídas que pudessem ser implementados em construções futuras. Embora de relação menos óbvia com o presente trabalho, há uma noção da construção do conhecimento e de soluções a partir de blocos de armar que, assim como aquela proposta por Feyerabend, é compatível com aspectos *playful* dos jogos e brinquedos de montar. Laboratórios nessa categoria são comparados, também, a “oficinas e viveiros com atividades e objetivos específicos (...) para manipulação experimental” (2009, p.37, tradução do autor). Finalmente, o terceiro tipo de laboratório é comparado à psicanálise, onde se deve interpretar e reconstruir o significado de eventos a partir dos signos fornecidos. Knorr Cetina usa como exemplo deste tipo de laboratório a interpretação de dados digitais capturados por colisores de partículas no estudo de física de altas energias. Novamente, a conexão com a pesquisa aqui realizada é menos óbvia, mas argumentamos que tal atividade – encontrar significados ocultos e não-triviais em signos – é uma atividade semelhante aos jogos de raciocínio lógico. De fato, Knorr Cetina relata um algoritmo usado em experimentos semelhantes em física de altas energias ao qual se deu o nome de “jogos de adivinhação” (no inglês, “*guesswork games*”). Outro aspecto que pode ser identificado como separação do real são as transformações, como sugere Knorr Cetina, de máquinas em organismos (em física de altas energias) e de organismos em máquinas (no caso de biologia molecular). No primeiro caso, a justaposição de um universo simbólico sobre o técnico pode ser percebida na descrição, por cientistas, de suas máquinas como entidades vivas que apresentam personalidades. De forma análoga à atribuição de forças secretas ou doenças em objetos materiais durante o pensamento pré-científico (Bachelard, 1996), equipamentos são descritos por cientistas como sendo seres fisiológicos (que envelhecem, morrem, ficam doente, etc.) capazes de comportamentos idiossincráticos (como cooperar, concordar, confundir-se, etc.). Em alguns casos, equipamentos podem cooperar e competir entre si. Como organismos “quase biológicos”, máquinas são descritas por um cientista como “uma ferramenta, um

brinquedo e um amigo" (Knorr Cetina, 2009, p.122, tradução do autor), com as quais físicos – que muitas vezes as constroem e mantêm – desenvolvem uma forte conexão. Em alguns casos – e assim como nos jogos - a máquina pode personificar um personagem antagônico, um inimigo contra o qual se luta e que deve ser eliminado.

É interessante comparar a visão de Knorr Cetina sobre o trabalho científico com a caracterização, elaborada por Juul, do jogo como sendo simultaneamente ficcional e real (Juul, 2005). Ficcional pelo mundo ou narrativas que pode apresentar (os personagens do universo de *Mario Bros.*, por exemplo, não existem de fato). Real pelas regras que, de fato, são verdadeiras no mundo real (inimigos sobre os quais o jogador faça Mario saltar são, de fato, eliminados do jogo). Knorr Cetina descreve relação semelhante entre realidade e ficção na ciência:

"A física processa e opera dentro dessa realidade construída, move-se dentro de um meio de simulações e" ficções "materiais de acordo com seus próprios propósitos. Entretanto, a partir desta operação em um meio irreal, podemos derivar os efeitos da verdade (através de uma epistemologia alternativa), tecnologias podem ser postas em prática, e universos podem ser "compreendidos". (Knorr Cetina, 2009 p.249, tradução do autor)

Há, portanto, apesar da objetividade, um aspecto da ciência ligado à ficção: "O ficcional tem algo a ver com o "como se" (...); a habilidade dos cientistas de criar e trabalhar dentro de mundos tecnologicamente imaginados (...) [mas], hoje, tem ainda mais a ver com o "e também" (Knorr Cetina, 2009, p.251, tradução do autor). Ela aponta que em alguns casos, como na biologia molecular, pode-se fazer uma distinção entre "realidade" e "ficção" ao comparar resultados do sistema ("ficcional") modelado ao sistema ("real") de referência. Tudo faz lembrar a seguinte frase de Feyerabend: "Precisamos de um mundo de sonho para descobrir as características do mundo real" (Feyerabend, 1993, p.22, tradução do autor).

No caso da biologia molecular, a desconexão com o real acontece de forma reversa: organismos são convertidos em máquinas - uma perspectiva que é compatível com a noção, central à biologia moderna, de que a vida é a expressão do material genético (Knorr Cetina, 2009). Ao invés de objetos naturais comumente encontrados na natureza, a biologia molecular recorre a micro-organismos e linhagens de animais que são especificamente projetados para o laboratório. Seus usos pela ciência apresentam menor resistência ética do que o uso de espécimes

retirado de seus *habitats* naturais. Além disso, tais organismos são "transformados em lugares de produção e em máquinas moleculares" (Knorr Cetina, 2009, p.145, tradução do autor), através dos quais organismos podem ser concebidos, modificados e experimentados como "blocos de construção". De certa forma, a ideia de que "matar" máquinas (sejam elas máquinas ou micro-organismos) não seja moralmente reprovável faz lembrar a clássica discussão sobre o impacto dos jogos violentos no mundo real.

Para além de laboratórios individuais, poderíamos destacar a importância daquilo que Latour (1987) chama de *Centros de Acumulação* e *Centros de Cálculo*, instituições que centralizam dados e informações a respeito dos domínios científicos – e que podem realizar processamento e recombinação desses dados. Embora o termo Centro de Acumulação sugira o armazenamento de informações digitais, ela serve, também, para descrever museus de história natural e mapas celestes combinados a partir de anotações manuais de diversos observadores. Por outro lado, estas centrais podem funcionar – como alguns dos laboratórios descritos por Knorr Cetina – como representações miniaturizadas do ambiente que se deseja estudar, permitindo a simulação antecipada de eventos.

A ciência computacional, descrita por Knorr Cetina como "um laboratório no laboratório" (2009, tradução do autor) amplifica e potencializa os conflitos entre ciência, suas regras e a realidade. Em *Simulation and its Discontents*, a cientista social e teórica de novas mídias Sherry Turkle baseia-se em dois de seus estudos etnográficos, realizados no Massachusetts Institute of Technology (MIT) durante as décadas de 1980 e 2000, para discutir o impacto das simulações por computador no estudo e prática de ciências, design e engenharia. Turkle apresenta a visão daqueles que encararam com ceticismo e desconfiança o estabelecimento de tecnologias computacionais em suas disciplinas – e também daqueles que identificaram suas virtudes e vantagens. O primeiro grupo, predominantemente composto de cientistas em estágios mais avançados da carreira, temiam a substituição de habilidades de cálculo mental e pensamento crítico por processos computacionais desprovidos de transparência, interação e gráficos sedutores. De fato, um posicionamento crítico em relação à coerência interna e à plasticidade sedutora das simulações podem ser virtudes profissionais. A visualização científica gerada por computação gráfica, apesar de revelar padrões ocultos e engajar

cientistas, pode tornar convincente dados desprovidos de significado ou importância atraentes. Sabendo que “a precisão do computador é equivocadamente tomada por perfeição” (Turkle, 2009 p.80, tradução do autor). Neste caso, há o risco do erro causado por imperfeições no modelo computacional ou na simulação não serem devidamente percebidos por seus usuários. Quanto à falta de transparência dos softwares, esta ocorre quando os códigos-fontes de seus modelos e simulações são inacessíveis para um usuário incapaz de compreender o significado dos dados e cálculos realizados que se dá por satisfeito em inserir e receber dados sem compreender o funcionamento interno do sistema. Perde-se, então, contato com as regras que governam a simulação e, conseqüentemente, aumenta-se a distância entre estas regras e o mundo real. Esta perda de transparência, caracterizada pela ausência de escrutínio é característica da *caixa preta*: sistemas técnicos complexos, sobre cujo funcionamento não se entra em detalhes, bastando-se conhecer os valores de entrada e os de saída (Latour, 1987).

Outra crítica detectada por Turkle é que, uma vez que a natureza é modelada e codificada, sua simulação se comporta de forma previsível – ao contrário de experimentos realizados na vida real, mais abertos a surpresas. Neste ponto, a simulação por computador se assemelha a seu predecessor – o experimento mental – através dos quais cientistas, baseados em dados e informações previamente conhecidos, buscam compreender a natureza através de processos imaginativos que simulam experimentos impraticáveis no mundo real (Kuhn, 1977).

Knorr Cetina observa que experimentos em ciência computacional – como no caso da física de altas energias – não estão imunes a eventuais erros e enganos. Tais experimentos, no entanto, facilitam o registro e recuperação de erros. A partir destes registros, cientistas podem voltar ao ponto desejado para entender o erro. Esta característica é, de certa forma, semelhante àquela encontrada em jogos, onde os jogadores podem se sentir seguros para errar.

Ao longo de seu ensaio, Turkle (2009) faz um número de referências aos *video games* – outro fenômeno impulsionado por tecnologias computacionais e baseado simulação que emergiu aproximadamente ao mesmo tempo da ciência computacional. Por um lado, a familiaridade com essa nova mídia tornara as gerações mais jovens confortáveis com simulações computacionais científicas, com a aceitação da realidade e tangibilidade dos objetos em tela e com as formas de

manipulá-los – e até mesmo uma forma enxergar melhor essa realidade. Por outro lado, elas podem ser mais propensas a confundir regras internas dos sistemas de computador com aquelas do mundo real – além de estarem mais inclinadas a aceitar a falta de transparência dos sistemas.

Simulações por computador – apesar das críticas e desconfianças – tornaram-se o principal ambiente de trabalho para muitos cientistas (Turkle, 2009) – ainda que possam ser combinadas a outras ferramentas e métodos de pesquisa (Galison, 1987). O papel do computador nas ciências é bem sintetizado por Keller: “uma máquina para o processamento de dados, resolver equações, modelar os fenômenos de interesse, refinar e reconstruir imagens e fornecer representações em quatro dimensões de descobertas experimentais e teóricas ao leitor” (Keller, 2003, pp.199-200, tradução do autor). Turkle (2009) relata uma série de vantagens e qualidades que a ciência computacional apresenta. A velocidade e precisão computacional, aliadas às capacidades de visualização e interação, encorajaram novas formas de experimentação e aprendizado. Lidar e aprender com erros tornou-se mais fácil; há mais tempo para coletar, experimentar e até brincar com os próprios dados; resultados e padrões podem ser visualizados mais rapidamente; anomalias e irregularidades podem ser melhor compreendidas; há um aumento na percepção da intimidade entre a teoria e a ciência sendo realizada. Desta forma, pode-se argumentar que a ciência computacional colabora de forma quantitativa e qualitativa na obtenção de resultados e realização dos objetivos do trabalho científico.

É arriscado, no entanto, afirmar uma separação total entre ciência e realidade – bastando observar influência e impacto que exercem mutuamente uma sobre a outra. Mesmo a construção da matemática – presumivelmente a mais abstrata das ciências aqui descritas – deve muito à observação do mundo real. Como afirmou Poincaré, “se não houvessem corpos sólidos na natureza, não haveria geometria” (1921b, p.73, tradução do autor). Para Poincaré, ainda que isso não faça da geometria uma ciência experimental, os seres humanos desenvolveram-na da forma mais vantajosa para sua própria condição sensorial, e suas regras – criadas e adotadas por geômetras – foram esculpidas a partir de suas percepções de uma realidade externa. O caminho inverso – da teoria a um mundo concreto – também pode ser percorrido: Popper discutiu a relação entre corpo e mente a partir da



interação entre três ‘mundos’: o mundo físico (ao qual chamou de Mundo 1), o mundo de processos mentais (Mundo 2) e o mundo de produtos da mente humana – incluindo teorias científicas (Mundo 3). De acordo com Popper, elementos do Mundo 3 seriam capazes de influenciar o Mundo 1, dando como exemplo a construção de um arranha-céu: “Um arranha-céu é um objeto físico, e pertence ao Mundo 1. Mas é construído de acordo com um plano, que é afetado por teorias e um grande número de problemas” (2010, p.26, tradução do autor). O desabamento de um arranha-céu, conclui Popper, pode ser consequência de problemas decorrentes de teorias do Mundo 3, ou de interpretações no Mundo 2.

Por outro lado, há o impacto das ciências na sociedade. Feyerabend defende que “*uma sociedade livre insiste na separação da Ciência e da sociedade*” (2011, p.41, grifo no original), sob o argumento de que – apesar do discurso da objetividade – a ciência, tal como aplicada por especialistas, pode interferir negativamente sobre questões fundamentais, as quais poderiam ser mais adequadamente abordadas pela sociedade como um todo de acordo com seus saberes e tradições. Para Kuhn, a influência da ciência no campo socioeconômico se dá através da tecnologia – ainda que esta tenha se desenvolvido sem o auxílio da ciência até o século XIX (1977).

A suposta autonomia das ciências e sua separação da vida cotidiana também pode ser contestada, uma vez que mesmo a solução de quebra-cabeças oferecidos pela ciência normal pode estar sujeita a questões políticas internas e externas ao ambiente científico (Kuhn, 2000). Em *Cogitamus – Seis cartas sobre as humanidades científicas*, Latour questiona esse isolamento, apresentando como exemplo a pílula anticoncepcional. Resultado de desenvolvimentos no campo da endocrinologia, seu surgimento se deveu, também, a mudanças de costumes na sociedade e a oportunidade de financiamento independente (Latour, 2016). É Latour também quem admite, em *Science in Action: How to Follow Scientists and Engineers Through Society* (1987), que o cientista faz do laboratório um mundo separado para si. No entanto, ressalva que o cientista profissional não trabalha em total isolamento, estando ligado – a partir de sua instituição ou laboratório – a outros atores, como os diretores e responsáveis pelo laboratório, que se empenham em buscar recursos, através de órgãos empresariais, industriais e governamentais, para a manutenção e financiamento ao laboratório e, conseqüentemente, realização e

continuidade da pesquisa. “Resumindo” - afirma Latour, “quando cientistas e engenheiros são bem-sucedidos em criar um vasto mundo interno, significa que outros estão se empenhando em direção a mais ou menos o mesmo objetivo; quando são malsucedidos, significa que cientistas e engenheiros estão a seguir sozinhos sua direção.” (1987, p.157-158, tradução do autor). Em alguns casos, a tensão entre o ambiente científico e outros setores da sociedade pode oferecer riscos e desafios. Em *Good Work – When Excellence and Ethics Meet*, Howard Gardner, Mihaly Csikszentmihalyi e William Damon discutem o crescimento do campo da genética, apontando para conflitos no que se refere a uma excessiva mercantilização da pesquisa científica (Gardner et al., 2002). Neste caso, a pressão de investidores pelo desenvolvimento de produtos e serviços como terapias genéticas e clonagem poderia ser danosa a ética e aos valores internos ao meio científico. Assim, a pesquisa não seria desinteressada, mas sujeita às exigências do mercado.

### 3.4 Medidas e valores

A definição de jogo proposta por Juul aponta, como característica dos jogos, resultados variáveis, quantificáveis e valorizados. A primeira característica prevê a possibilidade de resultados diferentes a cada partida; a segunda, que se possa identificar de forma clara os resultados que devem ser buscados (ou seja, o *objetivo* do jogo); finalmente, a terceira característica indica que certos resultados são mais desejáveis que outros. Ao longo desta subseção, discutiremos os resultados e objetivos gerais da ciência e, mais especificamente, dos cientistas.

Em *Reflections on Gender and Science*, Evelyn Fox Keller aponta como dois objetivos da ciência o conhecimento e poder sobre a natureza. Para Keller, o caráter masculino sob o qual a ciência foi forjada equiparou tais objetivos a, respectivamente, objetificação e submissão da natureza – uma relação de poder que poderia ser substituída por outra, estabelecida “em termos de interesse mutuo e bem-estar, e não em termos de conflito” (1985, p.114, tradução do autor). De todo o modo, os objetivos identificados por Keller refletem não somente aspectos teóricos da ciência – associados ao conhecimento – como suas aplicações na realidade – associadas ao poder. Essa dualidade é descrita por Hacking em termos de *representação* e *intervenção*, que abarcam respectivamente a *teoria* e a

*experimentação* que se caracterizam como objetivos da ciência: “Teorias tentam dizer como o mundo é. Experimentos e novas tecnologias transformam o mundo” (Hacking, 1983, p.31, tradução do autor). Dessa forma, os objetivos declarados da ciência – e dos cientistas – podem ser descritos em termos que se inclinam, em alguns casos, para o lado da teoria e, em outros, para o lado da experimentação. Poincaré definia a ciência como um sistema de relações, uma classificação por onde “conhecer as verdadeiras relações entre as coisas” (1995, p.167). Apesar de reconhecer a importância das aplicações da ciência na indústria, Poincaré apontava como objetivo maior da ciência a busca incessante por uma verdade inatingível. De forma semelhante, Popper (2004) apontou como motivações para a prática científica a busca por conhecimento e pela verdade – ainda que não se possa alcançá-la. Ainda assim, Popper considerava como tarefa do cientista a busca por leis que possibilitassem a previsão de eventos (2004), e descreveu o objetivo da ciência como “a criação de teorias satisfatórias” (2010, p.14-15, tradução do autor).

Por outro lado, os objetivos da prática científica podem ser apresentados de forma mais imediata quando descritos de um ponto de vista ligado à experimentação. Para Hacking (1983), não é necessário que uma conjectura teórica esteja sendo testada para que um experimento faça sentido, e que, de fato, assim como certos experimentos têm a teoria como ponto de partida, algumas teorias surgem a partir de experimentos onde não há uma teoria pré-estabelecida. Para ele, é apenas através do relato, em retrospecto, dos experimentos que se impõe uma perspectiva direcionada pela teoria. Em alguns casos, a preocupação com aspectos teóricos não se faz tão presente: em *How Experiments End*, Peter Galison afirma que

“a real preocupação dos experimentalistas não é com mudanças abrangentes na visão de mundo. No laboratório, cientistas querem encontrar métodos disponíveis para eliminar ou quantificar interferências, identificar onde o sinal está se perdendo e corrigir erros sistemáticos” (Galison, 1987, p. 245, tradução do autor).

Trata-se de uma observação semelhante à de Knorr Cetina (1981), para quem o trabalho em laboratório se descola da teoria e se concentra em especulações e experimentações baseadas na prática. Para ela, cada laboratório está carregado de *know-how*, práticas e idiosincrasias que influenciam a construção do conhecimento – situando esse conhecimento localmente, e o afastando de um ideal universal.

Não se deve, portanto, menosprezar a importância dos objetivos imediatos dos experimentos. Em primeiro lugar, como aponta Galison (1987) eles podem determinar o sucesso de empreendimentos de altíssimo custo financeiro, envolvendo dezenas de profissionais – além da reputação profissional dos envolvidos. Além disso, apesar da separação dos ramos teórico e experimental de algumas ciências, teoria e experimento permanecem dialogando e informando um ao outro. Neste caso, ainda de acordo com Galison, objetivo do experimento é transformar-se na *demonstração* de uma teoria – com três possíveis resultados: publicação; correção e recomeço; e desistência. Enquanto o primeiro e o último destes resultados pressupõem o fim do experimento, o segundo abre espaço para que experimentos – e seus objetivos – sejam renegociados, modificados e aperfeiçoados. Oportunidades – ou a necessidade – para a redefinição de objetivos aumentam na medida em que o intervalo de tempo entre proposta de pesquisa e publicação de resultados se torna mais alongado – embora deva-se lembrar que experimentos precisam ser elaborados em torno dos fenômenos que buscam examinar, e dos resultados que pretendem alcançar. Ainda de acordo com as observações de Galison (1987), o papel e a liberdade dos cientistas em definirem seus próprios objetivos e teorias pode levar a erros de avaliação – como confiar em resultados preliminares positivos e diminuir, ou encerrar, a busca por problemas.

Kuhn enfatiza ainda a importância dada à *descoberta científica* enquanto objetivo do cientista – não apenas pelo prestígio profissional atrelado, mas, também, por sua utilização por parte de historiadores como unidade de medida para o progresso científico (1977). Kuhn, no entanto, apresenta uma distinção entre dois tipos de descoberta: em primeiro lugar, aquelas nas quais se sabe o que se busca de antemão – como, por exemplo, elementos químicos que completem a tabela periódica dos elementos. Em segundo lugar, aquelas resultantes da observação de anomalias que, ao menos inicialmente, contradizem expectativas em experimentos. Este segundo tipo pode acarretar mudanças como “o *redesign* de antigos aparatos e novas formas de se fazer antigas perguntas” (Kuhn, 1977, p.176, tradução do autor). Neste caso, uma possível consequência do sucesso de um experimento pode se converter na modificação ou na extensão das regras do jogo e de suas peças – o que indica a *tensão essencial* entre tradição e mudança da qual fala Kuhn: “O cientista deve ter um compromisso completo com a tradição à qual, se bem-sucedido, irá romper” (1977, p.235, tradução do autor).

Em *The Structure of Scientific Revolutions*, Kuhn identificava como objetivos do cientista “ser útil (...) explorar novos territórios (...) encontrar ordem (...) testar o conhecimento” (Kuhn, 2012, p37-38, tradução do autor), além de – em comparação direta a jogos – “solucionar um quebra-cabeça que ninguém tenha solucionado anteriormente, ou solucionado tão bem” (Kuhn, 2012, p37-38, tradução do autor). Neste sentido, a frase de Kuhn revela um aspecto competitivo que pode se mostrar bastante pronunciado. A descoberta da estrutura do DNA por James D. Watson e Francis Crick, tal como relatada por Watson, é um destes casos. Em seu relato autobiográfico *The Double Helix*, Watson se mostra um cientista competitivo, equilibrando-se entre “a ambição e uma noção de *fair play*” (Watson, 1980, p.3, tradução do autor), esforçando-se para “imitar Linus Pauling e derrotá-lo em seu próprio jogo” (Watson, 1980, p.32, tradução do autor). Como em uma corrida, não bastaria aos envolvidos descobrir a estrutura do DNA, mas descobrir *primeiro*. De fato, observa Knorr Cetina (1981), a constante busca por descobertas, publicações e citações necessárias para avanços profissionais faz com que cientistas se definam como profissionais que não tem tempo a perder. Em alguns casos, no entanto, estabelecer um autor, momento e local da descoberta científica pode parecer uma simplificação do processo de descoberta. De acordo com Galison, “[a] ideia de um momento específico da descoberta, talvez valiosa para comitês de premiações e livros didáticos de física, correspondem a pouco ou nada do registro histórico” (1987, p.127, tradução do autor), ainda que a competição possa desempenhar papel importante nos rumos da pesquisa, na definição de seus objetivos, e na decisão sobre quando encerrá-la. Latour (1987) também critica uma importância desproporcional atribuída aos realizadores de descobertas científicas e à preocupação de se estabelecer quem foi de fato o primeiro a alcançá-la, atribuindo todo o progresso a apenas um indivíduo. Por outro lado, alguns autores preferem reforçar o papel do indivíduo nas descobertas científicas. Nersessian (2008) e Csikszentmihalyi (2008) – ambos ligados à psicologia – enfatizam a importância do trabalho realizado por mentes individuais, apesar de reconhecerem a importância do meio científico no processo de construção desse conhecimento.

Algumas das motivações apontadas por Kuhn são também descritas por Latour & Woolgar, que destacam a satisfação do cientista em resolver problemas e comunicar suas soluções; Knorr Cetina, que aponta para a motivação de cientistas em explorar novas técnicas, gerar melhores experimentos e “[buscar] a teoria

“final” da natureza” (Knorr Cetina, 2009, p. 14, tradução do autor); e Turkle, que relata o prazer de cientistas em explorar o desconhecido e solucionar problemas difíceis. Esta noção de desafio está presente, também, na visão de Feyerabend, para quem a tarefa do cientista é “tornar mais forte o caso mais fraco” (1993, p.21, tradução do autor).

Deve-se levar em consideração uma variedade de objetivos de diferentes naturezas, a depender do campo de atuação dos cientistas. Gardner et al. descrevem quatro diferentes abordagens associadas a diferentes áreas da pesquisa genética: “o objetivo da descoberta” para acadêmicos, “a importância de ajudar os outros” para profissionais da área médica, “educação e informação do público” para clínicos, e “trabalhar para os objetivos da companhia” para cientistas posicionados na indústria (2002, p.75, tradução do autor). Por outro lado, objetivos podem variar de acordo com o domínio científicos. Para Keller, biólogos são menos superlativos do que, por exemplo, físicos: enquanto os últimos se dedicam à busca de uma “teoria de tudo”, os primeiros buscam “compreender a vida” (Keller, 2003, p.2, tradução do autor) – uma tarefa de esforços e resultados mais heterogêneos e provisórios, quando comparados aos de outras ciências.

Para seguir com a comparação entre jogo e ciência, precisamos compreender como os objetivos descritos anteriormente se convertem em resultados variáveis, quantificáveis e valorizados. Neste caso, dois paralelos podem ser estabelecidos: em primeiro lugar, podemos afirmar que a *variação* e *quantificação* dos resultados na pesquisa científica é consequência da *mensuração* e *análise* dos experimentos e, também, da *produção de textos* realizados a partir destas análises. Já a *valorização* do resultado é, geralmente, fruto, da *validação* dos experimentos e teorias, e que deve ser, em última análise, conferida pelos próprios cientistas: “E ao final, somos nós quem damos a resposta; somos nós mesmos quem, após intenso escrutínio, decidimos a respeito da resposta à questão que colocamos à natureza” (Popper, 2005 p.280, tradução do autor). Kuhn, de forma semelhante, aponta para o fato de que as comunidades científicas sejam, efetivamente, a única plateia e os únicos juízes do trabalho científico: “não há outra comunidade profissional onde o trabalho criativo individual seja tão exclusivamente endereçado e avaliado por outros membros da mesma profissão” (Kuhn, 2012. p.163, tradução do autor). Cientes deste fato, cientistas *buscam maximizar as chances de aceitação de seus trabalhos, empenhando-se em identificar opções que facilitem a validação*

*dos resultados* pela comunidade científica: periódicos mais favoráveis para publicação, temas em alta na comunidade, aliados e oponentes em potencial – de fato, responsáveis pela validação dos resultados podem ser os maiores adversários dos cientistas na busca por autoridade e crédito dentro da comunidade científica. (Knorr Cetina, 1981).

A promoção de um enunciado científico a um fato se assemelha, para Latour (1987), ao fechamento de uma *caixa preta*, uma vez que seu conteúdo se torna aceito como conhecimento tácito. Pode-se considerar, portanto, como objetivo do cientista, a transformação de suas teorias em caixas pretas – o que pode parecer contraditório à ideia da ciência como um jogo eternamente em aberto. Ainda de acordo com Latour, para que um enunciado se estabeleça como fato, geralmente precisa que as gerações seguintes de enunciados o sejam favoráveis. Este reconhecimento passa, também, pela quantidade de citações, registradas em índices de citações científicas. A necessidade de reconhecimento pelos pares é destacada, ainda, por Csikszentmihalyi, como condição para a validação do trabalho criativo (2013). Em alguns casos, no entanto, os critérios que definem o resultado de um experimento como satisfatório são dados por aqueles que o conduzem, e não por observadores externos (Galison, 1987). Finalmente, deve-se considerar o contexto do trabalho científico para compreender o que torna um resultado valioso, uma vez que ambientes acadêmicos e industriais podem oferecer diferentes métricas. Como descrito por Gardner et al.,

“o departamento de biologia de uma universidade pode buscar descobertas que façam avançar o conhecimento sobre mecanismos básicos do desenvolvimento orgânico, e sua medida de sucesso seria a publicação em um periódico de prestígio. Em contraste, o departamento de pesquisa de uma companhia farmacêutica pode ignorar quebra-cabeças sobre o desenvolvimento, priorizando buscar o modo mais barato e eficiente de se curar uma doença – e sua medida de sucesso seria uma fatia maior do mercado” (2002, p.71, tradução do autor).

A respeito da quantificação dos resultados, destacam-se a já mencionada matematização das ciências e, também, o prevalecimento de métodos quantitativos de pesquisa, percebidos como de maior importância para seu sucesso. A importância das medições está relacionada à certificação de resultados. Trata-se de um ponto destacado por Hacking, ao descrever o pensamento do filósofo Charles Sanders Peirce, para quem “medições (...) convergem, e o ponto onde convergem é, por definição, correto” (Hacking, 1983, p.59, tradução do autor). No ensaio *The Function of Measurement in Modern Physical Science*, Kuhn (1977) discute motivos e consequências dos atos de medição praticados por cientistas. Supostamente, um destes motivos seria a comparação do resultado de experimentos com suas teorias subjacentes, para fins didáticos. Kuhn exemplifica este ponto através de um diagrama – típico de livros didáticos – contendo (1) leis e enunciados no canto superior esquerdo, (2) o equipamento através da qual o experimento ocorre ao centro e (3) resultados numéricos da operação, a serem comparados aos propostos pela teoria. O diagrama está reproduzido na Figura 19 – e em muito se assemelha à divisão das etapas da ciência computacional em modelagem/simulação/análise de resultados.

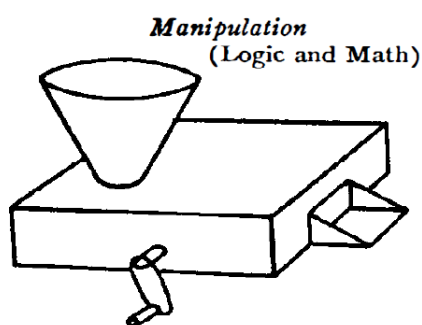
**Theory**

$$(x) \phi_1(x)$$

$$(x) \phi_2(x)$$

$$\dots$$

$$(x) \phi_n(x)$$



**Results**

Theory	Experiment
1.414	1.418
1.732	1.725
2.236	2.237

Figura 19 - Diagrama de livro didático (Kuhn, 1977)



Além de buscar confirmar teorias, dados numéricos como os resultados apresentados no diagrama poderiam, supostamente, sugerir novas leis e teorias. Kuhn, no entanto, propõe que a medição de resultados não visa necessariamente confirmar teorias, mas apresentar didaticamente os limites e consensos acerca destas teorias a seus novos praticantes. Além disso, medições podem detectar anomalias que – prevalecendo a sucessivas medições – dariam início a crises nos paradigmas vigentes e abririam as portas a novos. A sugestão de novas leis e teorias, por sua vez, não seria possível a partir de resultados numéricos confirmando teorias passadas, pois a busca de padrões só seria possível após a elaboração de instrumentos e medidas projetados a partir das teorias. Poincaré teria concordado com este raciocínio, tendo afirmado que “o instrumento de medida acaba sendo uma teoria, e é preciso compreender que o microscópio é um prolongamento mais do espírito que do olho” (1995, p.297) – uma observação o que se reflete na preocupação metodológica da pesquisa científica: “é o seu método de medir, mais do que o objeto de sua mensuração, que o cientista descreve.” (Poincaré, 1995, p.261). Tal posição pode ser encontrada mesmo no trabalho de Feyerabend, que afirma: “Instrumentos de medida são construídos de acordo com leis e suas leituras são testadas sob a premissa de que essas leis são corretas” (2011, p.44).

Para além da obtenção e análise de resultados, devemos discutir o *registro* e *reconhecimento* dos experimentos e enunciados produzidos a partir do trabalho científico. Estes são temas centrais do trabalho de Latour & Woolgar, que percebem o laboratório como um "sistema de inscrição literária" (1986, p.52, tradução do autor) e averigam o desenvolvimento de literatura científica – foco da atividade daquele espaço. A produção de artigos é apontada como o principal tema e motivação das conversas e interações entre cientistas. Produzir artigos, para os cientistas observados, é "o principal objetivo de suas atividades" (Latour & Woolgar, 1986, p.71, tradução do autor) – uma opinião semelhante à de Knorr Cetina, para quem o objetivo do cientista é transformar aquilo que constrói em “créditos na vida científica cotidiana através de publicações” (Knorr Cetina, 1981, p.4, tradução do autor). Para Latour & Woolgar, é através da produção literária que enunciados resultantes de suas pesquisas podem ser elevados de um *status* especulativo, avançando através de níveis de autoridade até serem estabelecidos como fatos científicos (desde que não refutados por outras declarações com

autoridade, especialmente as de pesquisadores concorrentes). "O objetivo do jogo científico é elevar um enunciado (...) na medida do possível ao *status* de fato [científico]" (Latour & Woolgar, 1986, p.181, tradução do autor). Pode-se, portanto, atribuir diferentes qualidades a diferentes enunciados e, em alguns casos, mensurá-los em escalas de valores. Popper, por exemplo, identifica duas classes: *enunciados universais* (que descrevem uma lei natural) e *particulares* (que se preocupam com eventos específicos), propondo uma regra: deve-se procurar leis universais e sistemas teóricos coerentes que possam explicar, através de relações de causa e efeito, eventos particulares observáveis. Reconhece, no entanto, que seria impossível verificar cada evento particular ao qual uma lei universal poderia ser aplicada – o que a mantém aberta à refutação. Popper discute também sistemas teóricos, descrevendo *sistemas axiomáticos* como rigorosamente estruturados, coesos e baseados em axiomas - pressupostos básicos sobre os quais o sistema teórico é construído, e a partir dos quais novos enunciados podem ser derivados. Axiomas são, portanto, enunciados do mais alto nível de universalidade. Além da universalidade dos enunciados, Popper aponta outras qualidades como *testabilidade*, e *simplicidade*. Graus mais elevados de simplicidade e testabilidade resultam em maiores chances de falseamento. Nesse caso, teorias com maior grau de testabilidade devem ser privilegiadas. Outra qualidade apontada por Popper é a *simplicidade*, não em um sentido estético ou de praticidade, mas em termos de universalidade e precisão. Teorias são consideradas simples na razão diretamente proporcional de seus graus de precisão e universalidade, quando podem ser mais generalizadas, aproximando-se de leis científicas. Assim, enunciados simples são mais desejáveis “porque nos dizem mais; porque seus conteúdos empíricos são maiores; e porque são mais testáveis” (Popper, 2005, p.128, tradução do autor). De forma semelhante, Poincaré afirmou que experimentos são bem-sucedidos na proporção em que permitem generalizações, indo “além de um fato isolado” (1921b, p.128, tradução do autor). Kuhn (2000) considera, no entanto, que quesitos como simplicidade, escopo e utilidade de uma teoria podem ser julgados diferentemente dependendo do pesquisador que os avaliam. Kuhn sugere, ainda, que a categorização de enunciados em termos de verdadeiro/falso são decorrentes da estrutura, lógica, e linguagem interna do domínio ao qual pertencem, devendo *mostrar-se coerentes e não contradizerem a si mesmos* (Kuhn, 2000).

Latour & Woolgar discutem a promoção de enunciados ao *status* de fatos científicos em termos lúdicos, os quais deixam transparecer não somente aspectos da variação, quantificação e valorização dos resultados, mas também da intensa *competição* que pode ocorrer entre enunciados, cientistas e grupos de pesquisa: "O objetivo do jogo é realizar todas as manobras possíveis para forçar o cientista (ou seus colegas) a admitir que outros enunciados não são igualmente plausíveis" (Latour & Woolgar, 1986 p.241, tradução do autor). Além disso, a construção de fatos científicos ordenados é extensivamente comparada pelos autores a *Go*, jogo de tabuleiro chinês, onde territórios são disputados ao longo de sequências de jogadas aparentemente aleatórias, mas que, ao fim da partida, resultam em um posicionamento ordenado das peças. Tal como peças de *Go*, argumentam Latour & Woolgar, enunciados científicos são posicionados estrategicamente, reforçando seus pares e cercando posições adversárias, até se consolidarem e se estabelecerem em seus campos. Este senso de competição também é observado por Csikszentmihalyi (2013) que, apesar de privilegiar a natureza autotélica do processo criativo, identifica um sentimento de realização quando o resultado desse processo se torna parte do domínio, visto que é comum a competição entre vários resultados diferentes por esse espaço.

De acordo com Latour & Woolgar, a pesquisa científica e seus resultados podem estar atrelados a vários tipos de ganho, tais como publicações, reputação, financiamento, credenciais e reconhecimento. Todos estes podem ser usados como moeda naquilo que Latour & Woolgar chamam do "ciclo de credibilidade" através do qual os cientistas – avaliados, em última instância, por seus pares - se posicionam e avançam em suas carreiras. Assim, a atuação de um cientista pode ser quantificada por seu *curriculum vitae*, que atesta suas qualificações, posição no *ranking* acadêmico, cargos, verbas, prêmios e publicações. A pesquisa científica pode ser descrita, portanto, como um esforço competitivo, um "campo de batalha" (Schally apud Latour & Woolgar, 1986, p.130, tradução do autor), onde participantes têm que lutar por seu "caminho para o topo" (Schally apud Latour & Woolgar, 1986, p.119, tradução do autor), buscando redefinir o campo de acordo com suas preferências em termos de dados, técnicas, métodos e recursos. Em outras palavras, buscando mudar "as regras do jogo" (Latour & Woolgar, 1986, p.122, tradução do autor).

Aspectos competitivos da produção científica também são identificados por Knorr Cetina (2009), e podem ocorrer entre cientistas, institutos, experiências paralelas, e até mesmo diferentes equipamentos em busca do mesmo objetivo. Mesmo no ramo da física de altas energias – onde aspectos colaborativos predominam – há um componente competitivo na forma como esses grupos são medidos em termos de velocidade, qualidade, medições e protótipos de seus trabalhos. O ramo da biologia molecular, por sua vez, apresenta uma competitividade individual mais acirrada em termos de liderança e autoria de trabalhos. Neste ramo, pesquisas científicas são estruturadas em torno de cientistas individuais, seus projetos e habilidades, e é ao redor destes profissionais que orbitam ferramentas, equipamentos e assistentes. O progresso da carreira científica em biologia molecular é visto como uma sucessão de estágios, da posição de aluno a de líder de laboratório – posição que pode depender da criação de um laboratório pelo próprio cientista.

Em complemento às informações obtidas através da pesquisa bibliográfica, julgamos importante acrescentar duas observações a respeito da valorização dos resultados: em primeiro lugar, observamos a presença de métricas para quantificação de produção científica (tais como números de artigos e citações, índice h, índice i10, fator de impacto, etc.) as quais servem de medida para o trabalho científico e parâmetro de avaliação de cientistas. Em segundo lugar, como mencionamos no capítulo anterior, observamos que, a despeito do ambiente competitivo, iniciativas científicas cada vez mais colaborativas e abertas tem se firmado como possível abordagem aos problemas cada vez mais complexos apresentados pela ciência (Franzoni e Sauermann, 2014). Neste contexto, novas formas de avaliação e colaboração podem ser facilitadas por uma nova mentalidade e novas tecnologias. Este ponto pode ser ilustrado por *Becoming a Rover*, ensaio complementar a *Simulation and its Discontents*, onde o cientista de computação da NASA William J. Clancey discute o trabalho coletivo realizado por um grupo multidisciplinar responsável por controlar remotamente uma sonda de Marte (ou MER, do inglês *Mars Exploration Rover*). Suas responsabilidades incluíam passar instruções ao veículo automatizado, revisar o progresso da missão e planejar metas de curto e longo prazo. Como o título do ensaio sugere, os cientistas precisaram renegociar suas identidades profissionais individuais em prol de uma identidade comum, mais colaborativa, personificada na própria sonda, com a qual os

pesquisadores estabeleceram uma conexão emocional. De fato, como no caso dos equipamentos de física de altas energias, a sonda é muitas vezes descrita como uma pessoa a ser nutrida, guiada e protegida. Esta conexão, aliada às imagens e dados obtidos em primeira mão, fizeram com que os pesquisadores se sentissem como se estivessem, de fato, em Marte. Tal esforço colaborativo envolveu o compartilhamento e negociação de recursos escassos - como o tempo de uso e capacidades da sonda. Tais limitações obrigaram cientistas a procurar conjuntos de dados que fossem "tão completos e tão atraentes quanto possível" (Baxter apud Clancey, 2009, p.117, tradução do autor). Neste contexto, métricas de produtividade científica foram substituídas por "uma percepção pessoal de quão bem eles serviram à sonda" (Clancey, 2009, p.113, tradução do autor). Percebemos ainda que algumas das observações de Clancey poderiam ser facilmente transportadas aos *videogames* modernos: a associação e conexão emocional com um *avatar*, o sentimento de se "estar lá", o esforço colaborativo e a busca por melhores estratégias. Enquanto duas primeiras observações remetem à virtualidade sugerida na subseção anterior, as duas observações finais apontam para as competências e ações do cientista, discutidas na próxima.

### 3.5 Esforço e recompensa

As duas características restantes da definição de Juul dizem respeito à relação entre o jogador e o jogo. Por um lado, o jogador deve se esforçar de forma a influenciar o resultado do jogo – o que, para Juul, indica haver um desafio ou situação de conflito. Por outro, o resultado do jogo influencia o estado emocional do jogador, que pode se sentir satisfeito por ganhar ou insatisfeito por perder.

Para Ian Hacking, “a ciência é uma liga entre duas faculdades, a racional e a experimental (...) uma colaboração entre diferentes tipos de pessoa: os especuladores, os calculadores e os experimentadores” (Hacking, 1983, p.248, tradução do autor). O esforço necessário pelo cientista pode, portanto, variar de acordo com natureza, etapa, ou seu papel dentro da pesquisa.

No âmbito racional, existem diversas capacidades e habilidades individuais as quais podem ser exercitadas pelo cientista. Poincaré apontava, dentre as formas de se chegar aos objetivos da ciência, a via da *lógica* – um trabalho lento e

sequencial – e, por outro lado, pela via da *intuição* – mais rápida e ousada. Neste caso, a última se encarregaria da invenção e a primeira, da demonstração. Para demonstrar a importância da intuição, Poincaré recorre à seguinte analogia: como num jogo de xadrez, é a intuição que permite ao jogador/cientista antever o desenvolvimento do jogo. A intuição poderia estar ligada aos sentidos, imaginação e capacidade de generalização (indução). Recentemente, no entanto, a quantidade e complexidade dos dados científicos dificulta o trabalho do cientista através apenas da intuição (Keller, 2003). Neste caso, computadores, os quais Turkle (2005) descreveu como objetos com os quais se pensa, tornam-se então indispensáveis na pesquisa científica atual.

Popper, embora reconhecesse o papel de ideias puramente especulativas, enfatizava a necessidade de se traduzir convicções e impressões sensoriais em enunciados solidamente lógicos e falseáveis que pudessem ser testados sistematicamente: “ideias ousadas, expectativas não justificadas e pensamentos especulativos são nossos únicos meios de/para interpretar a natureza: nosso único *organon* e instrumento para compreendê-la. E nós devemos arriscá-los para ganhar nosso “prêmio” (Popper, 2005, p280, tradução do autor). Ao mesmo tempo, Popper destaca o caráter *ativo* do cientista: “não chegamos ao acaso em nossas experiências (...) devemos ser ativos; nós temos que ‘fazer’ nossas experiências” (Popper, 2005, p280, tradução do autor). As habilidades necessárias ao cientista apontadas por Popper refletem uma *aderência às regras metodológicas*: além de desenvolver e testar suas teorias através do método científico, o cientista deve se esforçar, não para corroborá-las, mas justamente para refutá-las: “Nosso método de pesquisa não é defender [nossa teoria] para provar o quanto estávamos certos. Pelo contrário, devemos derrubá-la. Usando as armas da lógica, matemática e da técnica, tentamos provar que nossas expectativas eram falsas” (Popper, 2005, p.278-279, tradução do autor). Já Feyerabend, para quem um distanciamento das regras metodológicas é uma necessidade, comparava cientistas – particularmente os experimentalistas – a artistas, que trabalham baseados não somente no raciocínio, mas também na prática e nos sentidos. Trata-se de um pensamento compatível com a proposta da citogeneticista Barbara McClintok, vencedora do Prêmio Nobel de Fisiologia ou Medicina por sua pesquisa em transposição genética, cujo trabalho chegou a ser marginalizado pela comunidade acadêmica, e para quem a complexidade da natureza supera a imaginação humana. McClintok sugere que se “deixe o

experimento lhe dizer o que fazer” (apud Keller, 1985, p.162, tradução do autor). O caráter prático do experimento é destacado também por Knorr Cetina (1981), para quem, cientistas se empenham em raciocínio prático – o que implica um esforço cognitivo direcionado a questões que emergem no trabalho em laboratório e que dizem respeito ao contexto daquele ambiente, equipamentos, saberes e colaboradores locais.

A tensão entre objetividade e subjetividade é tema do trabalho de Daston & Galison (2010) que discutem características de diversas abordagens, surgidas ao longo da história, ao trabalho científico – em particular na produção de atlas científicos. Em primeiro lugar, a *truth-to-nature*, que recorre à subjetividade do cientista – seu poder de seleção e síntese – para gerar representações universais (de certa forma idealizadas) de fenômenos, tais como as folhas da Figura 20, ilustradas para transmitir características gerais de suas espécies – e não particularidades de espécimes individuais.

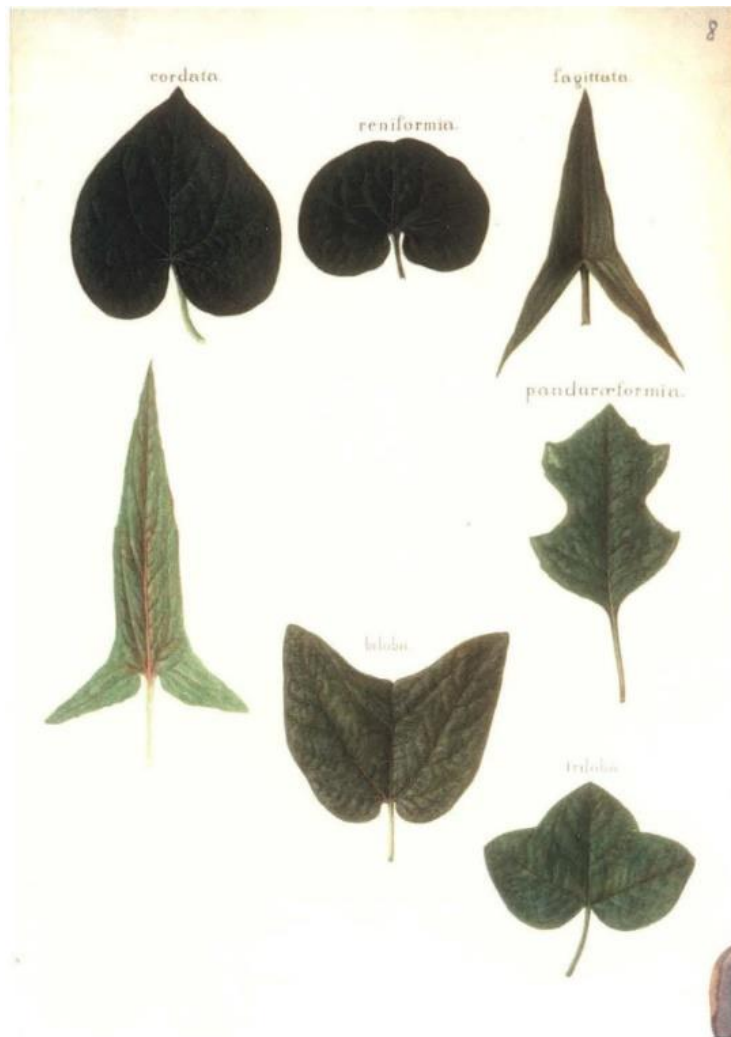


Figura 20 - Folhas - truth-to-nature (Daston & Galison, 2010)

Como contraponto a esta abordagem, surge em meados do século 19 a *objetividade mecânica* – caracterizada pelo autocontrole e neutralidade do cientista, mecanização de procedimentos, e o foco em espécimes e fenômenos individuais (Fig.21).



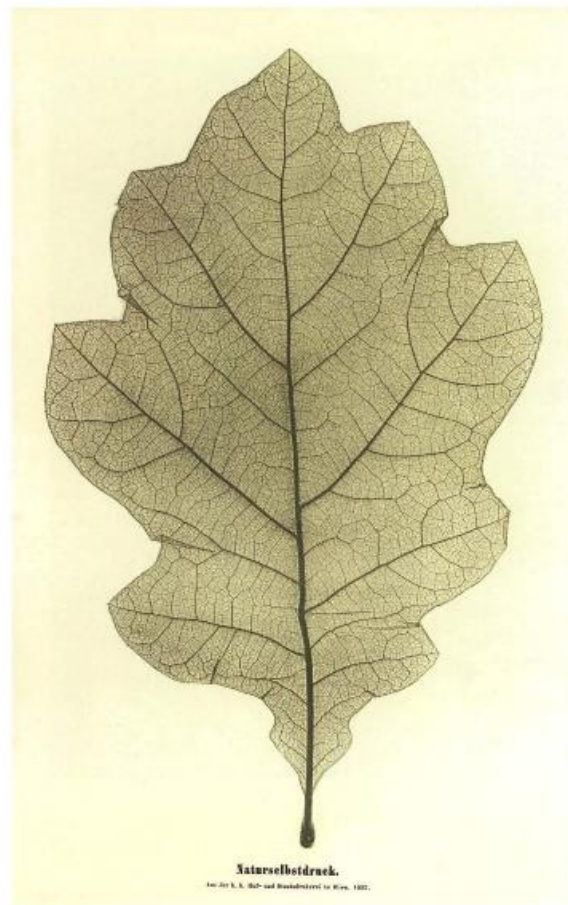


Figura 21 - Folha - objetividade mecânica (Daston & Galison., 2010)

Além da objetividade mecânica que, auxiliada por tecnologias de registro fotográfico, buscava registrar fenômenos de forma sistemática e automatizada, desenvolve-se, entre o fim do século 19 e início do século 20, a *objetividade estrutural*: proposta radical de busca à neutralidade, inspirada na lógica e na matemática, através de uma representação simbólica e estrutural que evidenciasse *relações* (Fig.22). Tal abordagem é condizente com a visão de Poincaré, em sua defesa da compreensão das relações entre as coisas e no uso da matemática para libertar-nos das contradições e incompreensões resultantes dos dados obtidos sensorialmente (1921b), e também com a visão de Bachelard, para quem uma “orientação sensualista da ciência” deveria ser substituída por uma abordagem “abstrata e matemática” (1996, p.131).

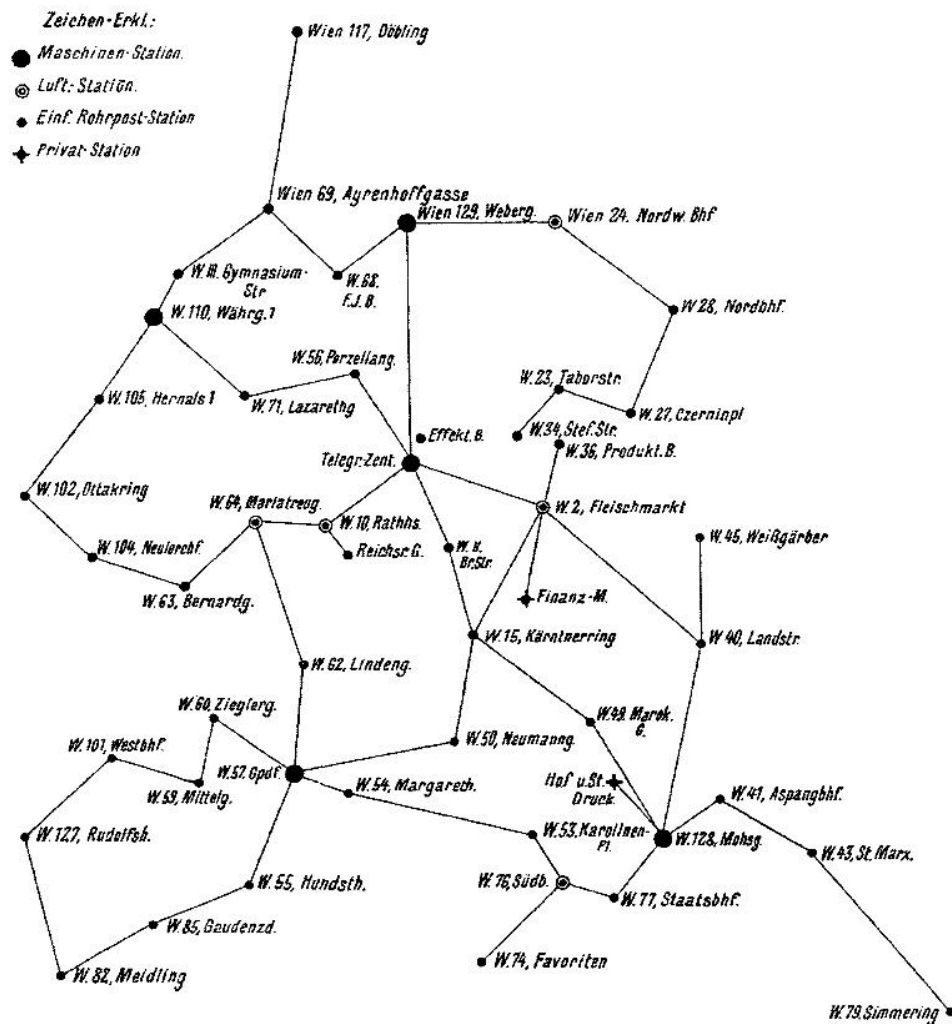


Figura 22 - Rede postal - objetividade estrutural (Daston & Galison, 2010)

Tais representações estruturais, poderíamos argumentar, parecem predecessores das recentemente popularizadas técnicas de visualização em *data science*.

Em resposta à objetividade mecânica, o *juízo capacitado* surge como abordagem que valoriza a subjetividade e a *expertise*, confiando na capacidade de interpretação e reconhecimento de padrões por parte do cientista e do leitor. Representações geradas de acordo com essa linha também fazem uso das tecnologias de registro fotográfico que possibilitaram a objetividade mecânica – mas podem incluir esforços do cientista em tornar claros tais padrões. É o caso, por exemplo, da radiografia reproduzida na Figura 23, que teve detalhes desenhados à mão sobrepostos à sua imagem original.

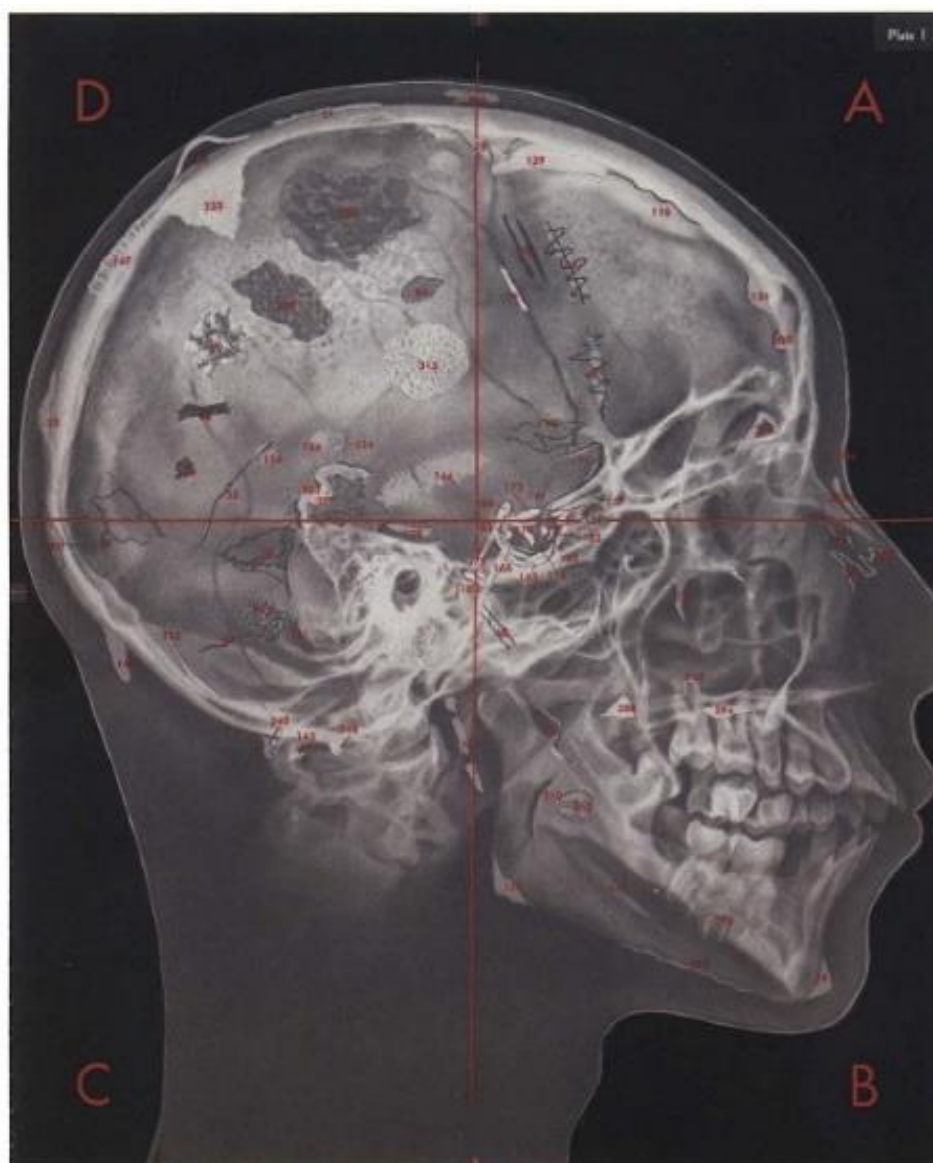


Figura 23 - Radiografia, julgamento treinado (Daston & Galison., 2010)

Finalmente, novas tecnologias surgidas no fim do século 20 e início do século 21 possibilitam novas abordagens. Desta vez, entretanto, a representação dá lugar à *apresentação*. Além das possibilidades estéticas abertas pela interação na navegação e geração de conteúdo – como a geração de paletas cromáticas para melhor visualização dos dados (Fig, 24) – há uma maior interação do cientista com o fenômeno. De fato, esta nova abordagem não se limita à observação e representação de fenômenos naturais, mas também à geração de artefatos de forma simultânea à observação e através dos mesmos instrumentos, como inscrições feitas através de equipamentos voltados à nanotecnologia (Fig.25). Nesse contexto, a persona do cientista se combina às de um engenheiro e de um artista (Daston & Galison, 2010).

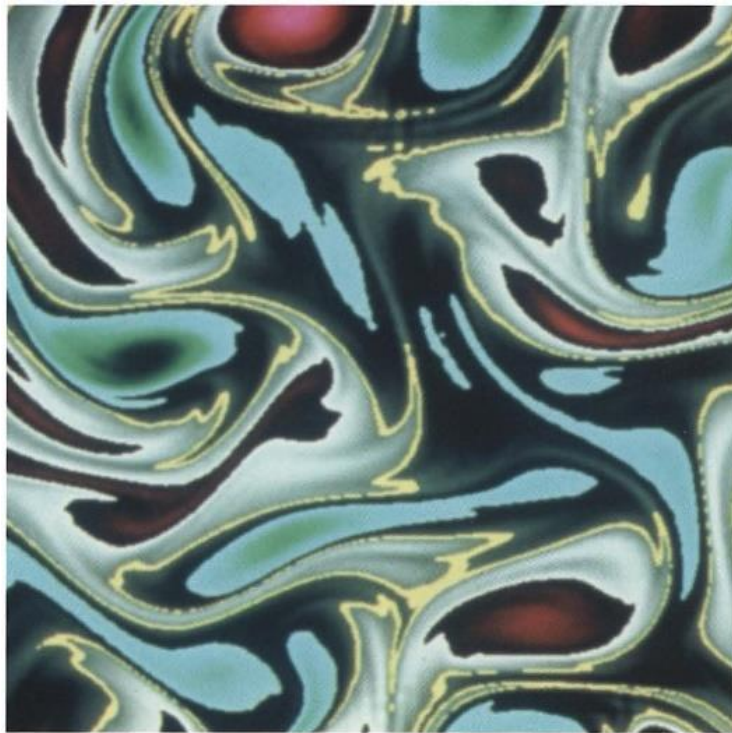


Figura 24 - Simulação de campos de vorticidade (Daston & Galison., 2010).

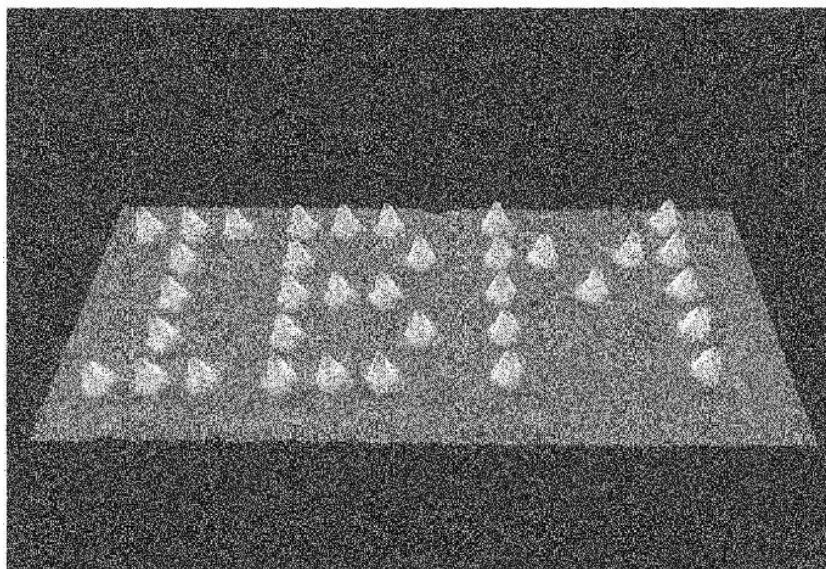


Figura 25 - Átomos manipulados para soletrar "IBM" (Daston & Galison, 2010)

A partir da análise de Daston & Galison, portanto, pode-se afirmar que o esforço do cientista reflete uma *atitude em relação à própria objetividade e subjetividade* e de sua *aderência a diferentes virtudes epistêmicas* no registro, interpretação, representação e apresentação de fenômenos e fatos científicos.



Podemos, no entanto, estender a discussão a respeito do papel da visualização de dados e da visualização científica na construção e estabelecimento do conhecimento científico – temas que gostaríamos de ligar à descrição do laboratório feita por Latour (2010), para quem o ambiente de trabalho científico descende, por um lado, do escritório (onde se realizam cálculos) e, por outro, do ateliê (onde materiais são transformados em criações). Tal descrição sugere diferentes esforços na obtenção e divulgação dos resultados propostos anteriormente, que podem ser ilustrados a partir de exemplos tirados da coletânea de ensaios *Representation in Scientific Practice Revisited* (Coopmans et al., 2010). Em primeiro lugar, há a *elaboração e interpretação da visualização de dados quantitativos*. Em *Visual Analytics as Artful Revelation*, Catelijne Coopmans discute a propagação de softwares de visualização de dados como o *Tableau*, os quais seriam capazes de gerar insight e revelar padrões que, ao menos de acordo com, permaneceriam ocultos. A geração de gráficos apropriados, no entanto, exige prática na *eliminação de ruído* e na *reconfiguração dos dados e de padrões de exibição* (Coopmans, 2014). Em segundo lugar, temos a *geração e manipulação de imagens para visualização científica*, tendo em vista – assim como no caso anterior – a interpretação dos dados e *insight*. Este é o caso ilustrado em *Drawing as: Distinctions and Disambiguation in Digital Images of Mars*, no qual Janet Vertesi (2014) comenta o papel da *transformação e processamento de imagens* da superfície de Marte, as quais tiveram suas cores transformadas para melhor visibilidade, facilitando a obtenção de informações a respeito do objeto de pesquisa. Em terceiro lugar, há o esforço em *interpretar imagens* através da *navegação* e da *interação com interfaces*, muitas vezes através da coordenação do olhar com gestos, tal como descrito por Morana Alač em *Digital Scientific Visuals as Fields for Interaction* (2014). Finalmente, pode haver esforço na *produção de imagens híbridas*, baseadas em dados científicos, mas complementadas por ilustrações hipotéticas ou fictícias como, por exemplo, o desenho de nano-robôs interagindo com células, tal como discutido por Martin Ruivenkamp e Arie Rip em *Nanoimages as Hybrid Monsters* (2014). Tais imagens podem ser eventualmente utilizadas em caráter ilustrativo para persuadir leitores a respeito de futuros desenvolvimentos científicos. No entanto, o uso de imagens ilustrativas deste tipo parece ser melhor aceito quando dirigido ao público não-especializado, e não a outros cientistas. Apesar das afinidades entre a ciência e as artes, ilustrações como aquelas parecem

destoar dos objetivos e valores da ciência. Kuhn, ao comparar ciência e arte, escreveu: “nas artes, a estética é, ela própria, o objetivo do trabalho. Nas ciências ela é, no melhor dos casos, uma ferramenta” (1977, p.342, tradução do autor). De fato, o estilo visual das figuras incluídas em artigos científicos é geralmente mais sóbrio que os equivalentes produzidos para o público geral. Um bom exemplo a este respeito é a diferença entre as duas representações do recém-descoberto sistema solar TRAPPIST-1. A primeira delas, incluída no artigo científico apresentando detalhes sobre os planetas do sistema e suas órbitas (Gillon et al., 2017), está reproduzida na Figura 26, e traz uma representação estilizada da referida órbita no canto inferior esquerdo. A segunda ilustração, utilizada em reportagem para divulgação no website da NASA (Landau, 2017), foi produzida com recursos avançados de computação gráfica e edição de imagem, provavelmente para maior apelo junto ao grande público (Figura 27).

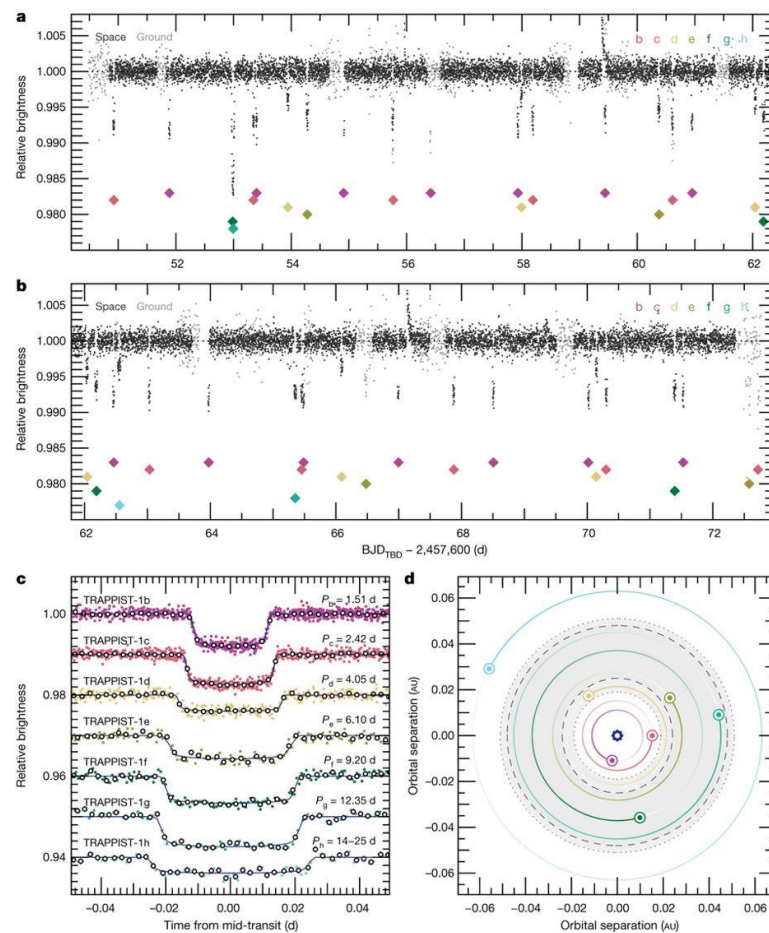


Figura 26 - Planetas e órbitas no TRAPPIST-1 (Gillon et al., 2017)

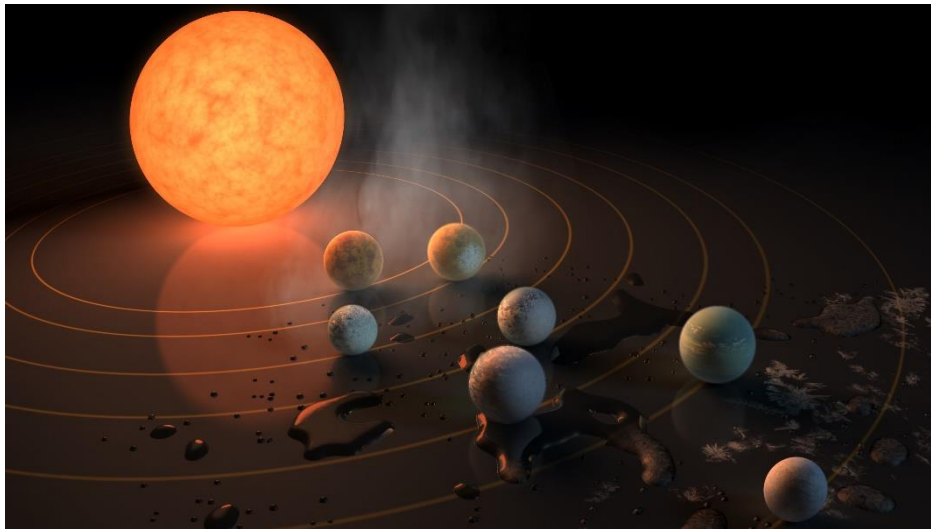


Figura 27 - Visão ilustrativa do TRAPPIST-1 (Landau, 2017)

Outro conjunto de esforços exigido pela pesquisa científica diz respeito à solução de problemas. Kuhn propôs que os paradigmas estabelecidos - teorias e práticas que servem de modelo para as comunidades e disciplinas científicas - são contínua e extensivamente testados através da ciência normal - pesquisas científicas baseadas em um paradigma estabelecido. É neste contexto que Kuhn compara a ciência normal com a resolução de quebra-cabeças. Nesse caso, dentre as habilidades necessárias ao cientista, identifica-se "a solução de todos os tipos de complexos quebra-cabeças instrumentais, conceituais e matemáticos " (Kuhn, 2012, p.36, tradução do autor). Outro autor a comparar o trabalho científico, sob o aspecto criativo, a quebra-cabeças é Csikszentmihalyi, que identifica a necessidade de *busca e reconhecimento de padrões* para afirmar: "o processo criativo começa com a sensação de que há um quebra-cabeça a ser resolvido, ou uma tarefa a ser cumprida. Talvez algo não esteja certo, em algum lugar há um conflito, uma tensão, uma necessidade a ser satisfeita"(2013, p.95, tradução do autor). Mais explicitamente, Kuhn afirma a importância da "habilidade adquirida de agrupar objetos e situações em classes" (2000, p.172, tradução do autor), e ainda: "para perceber semelhanças entre problemas aparentemente díspares" (1977, p.306, tradução do autor). Trata-se do que Knorr Cetina se refere como 'raciocínio analógico', percebido como inovador, onde "através da metáfora, dois fenômenos não comumente associados um com o outro são subitamente percebidos como tendo algum tipo de correspondência" (Knorr Cetina, 1981, p.50, tradução do autor). Esta habilidade também é enfatizada por Nersessian (2008), que identifica a transposição

e hibridização de conceitos através de diferentes domínios como fatores positivos na criação de modelos científicos. Este esforço, no entanto, inclui a *atenção para vieses subjetivos*, como a tendência humana de enxergar padrões em tudo (Popper, 2002), ou, ainda, de se valorizar certos padrões em detrimento a outros (Poincaré, 1921b). Csikszentmihalyi destaca a habilidade de *solucionar problemas de acordo com as regras fornecidas* e, ainda, a habilidade de *identificar novos problemas*. Embora associada à ciência normal, a solução de problemas técnicos – semelhantes a quebra cabeças – oferecem oportunidades e ferramentas para revoluções científicas (Kuhn, 2000). Neste sentido, Popper aponta a identificação de problemas – e não a mera observação – como o ponto de partida para o pensamento científico, argumentando que “*sem um problema, não há observação*” (Popper, 2010, p.6, tradução do autor, grifo no original). Hacking (1983) também se manifesta a respeito do considera uma supervalorização da observação, acrescentando que cientistas devem, sim, estar atentos – não somente aos fenômenos, mas aos detalhes de funcionamento dos instrumentos que os medem. Popper observa ainda que o cientista deve se empenhar na *formulação de hipóteses e teorias de forma objetiva*, em uma linguagem na qual possa ser objeto de estudo e abertas a experimentação. Tais hipóteses e teorias devem ser, então, postos a prova em processos de tentativa e erro – tema amplamente discutido por Popper em *Conjectures and Refutations* (2002), onde afirma que “a refutação de uma teoria (...) é sempre um passo adiante que nos leva mais próximos à verdade. E é assim que podemos aprender a partir de nossos erros” (p.xii, tradução do autor). Assim, tornam-se qualidades essenciais ao cientista a *capacidade de aprender a partir dos erros*, através da crítica e autocrítica e, Kuhn acrescentaria, através do apontamento dos erros a partir do grupo especializado no domínio científico do cientista (Kuhn, 1977). A busca pela solução de problemas pode ser, também, de natureza menos teórica ou abstrata: de acordo com Knorr Cetina (1981),

“o sucesso prático da ciência depende mais da habilidade de cientistas em analisar uma situação como um todo, pensando simultaneamente em diferentes níveis, reconhecendo pistas e juntando pedaços de informações diferentes, do que das próprias leis [científicas]. Como em qualquer jogo, a vitória depende menos das regras e mais do que é feito no espaço criado por essas regras.” (Knorr Cetina, 1981, p.3, tradução, do autor).

O esforço e investimento de cientistas na solução de problemas visando resultados e publicações, acrescenta Knorr Cetina, faz com que o trabalho em



laboratório não se resume a uma simples medição de dados para teste de hipóteses teóricas.

Latour & Woolgar observam que cientistas não descobrem, mas constroem - hábil e criativamente - fatos científicos. Para tal, devem se esforçar em *dominar os paradigmas* em que estão inseridos, compreendendo a "mistura complexa de crenças, hábitos, conhecimento sistematizado, realizações exemplares, práticas experimentais, tradições orais e habilidades manuais "(Latour & Woolgar, 1986, p.54, tradução do autor) que os compõem. Para Kuhn (2000), esse processo pode se assemelhar ao aprendizado de uma língua, onde aprendizes devem ser expostos a uma série de exemplos para adquirir vocabulário e sintaxe necessários para a interpretação dos problemas aos quais serão apresentados. Compartilhar desta linguagem é, como afirma Keller (1985), o que torna possível a *interpretação dos dados* obtidos. Além disso, como reforçou Galison (1987) é preciso se empenhar na *obtenção, seleção e tratamento dos dados*, o que pode exigir a *manutenção dos equipamentos que os capturam e processam*. Kuhn apontou para outros fatores que, além do domínio habilidoso das regras, poderiam desempenhar papel importante no progresso científico: descobertas acidentais (que acreditava ser um fator pouco relatado) e *experimentações livres e especulativas*. Para Kuhn, uma mentalidade *playful* – aquela elogiada em Geim e Novoselov pelo comitê do Prêmio Nobel – é o que pode levar a teorias especulativas que, caso bem-sucedidas, tornam-se base de um novo paradigma e, com ele, seus instrumentos e métodos e padrões. No entanto, deve-se destacar que mesmo as descobertas acidentais quanto aquelas resultantes de experimentações livres pressupõem um domínio das regras, por parte dos cientistas, que possibilitem a identificação de novas descobertas como tal.

A *habilidade manual* é destacada por Knorr Cetina, principalmente em biologia molecular, cujos experimentos exigem a manipulação hábil de objetos naturais para gerar conhecimento. Esses objetos são frequentemente mantidos e transformados através de habilidoso trabalho manual em pequenos laboratórios que servem como "repositório de ferramentas e recursos [e] materiais de trabalho para vários estágios do trabalho experimental" (Knorr Cetina, 2009, p.85, tradução do autor). Esses recursos, muitas vezes adquiridos de terceiros, permitem aos cientistas se concentrarem em metas e técnicas mais interessantes, que serão posteriormente incluídas ao repertório do laboratório. O conhecimento, as habilidades e o domínio desses processos são recursos valiosos para o laboratório. Curiosamente, cientistas

que realizam trabalhos experimentais muitas vezes superam erros através de uma estratégia que é semelhante à variação cega do processo de seleção natural: fazer mudanças aparentemente aleatórias no processo até que ele funcione. Tais mudanças, é claro, não são completamente aleatórias, pois são informadas pelo conhecimento prévio dos cientistas. Esta estratégia muitas vezes tem melhores chances de sucesso do que o aprofundamento no problema, especialmente quando há restrições de tempo para o experimento. Ainda a respeito de habilidades manuais, Latour certa vez comentou a respeito de um procedimento cirúrgico realizado em um porquinho da Índia: “Estamos agora em uma poça de sangue e vísceras. No capítulo anterior, admiramos a retórica do professor enquanto autor. Agora, percebemos as outras habilidades manuais necessárias para que se produza um artigo convincente mais tarde” (1987, p.66, tradução do autor).

Feyerabend, por sua vez, propôs que, apesar das alegações de imparcialidade do raciocínio lógico, a *persuasão* é essencial para a proposição de teorias científicas. Neste sentido, enfatiza-se a necessidade de deleite, ressonância emocional e poder imagético: “estilo, elegância na expressão, simplicidade na apresentação, tensão e narrativa da trama e conteúdo sedutor tornam-se características importantes de nosso conhecimento” (Feyerabend, 1993, p.118, tradução do autor). Para ele, apenas a discussão crítica não é suficiente para o estabelecimento de novas teorias. Latour & Woolgar reconhecem, como Feyerabend, o papel da persuasão na construção de fatos científicos – e sua necessidade de *não ser percebida* como persuasiva. Latour (1987) reconhece o valor da retórica na redação de artigos científicos, observando que a escolha de palavras pode transformar um artigo com ares de um simples relatório em outro, que se anuncie como uma contribuição inovadora. George Stigler, vencedor do prêmio Nobel de Economia, afirmou em entrevista a Csikszentmihalyi: “sempre considerei a tarefa de um cientista a responsabilidade pela persuasão de seus contemporâneos a respeito da força e a validade de seu pensamento” (Stigler apud Csikszentmihalyi, 2013, p.42, tradução do autor). A frase sugere que a força e a validade do pensamento não são suficientes para o pleno convencimento. Este poder de persuasão não é apenas dirigido a terceiros, mas também entre as equipes que efetivamente conduzem a pesquisa (Galison, 1987).

Há, no entanto, que se estabelecer limites para a persuasão – principalmente quando simulações computacionais estão envolvidas. Turkle apresenta uma série

de observações referentes aos riscos da qualidade persuasiva inerente às simulações por computador (Turkle, 2009). Para Turkle, a imersão provocada pela simulação é tão convincente que um *posicionamento crítico* se torna uma virtude. De fato, como indica Turkle, alguns cientistas propositadamente diminuem o apelo estético das imagens geradas por suas simulações.

Um posicionamento crítico em relação a simulações deveria, em princípio, ser facilitado pela transparência do software, ou seja, a disponibilização e clareza dos modelos matemáticos e processos computacionais responsáveis por executar a simulação. Nesse caso, é desejável que cientistas que trabalhem em ambientes computacionais tenham a *capacidade de ler (e se possível, escrever) código*. Além disso, a integração entre diferentes softwares de simulação pode tornar o processo mais difícil de verificar – caso em que a professora Diane Griffin recomenda que o trabalho em computador seja complementado pela perspectiva de uma “lente humano-computador” (Turkle, 2009, p.63), evitando o risco de se pensar apenas pela perspectiva da simulação. Afinal, a velocidade computacional é mais eficiente quando aliada ao raciocínio científico humano. O esforço de se dominar essa linguagem, no entanto, deve ser acompanhado de um bom *juízo quanto à qualidade dos dados fornecidos à simulação* (Galison, 1987).

A respeito da *transparência* das simulações, faz-se necessário enfatizar um ponto destacado pelo professor de antropologia Stefan Helmreich em seu ensaio *Intimate Sensing* (Helmreich, 2009): em *video games*, o termo *transparência* descreve a ausência de contato entre jogador e o funcionamento interno da simulação, ampliando a sensação de imersão. É possível, portanto, que a transparência das interfaces, nesse sentido, seja indesejável ou arriscada no software científico. Este é um ponto abordado em *The Second Self: Computers and the Human Spirit*, no qual Turkle discute de forma mais elaborada a relação entre as pessoas e os computadores – uma relação que muitas vezes passa pela programação que permite a indivíduos ganharem acesso às funções e códigos, garantindo ao mesmo tempo maior intimidade, conhecimento e poder sobre a máquina (Turkle, 2005). Turkle descreve como a transparência das regras nas quais se baseia o funcionamento dos programas de computador foi tornada opaca por decisões de design privilegiando interfaces visuais e gestuais em detrimento da linguagem de comandos. Para os responsáveis por estas mudanças, diz Turkle “jamais haveria a necessidade do usuário acessar os mecanismos subjacentes à

máquina. Na visão deles, usuários deveriam ser completamente liberados de pensar sobre a máquina” (2005, p.8, tradução do autor). Nesta nova lógica, apresentar uma interface como ‘transparente’ significava descrevê-la como de fácil operação, e não entender sobre o seu funcionamento. Ler e escrever código pode ser, portanto, uma forma de se aproximar da lógica matemática que governa o computador e a ciência computacional e, assim, possivelmente escapar da lógica da *caixa preta*. No sentido inverso – extremamente relevante na ciência computacional – cientistas devem se esforçar em *codificar e fornecer sistemas transparentes*, abertos ao escrutínio por outros cientistas – ainda que a opção inversa torne mais difícil a invalidação das teorias propostas. Trata-se, portanto, de uma boa prática e, também, da demonstração de *fair play*.

Turkle (2009) argumenta que o uso da simulação por computador gera uma ansiedade, uma sensação de que valores importantes podem ser negligenciados por usuários atraídos pela coerência e plasticidade das simulações e, embora encontrasse mais resistência entre membros mais antigos dos corpos docentes, alguns jovens estudantes também preferiam o trabalho manual como meio para compreensão e intuição científica. Neste caso, temia-se que a automatização do processo por computadores pudesse remover partes do processo mental e sugerir entendimento sem promover uma verdadeira percepção. À medida que o corpo se tornou, ao longo dos anos, cada vez mais envolvido na simulação (como, por exemplo, em ambientes de realidade virtual), a recepção crítica torna-se ainda mais difícil. Havia, também, o medo da perda de habilidades e métodos tradicionais. Reações moleculares, por exemplo, costumavam ser compreendidas através da brincadeira com modelos (físicos) e, também, através de performances que envolviam a movimentação dos membros. Para Nersessian (2008), este esforço de visualização através dos sentidos facilitaria a apreensão e simulação mental de conceitos científicos, um exemplo sendo a disposição de Richard Feynman em rolar no chão para sentir-se como um elétron. Desta forma, cientistas se beneficiariam deste “sentido sinestésico” (Nersessian, 2008, p.68, tradução do autor), tal como desenhos a mão livre facilitariam “insights geométricos” (p.72, tradução do autor). Práticas como estas reforçam a necessidade de *esforços imaginativos* que, de acordo com Nersessian, formam a base cognitiva para criação de modelos através da imaginação de situações hipotéticas, e podem ser auxiliados por várias formas de representação externa, como “[representações] linguísticas (descrições, narrativas,

comunicação verbal e escrita), equações matemáticas, representações visuais, gestos, modelos físicos e modelos computacionais" (p.92, tradução do autor). Tais esforços podem ser acompanhados pela manipulação e transformação dos elementos representados mentalmente. Nersessian se apoia em conceitos oriundos da psicologia, história e filosofia da ciência para analisar o *raciocínio baseado em modelos* construídos através de analogias, representações visuais e matemáticas, para a solução de problemas científicos. Ela enfatiza a importância da “construção, manipulação, avaliação e adaptação [de modelos]” (Nersessian, 2008, p.10, tradução do autor) na criação de representações conceituais anteriores à formalização de teorias. Trata-se de um *trabalho criativo*, diferente da lógica, mas o qual pode (e deve) ser examinado à luz da lógica posteriormente. A construção de tais modelos pode se basear em *analogias* entre o domínio científico estudado e outros domínios - um modelo híbrido - e ser aperfeiçoado através de modelos intermediários, para explicar o funcionamento e estrutura de sistemas pesquisados. Nersessian apresenta como exemplo o trabalho de James Clerk Maxwell, que se inspirou no funcionamento de sistemas mecânicos e hidrodinâmicos para criar representações conceituais de sistemas eletromagnéticos. Tais representações - continuamente avaliadas e aperfeiçoadas - incluíam detalhados diagramas e instruções de como se deveria imaginar o funcionamento do sistema. Vale observar que tal esforço, assim como a diagramação e construção de modelos, pode ser facilitado por recursos computacionais, capazes de armazenar e interpretar diferentes formas de representação (visuais, verbais ou matemáticas). A respeito das simulações por computador, Nersessian reconhece a ajuda que essas podem oferecer, aprofundando o raciocínio baseado em modelos – ressaltando, no entanto, que o raciocínio humano não necessariamente funciona como o computacional. Esforços no raciocínio científico e construção de modelos, sugere Nersessian, podem ser auxiliados pela *anotação e registro da evolução do raciocínio* em cadernos e diários – um processo que poderia facilitar, também, aquela que é uma etapa crucial no estabelecimento de enunciados científicos de acordo com Latour & Woolgar (1986): a *redação de artigos*.

A redação do artigo científico é descrita por Knorr Cetina (1981) como um processo de construção em laboratório. Este processo envolve uma dose de estratégia literária, incluindo uma escrita persuasiva e julgamento sobre o que deve ser incluído ou deixado de fora no relato da experiência. Há, também, a

contextualização deste relato, que segue uma estrutura clássica, podendo haver a tradução de resultados quantitativos em termos qualitativos. Além disso, argumentos são organizados no artigo de modo a descrever a obtenção de resultados como consequência lógica do problema de pesquisa. Ao longo do processo de escrita, frases são redistribuídas ao longo do texto – às vezes removidas – além de reescritas para maior ou menor assertividade, dependendo da estratégia dos autores e necessidades do argumento em construção. As várias seções do artigo científico são, portanto, redigidas de acordo com esses interesses. A *introdução*, por exemplo, busca contextualizar o trabalho, valorizar seus resultados e *convencer o leitor da relevância* do artigo. A seção referente às *metodologias*, por sua vez, descontextualiza o trabalho de seu local de realização, propondo um *protocolo, tarefas e instruções* para a reprodução do experimento, omitindo, no entanto, as negociações, acidentes e dinâmicas internas que ocorreram ao longo do experimento – evitando revelar questões problemáticas no desenvolvimento do trabalho. A seção dedicada à *discussão dos resultados* é novamente descontextualizada do trabalho laboratorial, sendo dedicada à comparação os resultados com enunciados produzidos por outros autores, sugerindo e argumentando em favor – ainda que de forma implícita – da superioridade dos resultados em relação a seus antecessores, tomando cuidado em apontar para aspectos inovadores do trabalho e, ao mesmo tempo, reconhecer esforços de trabalhos precursores, cujos autores possam ser tomados como aliados. Por outro lado, é necessário apontar aspectos da pesquisa que tornam trabalhos anteriores, realizados por terceiros, inexatos, desatualizados ou obsoletos. Neste caso, o caráter competitivo da ciência se faz presente: “muito do que é escrito em um artigo científico é escrito *contra* outra pessoa” (Knorr Cetina, 1981, p.126, tradução do autor).

Sobre aspectos psicológicos da invenção e trabalho científico, Poincaré argumenta que matemáticos frequentemente atingem momentos de “iluminação súbita” ou, nas palavras de Csikszentmihalyi, o “momento eureka”. Estes momentos seriam resultados de trabalho inconsciente, após longos períodos de trabalho aparentemente improdutivos (Csikszentmihalyi se referia a este intervalo de tempo como “períodos de incubação” após a “imersão no problema”). Poincaré aponta também para momentos de inspiração provocados pela beleza e elegância da matemática. Porém, ressalta que estes momentos de iluminação e inspiração podem

ser validados somente por rigorosos cálculos em momentos de trabalho consciente – a fase de “elaboração”, tal como definida por Csikszentmihalyi.

Popper, voltado à lógica, exclui de sua análise fatores psicológicos da descoberta científica – embora acredite em uma ‘intuição criativa’ e um ‘elemento irracional’ que possam levar a novas ideias. Estes fatores, no entanto, são extensivamente pesquisados por Csikszentmihalyi, que se debruça sobre a criatividade – particularmente aquela capaz de gerar novas teorias e transformar os domínios onde estão inseridas. Criatividade, argumenta, não é apenas uma questão de “*insight* súbito”, mas a culminação de longos períodos de trabalho e a sinergia entre muitos fatores - incluindo ambientais. Ela depende não somente da percepção pessoal de um indivíduo, mas também de sua interação com o domínio em que está inserida. Nesse contexto, o indivíduo deve aprender as regras do domínio antes de contribuir criativamente. Embora sorte seja frequentemente apontada como fator determinante em descobertas científicas (assim como proposto por Kuhn), Csikszentmihalyi enfatiza que tais descobertas não são inteiramente feitas por acaso, pois dependem de profundo e vasto conhecimento anterior de seus descobridores (assim como ocorre na variação cega identificada por Knorr Cetina). Este conhecimento prévio, por sua vez, é construído sobre um grande *investimento e gerenciamento de tempo, trabalho, atenção e dedicação* por parte do cientista – ponto também colocado por Poincaré, que afirmara quase um século antes, em *Ciência e Método*, a necessidade de gerenciamento de tempo, cuja escassez força o cientista a decidir quais fatos merecem investigação e, também, o quanto simplificar cálculos para propósitos práticos de aplicações da ciência (Poincaré, 1921). Csikszentmihalyi, no entanto, enfatiza também a capacidade do indivíduo criativo em *internalizar as regras e conteúdo do domínio*, além dos critérios de validação do campo. Tal internalização dependia em maior grau da *memorização* – habilidade mais prontamente associada ao conhecimento no passado, quando o registro de informações era mais raro e custoso (Csikszentmihalyi, 2008).

Outro ponto colocado por Csikszentmihalyi é a complexidade da personalidade do indivíduo criativo e sua *capacidade de alternar e transitar entre extremos* tais como: disciplina e brincadeira; imaginação e senso de realidade; orgulho e modéstia; tradição e rebelião; esperteza e ingenuidade; extroversão e introversão; paixão e objetividade; prazer e desconforto; pensamento convergente (racional) e divergente (flexível) – neste último caso em acordo com Kuhn (1977),

para quem o cientista deveria ser, simultaneamente, tradicionalista e iconoclasta. No contexto do presente estudo, a personalidade criativa, tal como descrita por Csikszentmihalyi, pode ser comparada a estudos sobre preferências do jogador. Em termos gerais, extremos como disciplina, orgulho e (a superação do) desconforto relacionam-se a uma mentalidade *gameful* enquanto, por outro lado, brincadeira, modéstia e alegria conectam-se a um modo *playful*. Assim, a capacidade de transição entre esses extremos sugere que o software deve ser aberto o suficiente em seu uso para permitir a expressão das duas mentalidades. Além disso, é interessante notar que jogos também recorrem simultaneamente a ambos os lados do espectro, exigindo pensamento racional e flexível; operando, como proposto por Juul, na fronteira entre imaginação e realidade; despertando a paixão de jogadores operam objetivamente; gerando frustração e alegria, nos jogadores, ao longo da experiência. Além disso, Csikszentmihalyi reconhece que a criatividade moldou a cultura humana - ciências, artes, religiões, política, etc. Uma visão de semelhante à de Huizinga, tal como apresentada em *Homo Ludens*, sobre o impacto dos jogos na civilização. Csikszentmihalyi (2013) também discute o impacto do ambiente (centros de conhecimento, recursos e o campos de estudo) no processo criativo, sugerindo que este se beneficiaria de uma atmosfera de colaboração e competição entre centros de pesquisa e profissionais. Por outro lado, sugere que um ambiente criativo também significa um local inspirador, que estimule o bem-estar e novas ideias. Neste caso, faz parte do esforço do cientista a *construção ou manutenção de um ambiente criativo* que, como discutimos anteriormente, pode ser o próprio laboratório. No ambiente do laboratório, como discute Knorr Cetina (1981), a exploração de ideias e recombinação de conceitos passa por aspectos instrumentais da pesquisa, tais como a disponibilidade de equipamentos, a oportunidade de *experimentar novas técnicas, experiências passadas e conselhos* – havendo nesse esforço um aspecto oportunista na forma como cientistas podem *explorar os recursos a sua volta* – sejam esses recursos materiais ou teóricos – o que não implica em falta de ética, objetividade ou método por parte dos cientistas.

Um aspecto criativo recorrente da pesquisa científica se refere à *construção e manipulação de modelos* que acompanha e dá suporte ao raciocínio. As descobertas de James D. Watson e Francis Crick sobre a estrutura do DNA, por exemplo, resultaram parcialmente do tempo gasto manipulando livremente um modelo físico da estrutura construído em seus escritórios – algo que poderia ser



considerado precursor daquele virtual simulado em *Foldit*, jogo de ciência cidadã. Watson, um cientista extremamente competitivo, via-se em uma corrida contra Linus Pauling, que já lançava mão de modelos em suas investigações. A este respeito, Watson escreveu: “a principal ferramenta de trabalho [de Pauling] era um conjunto de modelos de moléculas parecido com brinquedos de crianças de pré-escola (...) tudo o que precisávamos fazer era construir um conjunto de modelos e começar a brincar” (Watson, 1980, p.34, tradução do autor).

A importância das colaborações, ambientação e contextualização da pesquisa são enfatizadas por Latour & Woolgar, que reforçam a influência de restrições práticas, locais, profissionais e temporais sobre as pesquisas. Estas são criadas através do intercâmbio informal de informações entre colegas e práticas coletivas, não sendo apenas um produto de insight individual súbito e abstrato. Pode haver, portanto, para além de habilidades individuais, uma necessidade de *saber trabalhar colaborativamente com outros cientistas, equipes e instituições*. Assim, cientistas devem saber *investir seu trabalho e credibilidade* em esforços coletivos, *gerenciar recursos* e *traçar estratégias* de pesquisa. Este trabalho envolve a constante negociação do cientista em seu grupo (e do grupo em um coletivo maior). Através desta habilidade colaborativa e da capacidade do cientista de se inserir na cultura de sua instituição – afirma Feyerabend – facilita-se a difusão e valorização de teorias. Desta forma, cientista e instituição conferem prestígio e reputação um ao outro.

Knorr Cetina (2009) destaca os esforços colaborativos em física de altas energias, onde iniciativas são estruturadas de forma comunitária e onde a individualidade do cientista não é tão importante quanto a coletividade formada por institutos, equipes, grupos de trabalho e experimentos com os quais cientistas desenvolvem uma conexão emocional e um profundo sentido de cooperação. Neste ambiente, a participação dos cientistas é negociada em termos relativamente igualitários, em busca de um objetivo comum, e negociada através de discussões. Em alguns casos, discussões formais são registradas através de sua *documentação na forma de relatórios*.

No contexto da física de altas energias, esse senso de coletividade é importante pois, dado o volume de conhecimento, esta precisa ser *gerenciada coletivamente*. Também devido à imensa complexidade e tamanho das experiências HEP, este espírito comunal facilita o *planejamento a longo prazo*, que se estende

além do período de tempo para suas experiências atuais. Por outro lado, as *perspectivas de curto e médio prazo* são mantidas por cronogramas de experiências visando objetivos, atividades e resultados específicos.

Knorr Cetina (2009) ressalta que, apesar do caráter comunitário na física de altas energias, as individualidades dos cientistas não são completamente apagadas. Isto é particularmente verdadeiro no início das colaborações, quando os cientistas de reputação estabelecida tentam obter financiamento e atenção ao seu trabalho e tecnologias. Grupos são formados e reunidos, muitas vezes com base em preferências pessoais e afinidades profissionais. Em alguns casos, esses relacionamentos são construídos sobre confiança e “fofocas técnicas”: conversas informais e privadas sobre o trabalho e o comportamento de terceiros. Individualmente ou em grupos, cientistas parecem escolher colegas em quem confiam para trabalhar - e que possam trazer, também, mão-de-obra e financiamento. Como contraponto ao aspecto comunitário da física de altas energias, Knorr Cetina apresenta o ambiente de trabalho em biologia molecular como mais competitivo – mesmo nas colaborações. Neste ambiente, o compartilhamento de conhecimento entre colegas poder ser percebido como *serviços a serem negociados e permutados*, e onde observa-se um maior esforço pelo *reconhecimento a contribuições individuais* – um ambiente comparado por Knorr Cetina (1981) a um sistema econômico, “um mercado de posições onde as commodities são cientistas” (Knorr Cetina, 1981, p.77, tradução do autor). O trabalho coletivo em laboratórios de biologia molecular, portanto, tende a ser centralizado em torno de um líder que, além acompanhar ativamente as pesquisas, precisa *gerir e promover o laboratório* (Knorr Cetina, 2009). Latour (1987) observou que profissionais nesta posição fazem movimentos para fora do laboratório, saindo do isolamento do laboratório onde praticariam o chamado de ‘*hard science*’.

A colaboração pode envolver vários especialistas de um mesmo domínio ou contar com a participação de especialistas de outras áreas. Assim, pode ser necessário para o cientista realizar *esforços colaborativos multidisciplinares*. A matemática, por exemplo, é descrita por Poincaré como capaz de “fornecer um instrumento para o estudo da natureza” (Poincaré, 1995, p.89), favorecendo uma íntima colaboração entre cientistas e matemáticos. Popper, por sua vez, destaca a necessidade de colaboração entre o teórico – que apresenta questões claras e

inequívocas – e o experimentador – que busca responder, refutando ou corroborando, a estas questões. Turkle relata a preocupação, por parte de cientistas, de se tornarem excessivamente dependente de engenheiros e cientistas da computação. De fato, como observado pela própria Turkle e destacado por Daston & Galison, há uma aproximação mútua entre ciência e engenharia fortalecida por tecnologias que permitem maior manipulação, criação e intervenção sobre fenômenos e objetos naturais. Por último, destacamos a colaboração entre cientista e designer – que pode ser importante na geração de visualização de dados (Coopmans, 2014).

O caráter coletivo da ciência, aliás, é sublinhado por vários autores. Kuhn, por exemplo, critica a definição do método científico como algo que possibilite a produção científica por qualquer indivíduo. Para ele, apesar da prática individual da ciência, a construção deste conhecimento passa necessariamente pelos grupos especializados nos domínios científicos (Kuhn, 1977). Latour, por sua vez, relaciona o aspecto coletivo ao desenvolvimento das comunidades científicas, criticando a exaltação do sujeito cartesiano:

“no momento mesmo em que se instaura uma comunidade científica europeia, questionadora, eficaz, bem equipada, com revistas renomadas, protocolos, experiências públicas, refutações, reproduções, patrocínios mais ou menos estáveis, incluindo academias, instrumentos mais bem estandardizados, coleções bastante completas, vale dizer, no momento em que essa comunidade pode finalmente exclamar : “*Nós pensamos, e graças ao fato de sermos muitos, a termos apoio, sermos instituídos e instrumentados, temos acesso ao verdadeiro*”, o que acontece? É o momento escolhido por René Descartes em seu cômodo aquecido em Utrecht para inventar o *cogito*. “Penso. O pensamento pensa em mim”” (Latour, 2011, p.97).

Latour (1987) compara a construção de fatos – particularmente na tecnociência – a um jogo de *rugby* onde o fato em construção é passado de mão em mão – por cientistas, técnicos e outros atores da sociedade. No caso da tecnociência, no entanto, o fato não é simplesmente conduzido, mas construído coletivamente, e pode modificar-se da ideia original do cientista à medida que se adapta a condições técnicas e necessidades concretas da sociedade onde se insere. À medida que o número de pessoas envolvidas no estabelecimento do fato cresce, o cientista enfrenta um paradoxo: precisa *aumentar o número de envolvidos para que o fato se estabeleça*, e, também, *controlar ou diminuir este número para que o fato – neste processo – não se afaste da ideia original* do cientista.

Knorr Cetina (2009) e Galison (1987) enfatizam, em suas respectivas obras, o caráter coletivo do campo de física de altas energias. Para o segundo, a organização e reorganização dos grupos de pesquisa é fundamental para o avanço dos experimentos. A esta visão fundamentalmente coletiva da ciência, se contrapõe Csikszentmihalyi (2008), para quem as maiores inovações científicas são frutos de mentes individuais, não necessariamente associadas a instituições ou equipamentos de última geração – ainda que indivíduos não possam, hoje, contribuir na mesma proporção que grandes centros de pesquisa.

Finalmente, temos o *apego* do jogador ao resultado, apontado por Juul como uma característica definidora dos jogos. Juul afirma que o jogador “pode se sentir genuinamente feliz ao ganhar, e triste se perder” (Juul, 2005, p.40, tradução do autor). Juul concede, no entanto, que esta é uma característica psicológica do jogo, e que há casos de jogadores (a quem chama de “estraga-prazeres”) que não demonstram satisfação ou insatisfação com o resultado dos jogos.

Transpor o apego ao resultado para o campo do trabalho científico parece, em princípio, uma tarefa simples. Em primeiro lugar – e aí reside uma considerável diferença entre o jogo (que, na definição de Caillois, deve ser não-produtivo) e o trabalho científico – bons resultados podem surtir efeitos positivos no *desenvolvimento da carreira do cientista*, e mesmo em possibilidades de *verbas para pesquisa* (Latour & Woolgar, 1986), reconhecimento e prêmios (Csikszentmihalyi, 2013). Em segundo lugar, independentemente de benefícios profissionais, há uma sensação de dever cumprido quando *o fruto do processo criativo se torna parte do domínio* no qual está inscrito (Csikszentmihalyi, 2013). Voltamos, nesse caso, às motivações intrínsecas de cientistas – como a resolução de problemas e no desenvolvimento de novos saberes e técnica – que seriam alguns dos objetivos do jogo da ciência tal como observados por Kuhn, Latour & Woolgar, Knorr Cetina e Turkle, registrados na seção anterior.

A busca por resultados positivos – e, portanto, satisfatórios para aqueles que os obtêm – parece contradizer a ideia de Popper, discutida anteriormente, segundo a qual cientistas devem buscar derrubar suas próprias hipóteses, e para quem a refutação de uma teoria deveria ser vista como um “grande sucesso” (2002, p.329, tradução do autor). De fato, por apego às suas teorias, cientistas chegam a tentar provar como incorretos seus próprios experimentos demonstrando o contrário (Galison, 1987). O próprio Popper, no entanto, aconselhava cientistas a

transformarem esses reveses em melhores teorias e experimentos, acrescentando que se ganhava “um novo problema, mais focado e afiado” (2010, p.13, tradução do autor). Poincaré, de forma semelhante, aconselhava cientistas a não se abaterem com resultados negativos, uma vez que a negação de uma hipótese poderia representar “uma oportunidade inesperada para descobertas”, indicando “algo inesperado e extraordinário” (1921b, p.134, tradução do autor).

No entanto, o apego do jogador ao resultado, tal como proposto por Juul, parece não abarcar uma das características mais importantes dos jogos – justamente aquela que se busca alcançar com a gamificação: a diversão. Ao atribuir a satisfação do jogador à obtenção de resultados positivos, Juul desconsidera o fato de que é possível apreciar o jogo apenas pelo prazer de jogá-lo. Assim, propomos que o apego do jogador, tal como do cientista, tem como objeto não somente o resultado, mas o *processo*.

Poincaré – que, como vimos anteriormente, relacionava a ciência ao lazer – apontava para uma direção semelhante ao defender a *ciência pela ciência*. Este fim em si mesmo, um caráter *autotélico* da ciência, também é identificado por Csikszentmihalyi, a quem foi relatada, por cientistas entrevistados, “a exaltação na busca pela verdade e pela beleza”. (Csikszentmihalyi, 2013, p.122, tradução do autor). Assim, mesmo ainda que não obtenha resultados positivos, o esforço do cientista não é considerado um desperdício pois, de acordo com Csikszentmihalyi, “atuar dentro das regras do domínio é recompensador por si só; eles continuariam fazendo o que fazem mesmo que não fossem pagos para tal, apenas pelo simples propósito de fazer aquela atividade” (2013, p.37, tradução do autor). O trabalho científico – e a busca por novas maneiras de discutir a natureza – pode ser tão entusiasmante quanto o jogo, dispensando a necessidade de estímulo externo (Csikszentmihalyi, 2008). Ambas as atividades podem ser muito difíceis e, também, divertidas. Pessoas trabalhando criativamente “experimentam de forma divertida a imersão e concentração em tarefas extremamente difíceis, como uma aventura, uma diversão emocionante (...) sem saberem se é trabalho ou brincadeira” (Csikszentmihalyi, 2013, p.106, tradução do autor). Este entusiasmo que aproxima o processo criativo do jogo é atribuído por Csikszentmihalyi (2008) ao estado psicológico definido e popularizado por ele mesmo como estado de *fluxo*: experiências inerentemente divertidas, recompensadoras em si mesmas. Os elementos que compõem este estado são: (1) objetivos claros; (2) *feedback* imediato

em resposta às ações; (3) equilíbrio entre desafios e habilidades; (4) fusão entre ação e consciência; (5) ausência de distrações; (6) despreocupação com fracasso; (7) perda da autoconsciência; (8) noção de tempo distorcida; (9) atividade se torna autotélica. Jogos (particularmente os eletrônicos) são frequentemente celebrados como a mídia perfeita para este estado psicológico. Deve-se, no entanto, enfatizar uma diferença entre o jogo e o processo criativo: enquanto o primeiro pode apresentar *feedback* constante para o jogador, o segundo pode exigir, de forma mais frequente, a internalização dos critérios do domínio para que os indivíduos possam fornecer *feedback* a si mesmos.

Csikszentmihalyi não é o único autor a descrever experiências de estado de fluxo na prática científica: Knorr Cetina descreve cientistas como pessoas motivadas que trabalham, por escolha própria, durante períodos de folga, capazes de experimentar uma espécie de transe no qual “esquecem-se de si mesmos e do ambiente à sua volta” (Knorr Cetina, 2009, p.170, tradução do autor). Além disso, Knorr Cetina descreve experimentos de física de altas energias como atividades estruturadas em torno do entendimento (conhecimento técnico de equipamentos e processos); da observação (compreender o que se está fazendo) e da descrição (o registro dos resultados). De certo modo, tal descrição se assemelha às condições para estados de fluxo em jogos (clareza das regras, das ações, e de *feedback*). Turkle (2009), por sua vez, descreve como simulações computacionais, assim como *videogames*, são capazes de fazerem usuários sentirem-se “em alfa” (no original em inglês, “*in the zone*”). Em outra ocasião, Turkle apontou para a programação de computadores como uma espécie de experiência “zen” (2005). Tais observações sugerem que o software científico tem potencial, ao menos em alguns casos, para potencializar experiências de fluxo e, conseqüentemente, o apego – ou a conexão emocional – do cientista ao processo científico e seus resultados. Como descreveu Kuhn, o fascínio pela solução de problemas “pode parecer uma obsessão – e é mais do que suficiente para torna-lo um fim em si mesmo. Para aqueles que se empenham, nenhum outro objetivo é necessário, embora indivíduos frequentemente tenham vários deles” (2000, p.251, tradução do autor).

Há, no entanto, outra interpretação possível para *apego* – mais próxima ao significado usual do termo, e oposta a uma visão de acordo com a qual a prática científica equivale a um conflito (contra problemas, outros cientistas ou a própria natureza). De acordo com Keller (1985), é possível se aproximar do objeto de

pesquisa com afeição e empatia – uma abordagem que exemplifica a partir do trabalho de Barbara McClintock, que descreveu problemas de pesquisa como “coisa[s] estimada[s] por você durante um certo tempo (...) por um breve período, você se sente fortemente apegada a aquele pequeno brinquedo” (apud Keller, 1985, p.164, tradução do autor).

### 3.6

#### Conclusão do capítulo

Buscou-se, ao longo deste capítulo, identificar características da ciência que pudessem ser tomadas como elementos fundamentais do jogo e, também, relacionar tais características àquelas propostas pelas definições de Juul. A contemplação e valorização destes pontos – tais como as características do software científico apontadas no capítulo anterior – tornariam processos de gamificação mais orgânicos e adequados àquele tipo de software. As próximas subseções se dedicam a apresentar um sumário das informações colhidas, além de discutir características da ciência, tais como identificadas nas seções anteriores, capazes de gerar tensões entre a visão sistêmica dos jogos e a realidade da prática científica. A partir desta discussão – e das características levantadas anteriormente – serão expostas as duas diretrizes restantes para a elaboração de um método de gamificação.

#### 3.6.1

##### Aspectos da pesquisa e trabalho em ciência

Em primeiro lugar, quanto às *regras e consequências negociáveis* do trabalho científico, estas apontam para as seguintes oportunidades e desafios: suporte às regras metodológicas; auxílio na simulação de leis científicas; atenção a anomalias e irregularidades; clareza quanto a etapas e soluções permitidas; possibilidade de manipulação livre; construção de modelos; regras e demonstrações; comparação de sistemas modelados aos equivalentes reais; narrativas e guias visuais auxiliares; reconstrução da realidade; lógica e abstração; lugar criativo; similaridade e fidelidade ao mundo real; manipulação experimental; “blocos de armar”; flexibilidade e mudança; repetição; clareza dos objetivos (gerais e específicos) de pesquisa; raciocínio lógico; registro e interpretação de sistemas de signos; matematização; computação; “personalidade” dos equipamentos; realidades artificiais; centrais e instituições; comunidades científicas; processamento de

dados; geração de imagens; transparência das simulações; distanciamento crítico das simulações; caixas pretas; possibilidade, registro e recuperação de erros; riscos; imersão; “realidade” e tangibilidade das simulações; coleta de dados e experimentação; visualização de resultados e padrões.

Em segundo lugar, quanto, aos *resultados variáveis, quantificáveis e valorizados* do trabalho científico, destacam-se as seguintes oportunidades e desafios: solução de problemas; descrições da natureza; domínio sobre fenômenos; demonstrações; correções; exploração de novas técnicas; geração de melhores experimentos; flexibilidade e independência quanto a teorias; rigor e reprodutibilidade dos experimentos; medição de resultados; descrição de fenômenos, métricas e métodos; elaboração e reformulação de enunciados; avaliação pelos pares; produção de artigos; elevação do *status* dos enunciados; transformação de enunciados em fatos científicos e caixas pretas; abrangência e escopo dos enunciados; derivação de novos enunciados a partir de outros; testabilidade e simplicidade dos enunciados; competição entre enunciados; coerência; estabelecimento de enunciados como parte do domínio; publicações; financiamento; *ranking* acadêmico; velocidade e qualidade dos experimentos; métricas de produtividade; autoria; liderança; desenvolvimento da carreira; interesses de outras partes envolvidas.

Finalmente, em relação ao *esforço e apego do jogador*, destacam-se os seguintes pontos: teorias, cálculos e experimentos; lógica; intuição; construção de experiências; rigor metodológico; refutação; imaginação; prática; objetividade; atenção ao experimento; poder de seleção e síntese; neutralidade; capacidade de representação simbólica e estrutural; interpretação e reconhecimento de padrões; geração de artefatos; geração, interação, navegação e apresentação de visualizações; solução e identificação de problemas e oportunidades de pesquisa; formulação de hipóteses; busca e reconhecimento de padrões; domínio sobre paradigmas; criação de analogias e modelos; registro dos passos da pesquisa; obtenção, tratamento e interpretação de dados; compreensão quanto a equipamentos de pesquisa; habilidade manual; percepção sensorial; criação de repertórios de soluções; variação “cega” de procedimentos; experimentação livre; persuasão; elegância; posicionamento crítico à imersão e persuasão; transparência; capacidade de ler e escrever código; produção de sistemas transparentes; emulação do



comportamento do objeto de estudo; períodos de imersão, incubação e elaboração dos problemas; *insight* através do domínio sobre regras e critérios; internalização de regras; entendimento, registro e descrição de atividades; gerenciamento de tempo, trabalho, atenção e dedicação; transição entre extremos de prazer/desconforto, paixão/objetividade, racionalidade/imaginação, disciplina/brincadeira, orgulho e modéstia; construção, manutenção e relação com ambientes criativos; manipulação de modelos; colaboração com cientistas, equipes e instituições; inserção na cultura de instituições, campos e domínios de pesquisa; gerenciamento de recursos e estratégias de pesquisa; exploração de recursos disponíveis; geração de documentação e artigos; planejamento; formação de equipes; negociação de habilidades; colaboração multidisciplinar; diversão e estados de fluxo; afeição.

### 3.6.2

#### Questões quanto à aplicação de uma visão sistêmica

Após a identificação e discussão acerca das características lúdicas da prática científica, tornam-se evidentes os motivos pelos quais a classificação da ciência como jogo não é trivial. De fato, ao negar a afirmativa de que a ciência seria um jogo, Huizinga tocou em um ponto central desta dificuldade: as regras da ciência, ao contrário das dos jogos, não são imutáveis. Esta, porém, não seria necessariamente uma questão incontornável: vários jogos tiveram suas regras alteradas ao longo do tempo – tome-se o futebol como exemplo, com alterações quanto às leis do impedimento, critérios de desempate e sistemas de pontuação. Vários outros jogos – sobretudo digitais – permitem aos jogadores realizar ajustes e customizações em suas regras de funcionamento, além de possibilitar a inclusão de conteúdo personalizado.

As regras da ciência, no entanto, não são apenas mutáveis, mas também simbióticas em relação aos esforços dos cientistas e aos resultados obtidos. Essa dificuldade é evidente ao lembrarmos da definição de regras proposta por Salen e Zimmerman ('conjuntos de instruções claras e explícitas a respeito do funcionamento do funcionamento do jogo e das ações do jogador'). Em relação às *teorias* (regras que descrevem o funcionamento dos fenômenos), estas se tratam de não apenas regras aceitas pelas comunidades científicas, mas, também, aquelas em

construção pelo cientista, e cuja aceitação como *regra* no domínio científico é o *resultado* final do jogo, e não seu ponto de partida.

Por outro lado, em relação às *metodologias* (regras que descrevem as ações e passos percorridos pelos cientistas), estas não são – sobretudo nas descrições de Latour e Knorr Cetina – verdadeiramente estabelecidas antes da pesquisa, mas construídas, adaptadas e renegociadas ao longo deste processo. Os mesmos autores concordariam, então, que as regras só se tornam claras e explícitas posteriormente, na forma de publicação ou relato dos experimentos, ou seja, como instruções para sua reprodução – e mesmo para esta finalidade poderia se mostrar problemática, de acordo com Galison.

Há uma outra dificuldade envolvendo regras e resultados: enquanto jogos costumam permitir a validação de resultados de acordo com critérios internos às suas regras, as regras elaboradas pelos cientistas – e os resultados de suas pesquisas – devem passar pelo crivo da comunidade científica. Assim, do ponto de vista do uso de software científico, embora a quantificação de resultados seja facilitada por recursos computacionais, a validação dos resultados deve ser feita posteriormente, pelo próprio cientista e sua comunidade.

Tais questões não inviabilizam a visão da prática científica como um jogo – por mais peculiar que este jogo seja. No entanto, assim como as questões envolvendo o desenvolvimento do software científico, tratam-se de desafios adicionais em um processo de gamificação, visto que regras, esforços e resultados se confundem, e a validação final dos resultados não pode ser obtida automaticamente ou rapidamente.

### **3.6.3 Diretrizes para a elaboração do método: da ciência e suas dinâmicas**

No capítulo anterior, foram listadas quatro diretrizes para o método de gamificação proposto – (1) ser adaptável ao *processo de design de interfaces* e usabilidade para esse tipo de software; (2) considerar *características peculiares* de uso e desenvolvimento; (3) atentar para a *natureza sistêmica* dos jogos; (4) considerar os *aspectos lúdicos da pesquisa* e trabalho científico - tendo sido aprofundadas as diretrizes (1) e (2).

Ao longo do presente capítulo, buscou-se desenvolver as diretrizes restantes: (3) atentar para a *natureza sistêmica* dos jogos e (4) considerar os aspectos *lúdicos da pesquisa e trabalho científico*.

No que se refere a uma visão sistêmica da ciência enquanto jogo, é preciso levar em consideração que serão os próprios jogadores que estabelecerão suas regras, objetivos, e – ao menos em caráter provisório – as métricas de sucesso de seus experimentos. Mais ainda, é necessário levar em conta que estas regras, objetivos e métricas serão provavelmente modificadas ao longo da pesquisa em que estão inseridas, de acordo com as etapas da pesquisa em que ocorrem, ou por revisões teóricas, de cálculo e experimentais. Desta forma, *a relação entre esforço, regras e resultados deve ser determinada pelos cientistas, levando em consideração suas necessidades e, também, a fase da pesquisa ou atividade em que se empenham*.

Além disto, a exemplo das características do uso e desenvolvimento de software científico, as *características lúdicas da ciência devem ser tornadas evidentes durante processos de design*, informando projetos em seus riscos e oportunidades. Estas características, tal como organizadas ao longo do capítulo, dizem respeito às regras, esforços, resultados, valorização, recompensas e distanciamento da realidade na prática científica.

#### **3.6.4** **Jogo, ciência e software**

É possível encontrar equivalentes entre os aspectos da prática científica – destacados neste capítulo como manifestações do jogo na ciência – e os aspectos do uso e desenvolvimento de software científico identificados no capítulo anterior. A forte relação entre os dois conjuntos, apresentados em caráter não exaustivo na Tabela 2, sugere que o trabalho em software científico tem grande potencial enquanto atividade-alvo para gamificação baseada em uma visão sistêmica dos elementos lúdicos da ciência. Neste caso, os próprios aspectos de uso e desenvolvimento são, dentro do modelo proposto pela tese, componentes do caráter lúdico do trabalho em software científico.

Elemento lúdico	Manifestações na ciência	Manifestações no software científico
<b>Regras</b>	Método científico	Tarefas; reprodutibilidade; procedimentos.
	Teorias	Modelos; requisitos emergentes; bases de conhecimento.
<b>Consequências negociáveis</b>	Matematização	Modelos; linguagens de programação;
	Laboratórios e simulação	Ambientes virtuais; representação
<b>Resultados variáveis e quantificáveis</b>	Medições e resultados	Análise de resultados; visualização de dados
	Enunciados científicos	Visualização de dados; suporte à publicação
<b>Valorização do resultado</b>	Publicações	Suporte à publicação
	Desenvolvimento profissional	Graus de especialização do usuário
<b>Esforço</b>	Conhecimento sobre domínio	Bases de conhecimento; bases de dados;
	Domínio sobre sistemas simbólicos	Linguagens de programação; DSL; programação visual
	Habilidade sensorial e motora	Visualização; prevenção de erros
<b>Apego</b>	Motivações extrínsecas	Publicação
	Motivações intrínsecas	Fluxo; imersão

Tabela 2 - Amostra de manifestações dos elementos do jogo na ciência e paralelos no software científico

Além disso, pode-se transpor as características da ciência e do software científico à visão sistêmica da definição de jogos de Juul. Através desta transposição construímos um modelo híbrido, ilustrado na Figura 28. As equivalências sugeridas na Tabela 2, a seguir, bem como o sistema ilustrado, irão orientar a construção do método de gamificação proposto em uma das experiências relatadas no próximo capítulo.

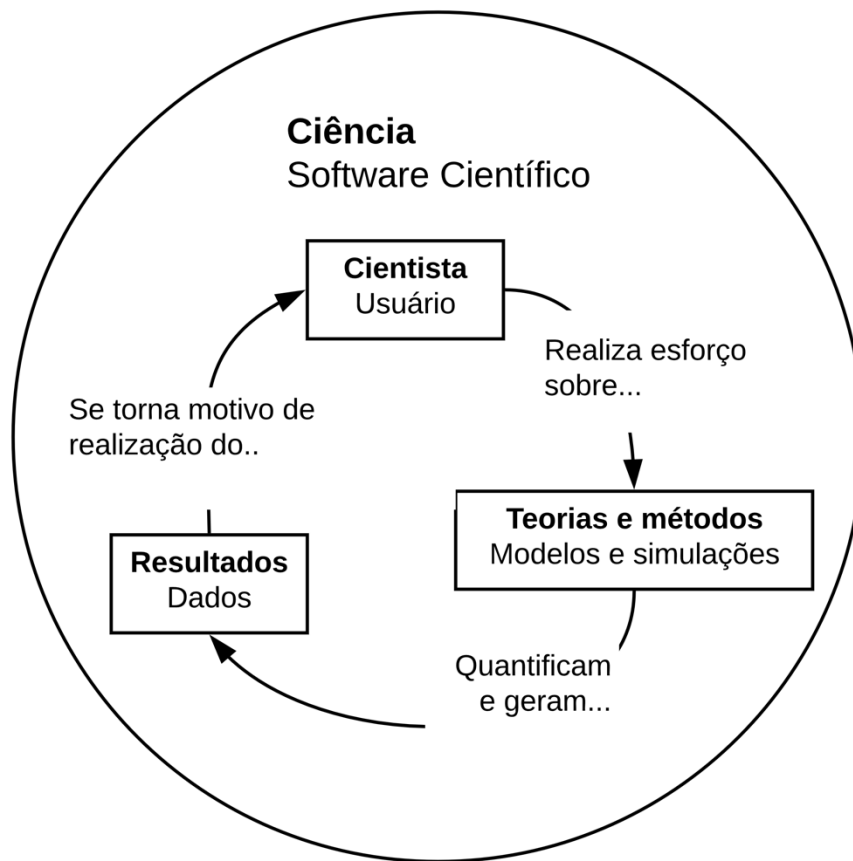


Figura 28 - Modelo híbrido jogo-ciência-software.

## 4

### Experiências em design de gamificação para software científico

Ao longo deste capítulo serão relatadas duas experiências em design colaborativo de interfaces para software científico. A primeira, um experimento piloto baseado na metodologia da pesquisa-ação, aborda a gamificação do ponto de vista da adoção de padrões de interatividade inspiradas em video games, e foi realizada ao longo da revisão de literatura, precedendo a pesquisa bibliográfica para fundamentação teórica. A segunda, realizada de acordo com as diretrizes desenvolvidas ao longo dos capítulos anteriores, se apoia nas características de uso e desenvolvimento do software científico, aspectos lúdicos da prática científica e, também, em uma visão sistêmica informada pelos elementos definidores do jogo para propor um método apropriado ao co-design de interfaces para aquele tipo de software. Buscou-se, através destas experiências, atender àquele que é o terceiro dos objetivos específicos desta tese – desenvolver métodos para o design de interfaces de software científico – e responder à questão norteadora que havia permanecido em aberto: como a dinâmica dos jogos pode ser transposta ao software científico através do design?

Este capítulo se divide em três seções. As duas primeiras, intituladas *Migrando padrões de interatividade: dos video games para a simulação integrada em engenharia de óleo e gás* e *Co-design de interfaces baseado em jogos: casos em matemática, física e engenharia*, descrevem, separadamente, as experiências em design mencionadas acima, suas metodologias, procedimentos e resultados. A terceira, *Análise comparativa*, comenta as diferenças entre as abordagens e seus resultados.

#### 4.1

#### **Migrando padrões de interatividade: dos video games para a simulação integrada em engenharia de óleo e gás**

Uma possível abordagem à gamificação, conforme identificado pela revisão de literatura, seria a transposição de padrões de interatividade e referências estéticas dos meios dos jogos digitais para produtos gamificados. Tal abordagem se apoia menos na reestruturação de atividades-alvo em formatos semelhantes a jogos, e mais na adoção de estéticas, formas de visualização e interfaces típicas daquele meio. A partir desta perspectiva, desenhou-se – antes da pesquisa bibliográfica e da concepção de um método – uma experiência piloto que permitisse explorar o potencial dos jogos eletrônicos como fonte de inspiração para software científico. Esta seção relata aquele que foi o primeiro estudo de campo realizado para tese, desenvolvido entre os meses agosto e dezembro de 2015 no Instituto Tecgraf / PUC-Rio. Naquela ocasião, quatro desenvolvedores e um designer compunham a equipe de desenvolvimento do software *SiVIEP*, destinado à visualização de modelos tridimensionais dos processos de exploração e produção de projetos de óleo e gás, bem como a visualização de resultados de simulação daqueles projetos (Raposo et al., 2016). Tal software – desenvolvido para a empresa de energia Petrobras – havia sido projetado para uso em computadores desktop e, também, ambientes de realidade virtual imersivos, e teve seu desenvolvimento iniciado em 2010. A pesquisa envolveu não apenas o autor e a equipe de desenvolvimento, mas também os dois *Product Owners* (POs) do software – posição equivalente a gerente de produto na terminologia da metodologia de desenvolvimento de software *Scrum*, na qual se baseava o desenvolvimento no produto em questão. POs, no entanto, costumam ter, mais do que gerentes de produto, poder de decisão sobre aspectos de desenvolvimento – ainda que, no caso específico deste software, equipe e POs não trabalhassem juntos na mesma locação. Esta pesquisa foi desenvolvida durante o curso de disciplinas em Tópicos de Interação Humano-Computador, e contou com a colaboração dos participantes e, também, com a supervisão do Prof. Alberto Barbosa Raposo, gerente do Instituto Tecgraf e professor daquelas disciplinas no Departamento de Ciência da Computação na PUC-Rio.

Ao longo das próximas subseções, será relatada a pesquisa realizada em torno do *SiVIEP*. Primeiro, a seção *Metodologia* descreve a dinâmica e procedimentos da pesquisa, baseada na metodologia da pesquisa ação. Em seguida,

*Desenvolvimento* relata o desenrolar da pesquisa ao longo de suas etapas. *Discussão* aponta para as lições aprendidas com a experiência e as contribuições da pesquisa ao campo de estudo. Os resultados são, então, sintetizados na seção *Conclusão*.

#### 4.1.1 Metodologia

A metodologia da pesquisa-ação foi escolhida como base deste estudo por privilegiar uma experiência empírica em primeira mão. Trata-se de uma metodologia frequentemente aplicada em processos de inovação e melhorias em práticas profissionais. Quando apoiada por fontes complementares como pesquisas bibliográficas e referenciais teóricos, a pesquisa-ação é capaz de gerar conhecimento através da solução de questões de cunho prático, em contextos locais. No caso deste estudo, buscou-se identificar e atender questões de usabilidade do *SiVIEP*, obtendo, ao longo do processo, conhecimento sobre desafios e oportunidades na gamificação do software científico. Projetada sobre a descrição e esquematização de David Tripp sobre o ciclo de investigação-ação (Tripp, 2005, p.446), a Figura 29 ilustra uma visão da pesquisa realizada.

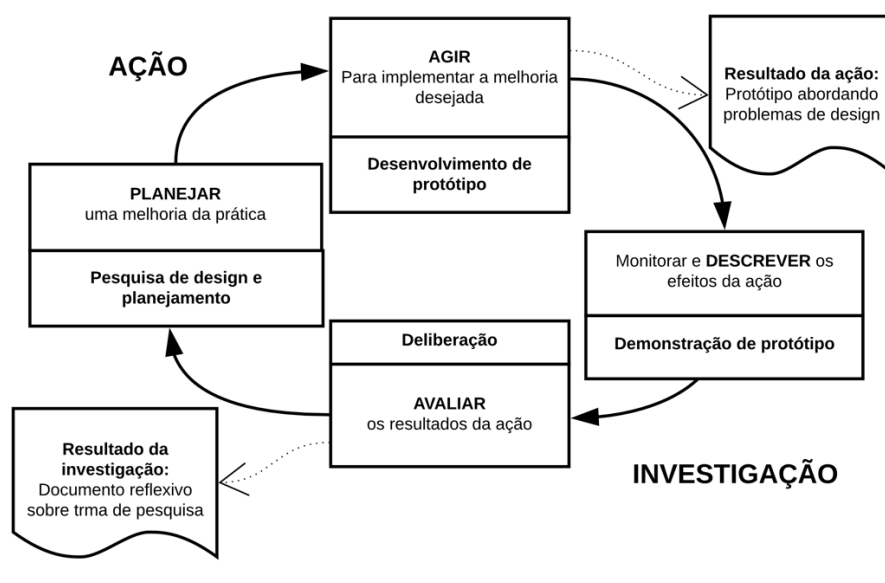


Figura 29 - Visão geral da pesquisa (Queiroz et al 2016)

A pesquisa foi realizada num período de aproximadamente quatro meses, durante o qual foram realizadas nove reuniões entre os participantes – aproximadamente uma



reunião a cada duas semanas. Durante as reuniões, problemas foram identificados e possíveis soluções, debatidas. As reuniões, que incluíram sessões de *brainstorming*, também foram utilizadas para a avaliação das ideias propostas, debates sobre práticas no design de jogos e planejamento de etapas seguintes. Durante o período entre reuniões, o autor – na condição de pesquisador – desenvolveu protótipos para a proposta de funcionalidades atendendo os problemas identificados. Em uma ocasião, foi realizada uma entrevista semiestruturada com os POs. Após cada encontro, o conteúdo das discussões foi sintetizado e distribuído aos participantes no formato de atas baseadas em anotações e gravações produzidas durante as reuniões. As atas, tais como os termos de consentimento assinado pelos participantes, estão incluídas nos Anexos, e incluem material apresentado ou produzido durante as reuniões, tais como rascunhos e capturas de tela, em formato digital.

#### **4.1.2 Desenvolvimento**

Relatadas a seguir estão as diversas fases da pesquisa, da definição do problema ao encerramento da investigação.

##### **4.1.2.1 Primeira fase: definição do problema**

Os três encontros iniciais se dedicaram à identificação de possíveis questões na usabilidade do software que pudessem ser solucionadas ou mitigadas a partir de soluções de design oriundas de jogos – particularmente os eletrônicos.

*Primeira reunião - visão geral do software e percepção de deficiências.* O primeiro encontro entre pesquisador e desenvolvedores foi voltado para a apresentação de uma visão geral do *SiVIEP*, questões de usabilidade e desenvolvimento. O software foi definido como um visualizador de simulações integradas de projetos de óleo e gás, capaz de incorporar dados de fontes diversas (plantas de processamento, unidades de produção, reservatórios, etc.) em uma única representação tridimensional de um campo de produção (Figura 30).

Além de possibilitar a navegação em primeira pessoa por aquele ambiente virtual, o software conta com funcionalidades de apoio à visualização de dados.

Tais funcionalidades incluem a *Timeline* – que permite a visualização da evolução dos projetos de óleo e gás ao longo do tempo, e o *Graph* – que exibe uma visão esquemática dos equipamentos interconectados. Ambas as funcionalidades estão ilustradas na Figura 31.

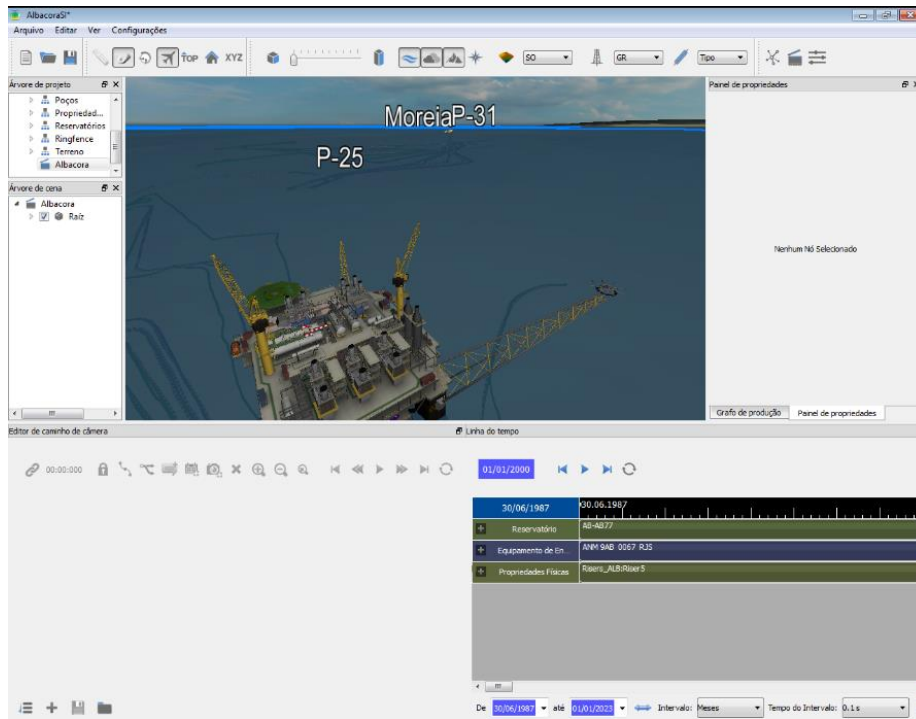


Figura 30 - Captura de tela do SiVIEP. (Queiroz et al., 2016):

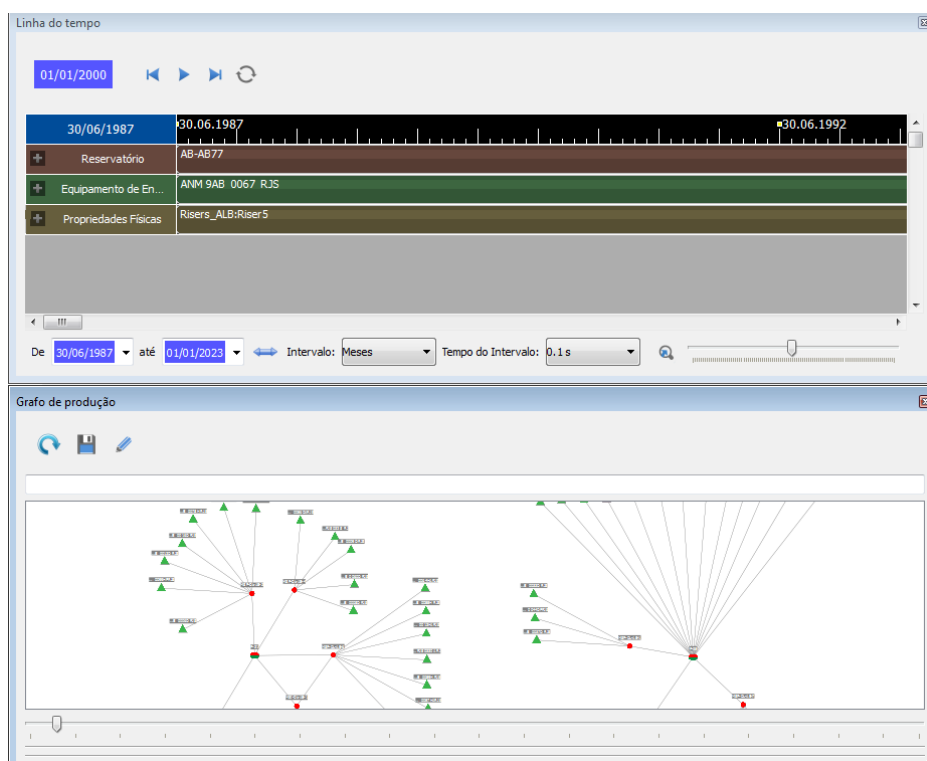


Figura 31 - Timeline (acima) e Graph (abaixo) (Queiroz et al., 2016)

Até o momento da pesquisa, a principal aplicação do software havia sido seu uso na exibição de campos de petróleo em ambientes de realidade virtual. No entanto, percebia-se que o modo de operação do software em realidade virtual não possuía algumas das funcionalidades presentes na versão para *desktop* – particularmente as janelas da *Timeline* e do *Graph* que possibilitam o controle do tempo e das visões esquemáticas dos ambientes representados. Além disso, os desenvolvedores expressaram o desejo de unir as interfaces para *desktop* e realidade virtual, evitando assim o trabalho extra necessário para a implementação das duas versões em separado. Desta forma, decidiu-se buscar soluções de design que possibilitassem a implementação daquelas funcionalidades em modo de realidade virtual.

Ao final da primeira reunião, foi decidido em comum acordo que as fontes de referência iniciais para a pesquisa em interfaces visando as melhorias pretendidas seriam jogos de simulação e do gênero *tycoon* – ainda que experiências passadas, tal como relatadas pelos desenvolvedores, sugerissem que os *Product Owners* poderiam ser reticentes quanto a ideias inspiradas em *video games*.

*Segunda reunião – indefinição quanto a escopo e precedentes em interfaces inspiradas em jogos.* Durante a segunda reunião entre pesquisador e

desenvolvedores, mencionou-se a possibilidade de redirecionamento do SiVIEP, num futuro próximo: dificuldades em obtenção de requisitos e aquisição de dados tornavam o foco em simulação integrada difícil de ser mantido. A reunião contou com uma sessão de *brainstorm* para o *redesign* das funções de Timeline e Graph. Além disso, a equipe de desenvolvimento mostrou ao pesquisador antigas propostas de *redesign* inspiradas em video games que haviam sido descartadas pelos POs, sob a justificativa de que (1) havia inconsistências entre a iconografia proposta e padrões da indústria e (2) não havia ênfase suficiente na tridimensionalidade da interface. Foi sugerido ao pesquisador que se entrevistasse os POs.

*Terceira reunião – Entrevista com Product Owners.* Em entrevista ao pesquisador, os POs enfatizaram a dificuldade na obtenção de dados e casos de uso concretos para o software. Na ocasião, os entrevistados descreveram simulação integrada para óleo e gás como um campo de desenvolvimento recente e sem padrões estabelecidos – o que tornava o próprio desenvolvimento do SiVIEP um projeto de pesquisa. De fato, este software era considerado, pelo centro de pesquisas da companhia, um projeto voltado para perspectivas futuras, e não para aplicação imediata em necessidades prementes.

Para os entrevistados, o verdadeiro valor do SiVIEP estaria em tornar possível, por parte de profissionais da área, a visualização tridimensional de informações e dados que só teriam visto, até o momento, em duas dimensões. Eles declararam, ainda, o interesse em interromper o desenvolvimento de funcionalidades de realidade virtual até que a versão para *desktop* estivesse completa e disseminada. Porém, reconheciam o potencial da realidade virtual como forma de engajar usuários e como ambiente para reuniões multidisciplinares. A respeito do papel da interface do software em uma experiência imersiva, os entrevistados demonstraram interesse em uma interface mais intuitiva, particularmente para posicionamento de objetos e construção de cenas em ambientes virtuais. O sistema de navegação – que havia sido melhorado pouco tempo atrás – poderia ser aprimorado, possibilitando a navegação no espaço tridimensional a partir do mapa bidimensional disponível na interface – uma funcionalidade que, acreditam, seria intuitiva para profissionais de engenharia.

Questões técnicas a respeito da experiência em realidade virtual também foram abordadas: observou-se que o controle do software em ambiente virtual fazia uso do controlador *Wiimote* – *joystick* de vídeo game produzido pela Nintendo com

funcionalidades para a detecção de movimentos e sensor de direção. Porém, tal controlador era utilizado, no SiViEP, *sem* auxílio da barra de sensor que acompanha aquele aparelho, impossibilitando a detecção da direção para qual o controlador está apontado. Dessa forma, preferia-se uso de menus contextuais, de preferência circulares, através dos quais opções pudessem ser selecionadas através das setas direcionais do controle.

Embora um dos POs tenha se mostrado familiarizado com jogos, particularmente dos gêneros simulação e *tycoon*, o outro declarou que não era um jogador ativo, nem familiarizado com o meio – embora tenha se mostrado bem informado a respeito do potencial dos jogos para educação e ciência, e disposto a aprender mais sobre o assunto.

*Encerramento da primeira fase.* Ao final desta primeira etapa, a produção do primeiro protótipo de baixa fidelidade foi iniciada, baseada nos requisitos coletados durante as reuniões iniciais: (1) *redesign* das funcionalidades da *Timeline* e *Graph* para experiência comum a desktop e realidade virtual; (b) Sistema de controle claro e intuitivo para públicos familiarizados e, também, não-familiarizados com video games.

#### 4.1.2.2

##### **Segunda fase: produção e recepção a protótipos iniciais**

A pesquisa para a criação de um protótipo inicial se deu em torno de vários conceitos diferentes. Em primeiro lugar, o uso do controlador *Wimote* e a tradição da Nintendo em desenvolver produtos para públicos de jogadores casuais nos levaram a pesquisar seus produtos (especialmente aqueles produzidos para o console Wii). Em segundo lugar, a proximidade do uso proposto para o SiViEP com jogos de simulação nos levou a investigar títulos como *Sim City* e *Cities in Motion*, principalmente para pesquisa a navegação de cenários tridimensionais baseada em pequenos mapas bidimensionais. Finalmente, buscando maior facilidade de uso por públicos não familiarizados com jogos, pesquisamos interfaces de *Smart TVs*. Diversos em suas naturezas, tais conceitos foram combinados em um protótipo no qual um painel, equipado com menu lateral, apresentava o *Graph* projetado como um mini mapa a partir do qual usuários poderiam navegar pelo ambiente virtual.

Para maior clareza, instruções quanto ao funcionamento do controle seriam projetadas na tela (Figura 32).



Figura 32 - Protótipo de baixa fidelidade (Queiroz et al., 2016)

O protótipo inicial incluía, também a funcionalidade da *Timeline*. Protótipos de baixa fidelidade – imagens estáticas ilustrando diferentes etapas do uso do programa – foram produzidos com o software *Adobe Illustrator* sobre capturas de telas do SiVIEP, e apresentadas em computadores *desktop*.

*Quarta reunião – protótipo de baixa fidelidade.* A quarta reunião contou com a demonstração do protótipo para um membro da equipe de desenvolvimento, o qual fez uma série de observações levadas em consideração na versão subsequente. A segunda versão teria o menu lateral removido, e seu sistema de controle retrabalhado, buscando uma semelhança ainda maior com interfaces de *Smart TV*.

*Pesquisa para novo protótipo de baixa fidelidade.* Para a segunda iteração do protótipo buscou-se pesquisar outros aspectos do uso do software – além de atender aos pontos levantados durante a deliberação anterior. Foi vislumbrado um novo modo de acesso rápido ao controle do tempo da simulação, tornando sua ativação mais fácil durante a navegação espacial. É importante enfatizar que tal funcionalidade foi inspirada pelo sistema de controle do jogo de plataforma *Braid*, de Jonathan Blow, no qual jogadores podem pausar, retroceder e avançar o

desenrolar das ações no jogo. A nova funcionalidade não foi projetada para substituir a *Timeline* (Figura 33) – que incluiria diversas outras funcionalidades além do controle de reprodução – mas para complementá-la.

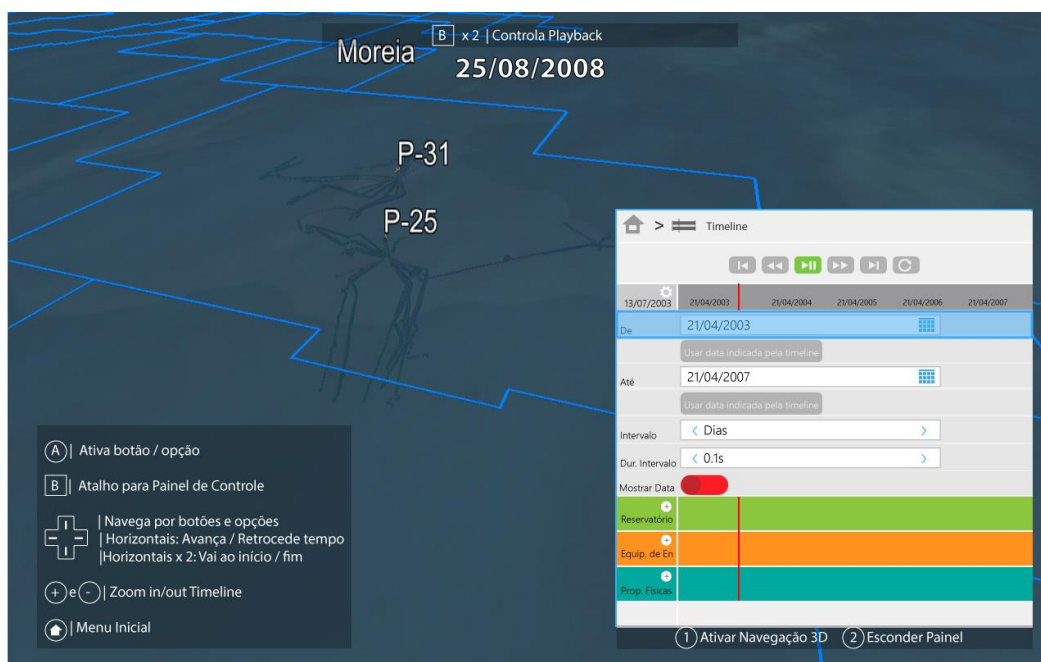


Figura 33 - Timeline e instruções (Queiroz et al., 2016)

*Quinta reunião – reações ao novo protótipo e mudanças no escopo.* Durante a quinta reunião, uma segunda versão do protótipo foi apresentada. *Product Owners*, presentes à reunião, expressaram a intenção de reposicionar o SiVIEP como ferramenta de software para construção livre de cenários, através do qual experimentar novas configurações para projetos de óleo e gás. Tal mudança exigiria ênfase em ferramentas para construção, incluindo funcionalidades para posicionamento de objetos em cena, desenho de conexões entre objetos, criação e carregamento de terrenos, etc. Quanto ao protótipo demonstrado, foram feitos comentários positivos ao sistema de controle de reprodução. Além disso, sugeriu-se a extensão do uso de menus circulares a toda a interface, e não apenas restrito aos painéis principal e *Graph*, conforme projetado naquele protótipo.

*Encerramento da segunda fase.* Ao fim da segunda etapa, concordou-se em pesquisar elementos de design de jogos que pudessem informar o design das ferramentas de construção que seriam integradas ao SiVIEP. Além disso, foi acertado que os próximos protótipos passariam a ser de alta fidelidade, gerados a partir de software voltado ao desenvolvimento de jogos, e incluiriam recursos avançados de interação.

#### 4.1.2.3

##### **Terceira fase: pesquisa a jogos e protótipos de alta fidelidade**

A nova direção a ser tomada pelo software exigiu a ênfase em um novo conjunto de funcionalidades – e em maneiras de apresentá-las.

*Sexta reunião – mudança de foco.* O sexto encontro reuniu desenvolvedores, designer e pesquisador, e foi dedicado a (a) rever os métodos e o desenvolvimento da pesquisa até o momento, (b) apresentar e gerar ideias para a interface do SiVIEP e (c) definir os requisitos e escopo para o próximo protótipo. A revisão da pesquisa incluiu uma recapitulação das fases anteriores, além de reforçar a ideia de se buscar, em jogos, soluções para interatividade e usabilidade do software. Capturas de telas dos video games pesquisados até aquele momento (*Cities in Motion 2*, *Democracy 3*, *Mini Metro*, *Another Code*, *Sim City Creator* e *Braid*) foram exibidas ao lado da descrição das funcionalidades que eles inspiraram – com destaque para *Braid* e *Cities in Motion 2*, que inspiraram o controle de reprodução os controles do *Graph*, respectivamente. (Figura 34). Foram discutidas ideias para novas funcionalidades – como uma ferramenta para o desenho de tubulações entre objetos, possivelmente incluindo a exibição da distância percorrida, comprimento da tubulação, e quantidade de material necessário para sua construção. Ainda que promissora, tal funcionalidade foi considerada para possível desenvolvimento somente em estágios mais avançados de desenvolvimento.





Figura 34 - Cities in Motion 2 (acima) e Braid (abaixo)

Quanto ao escopo do protótipo, ficou estabelecido que seu objetivo seria permitir a usuários que montassem um cenário semelhante ao cenário pré-configurado incluído nas demonstrações do SiVIEP. O protótipo deveria, ainda, permitir aos usuários carregar modelos tridimensionais em locais pré-definidos – mas também ilustrar como poderiam ser posicionados manualmente. Foi sugerido que um protótipo interativo em alta fidelidade seria mais eficaz em contextualizar as novas funcionalidades e demonstrar melhorias no uso do software. Finalmente, foi decidido que o protótipo faria uso de menus circulares – uma forma de interação presente em jogos voltados ao público casual (como no caso de *Viva Piñata*) e, também, ao público *hardcore* (como *Mass Effect*), o que indicaria ser apropriado a usuários de diversos graus de familiaridade com o meio (Figura 35).



Figura 35: Viva Piñata (esquerda) e Mass Effect (direita)

*Sétima reunião – refinando o escopo.* O sétimo encontro reuniu os mesmos participantes da reunião anterior, e foi organizada para definição do escopo e requisitos do protótipo. Foi decidido que usuários deveriam escolher, a partir de uma tela inicial, qual região geográfica seria carregada na cena. Referências para a interface incluíram o menu de seleção do jogo *Sim City 4* e, também, um conceito para *redesign* do SiVIEP proposto em anos anteriores (Figura 36).

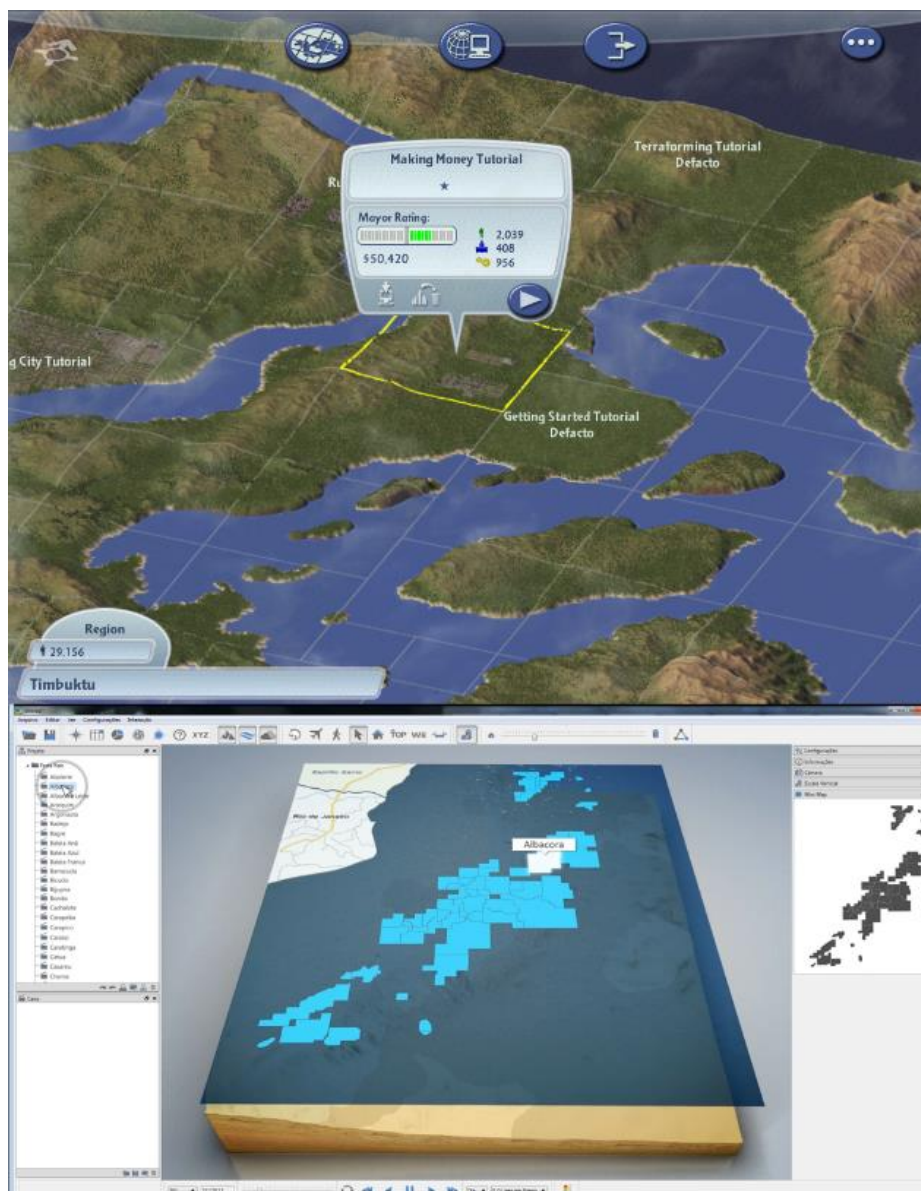


Figura 36 - Sim City 4 (acima) e proposta (abaixo) (Queiroz et al. 2016)

*Produção do protótipo de alta fidelidade.* Decidiu-se que o protótipo de alta fidelidade, interativo, seria produzido com auxílio da ferramenta de produção de jogos *Unity* e do plug-in *Sloubi Tools' Circular Menu*. A experiência do uso do protótipo incluiria os estágios iniciais do trabalho no SiVIEP: (1) tela inicial para seleção dos limites geográficos da nova cena; (2) ambiente virtual incluindo terreno da locação geográfica escolhida; (3) ativação de janelas de diálogo para importação de objetos através do menu circular; (4) uso da janela de diálogo para importação; (5) exibição dos objetos importados no ambiente virtual. Além disso,



funcionalidades deveriam ser projetadas para uso em *desktop* e ambientes de realidade virtual.

A tela inicial do protótipo pode ser vista na Figura 37, e apresenta uma caixa de diálogo para confirmação da seleção do usuário.

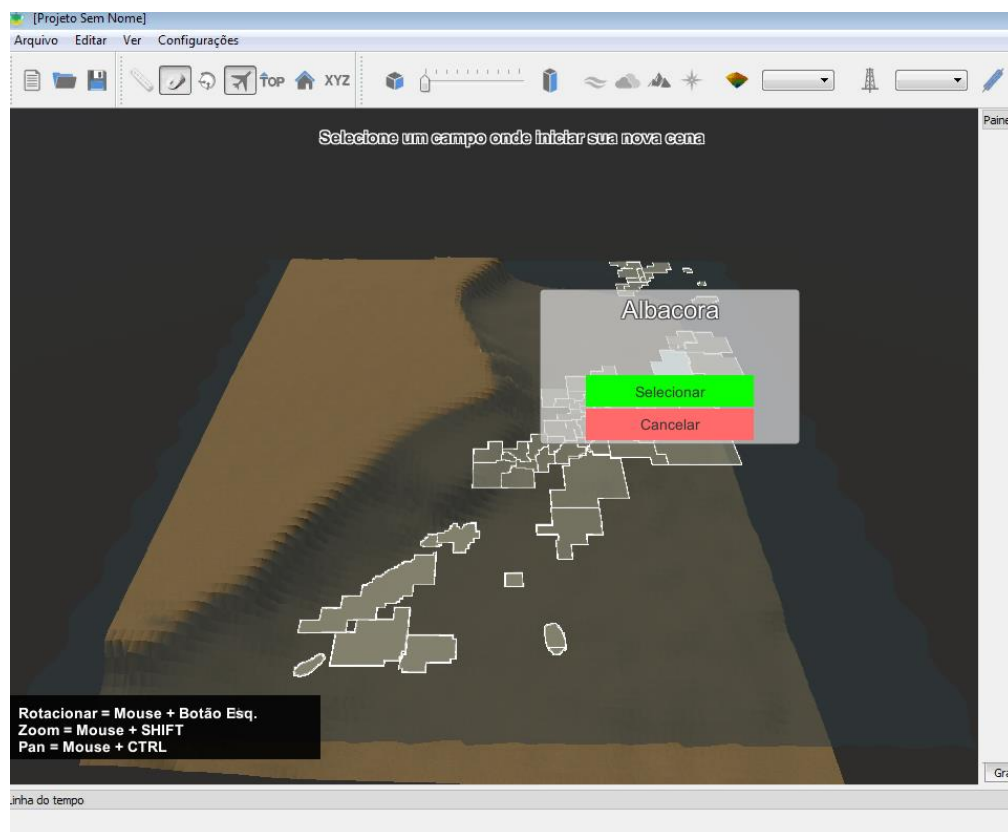


Figura 37 - Tela inicial do protótipo de alta fidelidade. (Queiroz et al. 2016)

Após confirmação por parte do usuário, o terreno inicial seria substituído por outro, representando a localização geográfica escolhida. A partir daquele cenário, seria possível ativar um menu circular para acesso aos comandos (Figura 38). No protótipo, as únicas opções de comando seriam aquelas associadas com posicionamento de objetos.

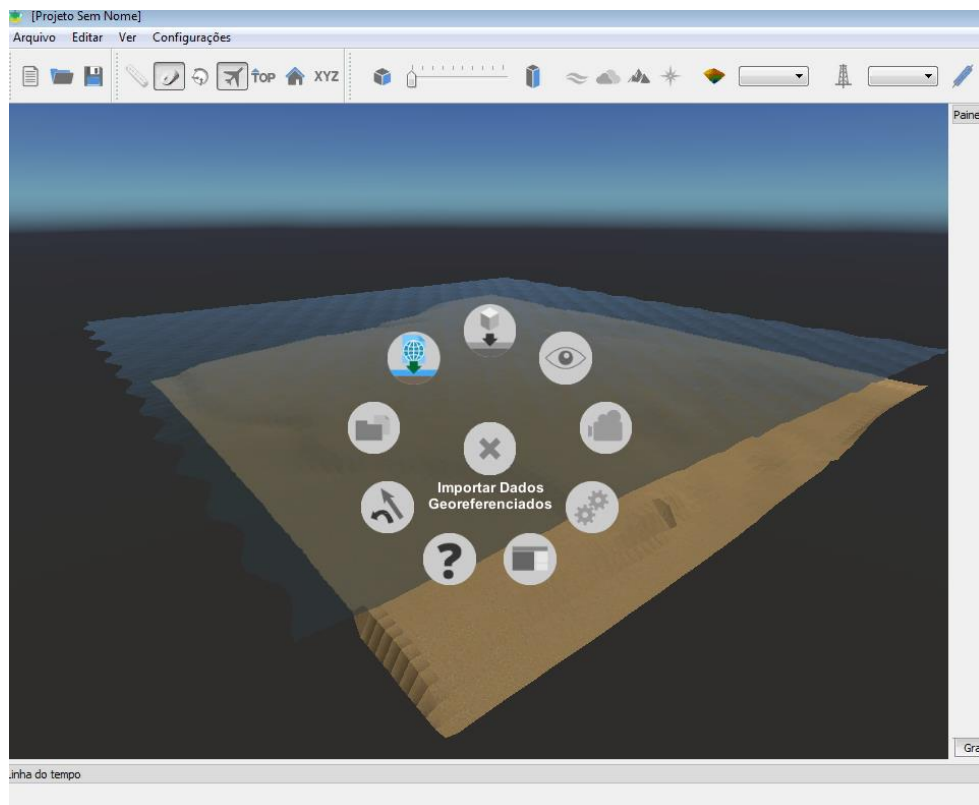


Figura 38 - Menu circular ativado (Queiroz et al. 2016)

Durante a produção do protótipo, foram estabelecidas duas diferentes opções para importação de objetos: A primeira opção seria dedicada a objetos contendo dados sobre sua localização geográfica, e ativaria uma janela de diálogo para seleção de objetos entre diversas categorias (Figura 39). A segunda opção, para objetos sem informações de geolocalização, revelaria um novo menu circular, a partir do qual seria possível abrir janelas específicas de acordo com o tipo de objeto selecionado (Figura 40).

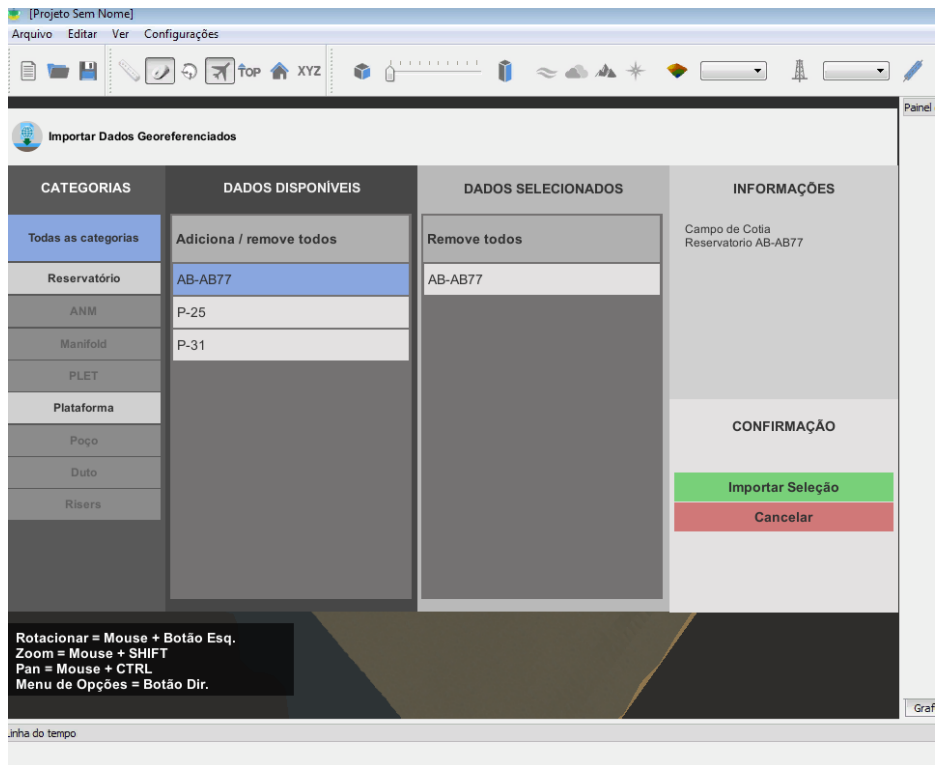


Figura 39 - Importação de objetos georreferenciados (Queiroz et al. 2016)

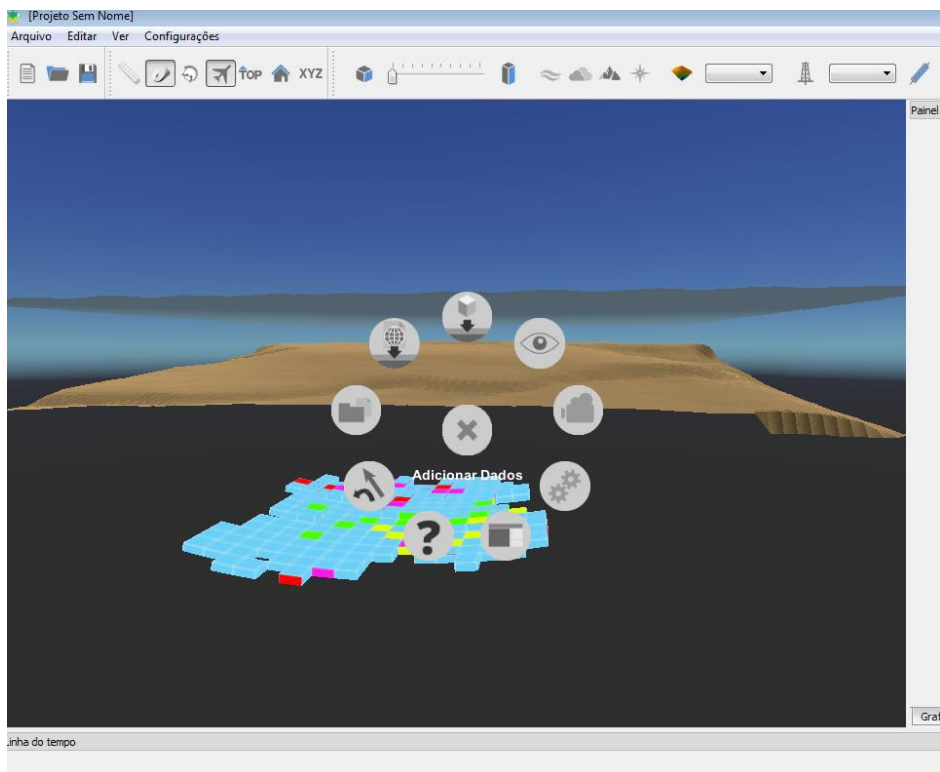


Figura 40 - Menu circular para carregamento de objetos (Queiroz et al., 2016)

No que se refere ao uso de video games como referência, deve-se acrescentar que a janela de diálogo para seleção de objetos exibida na Figura 39 foi influenciada pelo menu de seleção de armas do jogo *The Elder Scrolls V: Skyrim* (Figura 41).



Figura 41 - The Elder Scrolls V: Skyrim – menu de seleção de armas

Após todas as funcionalidades planejadas terem sido incorporadas ao protótipo, decidiu-se implementar a ferramenta de posicionamento de objetos. Tratava-se, no entanto, de uma funcionalidade particularmente desafiadora, pois exigia (a) controle intuitivo para modos *desktop* e em realidade virtual e (b) alta precisão para entrada de informações de latitude e longitude (cada um composto de sete dígitos, cada). Novamente, jogos de simulação foram tomados como referência: jogos como *Sim City Build It* e *Farmville* (Figura 42) contam com eficazes ferramentas de posicionamento que respondem a diversos modos de entrada (mouse e teclado, controles direcionais, e até mesmo telas sensíveis ao toque), além de serem amigáveis ao público casual.



Figura 42 - Sim City Build It (esquerda) e Farmville (direita)

Entretanto, apesar da facilidade de uso, as ferramentas de posicionamento pesquisadas não contavam com a precisão exigida por software científico de modo geral, e pelo SiVIEP em particular. Para proporcionar um posicionamento preciso dos objetos, duas soluções foram incorporadas: (1) entrada numérica (via teclado) para os valores de latitude e longitude e (2) translação do objeto em ordem de grandeza proporcional à distância do objeto. A segunda solução poderia ser descrita da seguinte maneira: ao se utilizar as setas direcionais para reposicionar o objeto, o deslocamento do mesmo seria inversamente proporcional à distância entre o objeto e o ponto de vista do usuário. Neste caso, aproximar a câmera do objeto resulta em deslocamentos menores e posicionamentos mais precisos. A Figura 43 ilustra esta funcionalidade: em 'A', o uso das setas direcionais causaria a translação do objeto em 1.0 unidade. Já em 'B', a proximidade da câmera indicaria o desejo do usuário em obter um controle mais preciso, alterando o valor de deslocamento da translação para apenas 0.01 unidades a cada vez que a seta direcional fosse acionada.

É interessante observar que esta solução foi inspirada em outra funcionalidade do SiVIEP: um sistema de navegação multi-escala que ajusta a velocidade da câmera dependendo da proximidade de objetos complexos (Taunay et al., 2015).



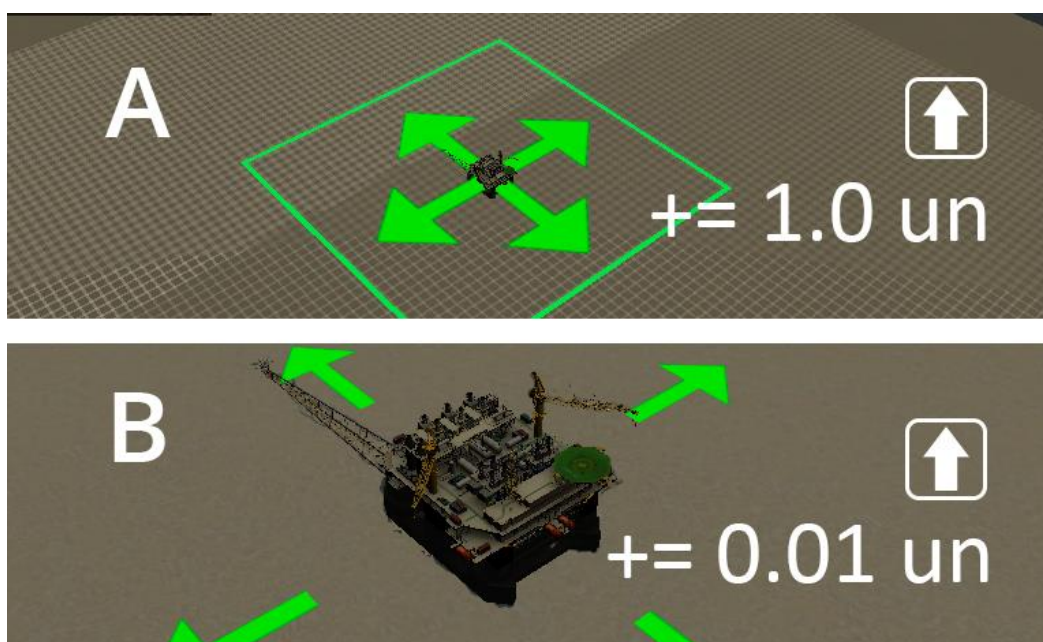


Figura 43 - Translação adaptável de objetos (Queiroz et al., 2016)

Além da prototipagem das funcionalidades escolhidas, todos os recursos existentes no SiVIEP foram mapeados para eventual implementação no sistema de menu circular. A Figura 44 mostra este mapeamento, destacando em um círculo as funcionalidades prototipadas.

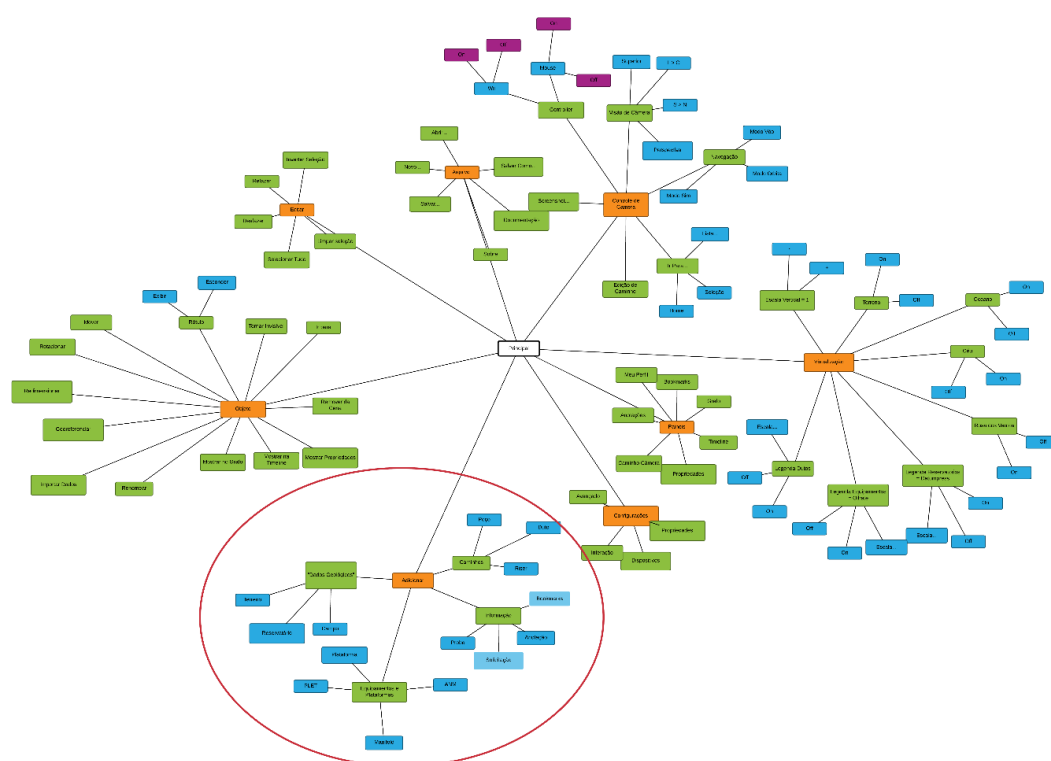


Figura 44 - Mapa de funcionalidades do SiVIEP (Queiroz et al., 2016)

#### 4.1.2.4

#### Encerramento da investigação e reação ao protótipo

O protótipo de alta fidelidade foi apresentado à equipe de desenvolvimento durante a oitava reunião. A reação à interface proposta foi positiva, e decidiu-se apresentá-la aos *Product Owners*. Quatro semanas depois, durante a nona reunião, as ideias contidas no protótipo foram bem recebidas. Porém, a implementação das funcionalidades apresentadas foi descartada por um motivo técnico: o desenvolvimento do menu circular através das ferramentas de produção do software exigiria um investimento de tempo e recursos humanos visto como excessivo – especialmente ao serem considerados os requisitos combinados do desenvolvimento para *desktop* e realidade virtual. A investigação foi formalmente interrompida, embora todas as partes tenham permanecido disponíveis para debate e conselhos a respeito dos temas pesquisados. Ferramentas para a construção de cenários no SiVIEP foram implementadas meses depois, apenas para versão *desktop*, em formato de *wizard* (série de janelas de diálogo organizada em etapas sequenciais) – uma funcionalidade de implementação, facilitada pelo *framework* de desenvolvimento de interfaces gráficas utilizado, ao contrário dos menus circulares, cuja implementação exigiria maior esforço de programação.

A exploração da interatividade inspirada em video games foi considerada, de modo geral, positiva, gerando um ganho de conhecimento no potencial de uso de elementos oriundos de jogos no software científico. De igual importância foi o ganho na conscientização quanto a questões na transição entre a *fase de design* e a *fase de implementação*. Este conhecimento gerado, resultado positivo desta investigação, é discutido mais extensivamente nas próximas subseções.

.

#### 4.1.3

#### Discussão

Apesar da ausência de implementação das funcionalidades gamificadas no software, o processo de pesquisa possibilitou experimentar, investigar e obter conhecimento sobre vários aspectos do design de gamificação para software científico.

#### 4.1.3.1

##### Quanto a desafios peculiares do software científico

Conforme discutido no segundo capítulo desta tese, o desenvolvimento de software científico oferece desafios específicos. No caso deste experimento, alguns destes desafios foram vivenciados e verificados em primeira mão, causando impacto sobre o processo de design e desenvolvimento, e por vezes determinando os rumos da pesquisa.

Em primeiro lugar, as *dificuldades na obtenção de requisitos* – descritos como “voláteis” (Sanders, 2008) e “emergentes” (Segal & Morris, 2008) – se refletiu na mudança de escopo do projeto que exigiu, em determinada fase da pesquisa, o abandono de um conjunto de funcionalidades – Timeline e Graph – em prol do foco em novas prioridades – ferramentas para criação de cenários. Tal dificuldade reforça a necessidade – tal como apontada pela revisão de literatura e pesquisa bibliográfica – de processos de design flexíveis o suficiente para permitir mudanças bruscas no foco do projeto, e também, para que ideias inicialmente descartadas possam ser revisitadas no futuro. De modo semelhante, o design de interfaces deveria, preferencialmente, fazer uso de elementos e mecanismos que apoiem esta flexibilidade. No caso deste projeto, o menu circular foi considerado adequado ao projeto não apenas por sua facilidade de uso, mas por facilitar a inclusão, exclusão e substituição de novas opções de modo não-intrusivo.

Em segundo lugar, o software científico pode oferecer *dificuldades na validação* de seus resultados, que podem exigir sua comparação com dados do mundo real, cálculos manuais ou avaliação de especialistas (Sanders, 2008). No caso do SiVIEP, não foi possível a implementação de modelos preditivos capazes de relacionar dados obtidos a partir de fontes diferentes. Isto sugere que uma dificuldade, quando não o impedimento, de uma implementação do “arbitrio baseado em máquina” (Ferrara, 2012, p.21, tradução do autor) presente nos jogos eletrônicos, que julgam o desempenho de seus participantes baseado em regras internas.

Em terceiro lugar, enquanto video games são livres para apresentar simulações simplificadas em prol da jogabilidade, o software científico exige *atenção à precisão, correção, exatidão e qualidade dos dados*. Tal exigência pode orientar soluções em design de interfaces, como no caso da ferramenta de

posicionamento projetada para o protótipo, que foi inspirada em funcionalidades presentes em jogos eletrônicos, mas modificada para maior grau de precisão. Desta forma, métodos de controle baseados em video games podem necessitar de mudanças que satisfaçam tal necessidade, seja através de sua combinação com outras soluções mais convencionais (no caso do protótipo, entrada de dados numéricos através do teclado) ou da modificação daqueles métodos (precisão numérica atrelada ao posicionamento da câmera).

Em quarto lugar, o desenvolvimento de software científico requer *mudanças iterativas em incrementais sobre sistemas complexos*. Além da dificuldade na obtenção de requisitos, esta exigência pode ser causada pelo uso de ferramentas e linguagens de programação mais tradicionais, as quais podem dificultar a implementação de funcionalidades altamente dinâmicas e interativas tais como as exibidas por jogos eletrônicos. A falta de atenção a este princípio foi causa fundamental do fracasso na transição entre protótipo e implementação. Isto sugere que adoção de uma mentalidade conservadora em termos de desenvolvimento deve ser adotada na fase de design.

Em quinto lugar, a proposta por interfaces inspiradas em jogos pode exigir uma *investigação quanto aos hábitos e habilidades dos usuários* a que se destinam. No caso do SiVIEP, seu direcionamento a profissionais em posições tradicionalmente ocupadas por pessoas de faixa etária mais elevada do que a média sugere menor familiaridade do seu público com video games – portanto, menos receptivos ao trabalho gamificado que usuários mais jovens (Oprescu et al., 2014), e mais resistentes a este tipo de abordagem. Neste sentido, a presença dos *Product Owners* foi providencial para este estudo, servido de contraponto à familiaridade com jogos demonstrada pela equipe de desenvolvimento, design e do próprio pesquisador, e funcionando – na ausência de usuários – como amostra do público ao qual o público se destina.

Finalmente, o *software científico deve oferecer oportunidades para insight* – necessidade para a qual por vezes se mostra insuficiente (Aragon et al., 2009). No caso do SiVIEP, encarou-se o desafio de aprimorar seus métodos de visualização de dados, além de providenciar novas maneiras e visualização adequadas à navegação interativa. Neste sentido, duas soluções foram propostas: em primeiro lugar, inspirado em *Cities in Motion 2*, sugeriu-se a exibição de gráficos ilustrando séries temporais, e não valores referentes a um único momento da série (Figura 45).

Em segundo lugar, projetou-se um sistema de controle de reprodução, inspirado em *Braid*, que facilitasse o *insight* a partir da combinação entre navegação espacial e navegação temporal. Tal funcionalidade, observou um dos membros da equipe de desenvolvimento, fazia lembrar controles de reprodução mais próximos ao tradicional, tal como aqueles oferecidos por sistemas de *streaming* como a *Apple TV*. Entretanto, foi a combinação espaço-temporal proposta por *Braid* que tornou tal funcionalidade adequada às necessidades do *SiVIEP*.



Figura 45 - SiVIEP (esq.) e Cities in Motion 2 (dir.) (Queiroz et al., 2016)

#### 4.1.3.2 Quanto à pesquisa e design de gamificação

A pesquisa-ação realizada foi útil na identificação de métodos para abordagem, pesquisa e design de elementos oriundos de jogos eletrônicos. Em primeiro lugar, a respeito da *abordagem à gamificação*, este estudo privilegiou a transposição de elementos não estruturais dos jogos, negligenciando a reformulação da atividade alvo como jogo e investindo na adoção de padrões de interatividade e estética dos video games. Uma das razões para tal decisão foram as já mencionadas dificuldades na obtenção de requisitos e validação de resultados. No caso deste projeto, não era claro como dados pré-existentes seriam integrados à simulação de modo a se estabelecer resultados mais ou menos desejáveis. A única sugestão neste sentido – feita pelo designer do software –foi a inclusão da ferramenta de desenho de tubulações equipada com calculadora de custos. É interessante lembrar, no entanto, que *Sim City*, grande influência durante o processo de design, por vezes escapa da definição de jogo para ser considerado um brinquedo de software – o que sugere

que modos de jogo aberto podem ser mais apropriados para a experimentação criativa.

Em segundo lugar, *quanto à pesquisa por elementos de design*, a inspiração para funcionalidades foi majoritariamente advinda de recursos existentes em jogos eletrônicos. Tais recursos foram lembrados durante sessões de brainstorming ou identificadas através de pesquisa a títulos específicos que atendessem a algum dos seguintes critérios:

- (a) **Proximidade temática.** No caso do software em questão, jogos dos gêneros de simulação foram escolhidos por simular a construção e gerenciamento de empreendimentos, sugerindo funcionalidades e mecânicas adequadas ao software.
- (b) **Funcionalidade pretendida.** Mesmo títulos com baixa ou nenhuma proximidade temática podem sugerir funcionalidades interessantes – no caso desta pesquisa, um exemplo deste critério a apropriação da navegação temporal presente em *Braid*.
- (c) **Tecnologias e equipamentos de entrada e saída.** O uso de equipamento especializado – como o *Wiimote*, no caso do SiVIEP – pode exigir a investigação de títulos projetados para essas tecnologias.

Finalmente, *quanto à prototipagem de alta fidelidade*, esta foi uma parte substancial do processo de pesquisa. Um dos motivos para tal foi a necessidade em demonstrar, de modo tangível, a tradução de funcionalidades de jogos para o ambiente do software a ser projetado. Neste sentido, protótipos de alta fidelidade foram capazes de *contextualizar* novas funcionalidades, transpondo-as para a interface familiar do software. Além disso, a interatividade proporcionada por estes protótipos permite maior *compreensão das habilidades exigidas* para o uso, uma vez que usuários podem, de fato, experimentar as funcionalidades propostas. Finalmente, protótipos de alta fidelidade permitem *melhor acabamento*, valorizando a experiência do usuário com elementos esteticamente agradáveis como gráficos tridimensionais sofisticados que estariam ausentes de versões em baixa fidelidade. Acabamento, propôs Steve Swink, pode ser um elemento crucial na experiência do jogador em ambientes interativos virtuais (Swink, 2009). No caso de SiVIEP, é importante observar que este software faz uso de modelos tridimensionais otimizados e bem-acabados visualmente, produzidos com

ferramentas semelhantes às utilizadas na produção de video games, ao contrário de modelos tridimensionais de altíssimo grau de detalhe geométrico, mas visualmente desinteressantes, normalmente utilizados em software de engenharia. De fato, os modelos utilizados no protótipo de alta fidelidade foram extraídos do SiVIEP.

Por outro lado, a prototipagem em alta fidelidade pode levar a propostas de realização problemática ou proibitiva. No caso desta pesquisa, A facilidade dos *engines* de jogos em dar suporte a interfaces sofisticadas e altamente interativas influenciaram, de certa forma, decisões de design, direcionando-o a uma solução complexa que não pôde ser, de fato, incorporada ao software. Por este motivo – quando desenvolvedores estão envolvidos – propõe-se que protótipos propostos tenham suas funcionalidades restritas a técnicas previamente testadas pela equipe de desenvolvimento – ou ao menos funcionalidades que possam ser rapidamente testadas. Neste sentido, uma visão clara e realista dos requisitos e limitações do software é essencial.

#### **4.1.4 Conclusão**

Este estudo, baseado na metodologia da pesquisa-ação, buscou compreender e vivenciar as particularidades do design de gamificação aplicado ao software científico, propondo-se a redesenhar funcionalidades de um software para simulação integrada em óleo e gás. Embora limitada a um caso apenas, esta investigação pôde confirmar o impacto das características do software científico sobre o design de gamificação e interfaces. A necessidade de atenção ao usuário, correção e *insight* levou a soluções interessantes, inspiradas em elementos de design oriundos dos video games – algumas das quais precisaram ser reimaginadas para atender às necessidades do software científico. Por outro lado, problemas quanto à validação de resultados, dificuldades em estabelecer requisitos e a complexidade do software influenciaram decisões de design – tal como não reestruturar a atividade do software em um formato de jogo, mas repensar suas funcionalidades para que se tornassem mais parecidas com aquelas encontradas em video games. Tal abordagem pareceu apropriada à natureza exploratória e criativa do software em questão.

A busca por funcionalidades inspiradoras se deu através da investigação de jogos representando atividades similares, contendo recursos de interação desejáveis, ou fazendo uso das tecnologias e equipamentos adotados pelo software. Funcionalidades projetadas foram apresentadas através de protótipos de baixa de alta fidelidade. O segundo tipo, desenvolvido em um *engine* de jogos, pareceu mais efetivo em demonstrar as funcionalidades propostas – mas se mostrou um problema: a facilidade na prototipagem de funcionalidades não era compatível com a dificuldade e custo de realmente implementá-las no software – considerados impeditivos. Neste caso, ficou evidente que a fase de design deve ser claramente informada a respeito de limitações nos recursos e processos de desenvolvimento, além de ser orientada a um processo iterativo, voltado para pequenas alterações, para que se maximize chances de implementação.

#### 4.2

#### **Co-design de interfaces baseado em jogos: casos em matemática, física e engenharia**

A segunda experiência realizada buscou explorar o potencial da gamificação dentro de processos de design colaborativo, propondo um método para o design colaborativo de software científico, experimentado através de oficinas de co-design e baseado nas similaridades entre jogos e prática científica. Tal proposta foi auxiliada, por um lado, pelas questões de uso e desenvolvimento do software científico e, por outro, pela análise a obras da história e filosofia da ciência, tal como relatadas nos capítulos 2 e 3, respectivamente.

Dentro dos objetivos propostos pela tese, o papel do método elaborado seria a comunicação de problemas, identificação de necessidades e a geração de ideias para melhores experiências do usuário em software científico. Desta forma, sua eficácia passa pelas seguintes questões: o método proposto é útil para que participantes expressem problemas que eles possam estar enfrentando? Os participantes conseguem explicar suas respectivas pesquisas e trabalho em software científico a partir de um modelo baseado em jogos? Eles são capazes de gerar ideias e rascunhar soluções que atendam a seus anseios?

Havia também, subjacente aos interesses práticos da pesquisa, questões de cunho teórico a serem investigadas, especificamente: quais relações entre jogo e



ciência são tornadas evidentes pelos participantes? Qual é o papel do software científico em uma ciência semelhante ao jogo?

A opção por se desenvolver um método para o design centrado no usuário se justifica pelo sucesso desses métodos na solução de problemas de usabilidade em software científico, possibilitando decisões de design mais apropriadas a este que é um campo altamente especializado e peculiar. Por outro lado, a gamificação – enquanto reestruturação de atividades como jogos – parece ter feito, até o momento, uma contribuição muito limitada ao software científico, possivelmente por não se basear em características intrínsecas de trabalho a que se destina, ou por não enfatizar seus aspectos sistêmicos. Esta experiência buscou, portanto, atender a estas duas questões, através de um método para gamificação que pudesse ser aplicado como um estágio individual de processos de design centrados no usuário.

Esta seção relata o segundo estudo de campo realizado para tese, ocorrido entre os meses de janeiro e junho de 2018, na Universidade de Leeds, no Reino Unido. A pesquisa foi desenvolvida pelo autor na condição de pesquisador visitante na Escola de Design daquela universidade, sob supervisão de Dra. Maria Lonsdale, professora associada em design gráfico e de comunicação, e líder do programa de mestrado em design. O estudo foi aprovado pelo Comitê de Ética em Pesquisa da Faculdade de Artes, Humanidades e Culturas da Universidade de Leeds, sob o código de referência LTDESN-081. Para coerência com o restante da tese, todo o material produzido e coletado pela tese foi traduzido para o Português pelo autor.

As próximas subseções descrevem o desenrolar da pesquisa de acordo com a seguinte estrutura: primeiro, a seção *Metodologia* descreve os métodos de pesquisa adotados, critérios para seleção de participantes, ferramentas, protocolos, etapas, procedimentos, e critérios para coleta de dados. Em seguida, três seções referentes aos *Estudos de Casos* descrevem as oficinas realizadas, seus contextos, processos e resultados. A subseção seguinte, *Discussão*, analisa os estudos de caso, discutindo a relevância e eficácia do método e suas ferramentas e, também, associando seus resultados às relações entre jogo e ciência. Finalmente, a seção de *Conclusão* resume as informações discutidas e as lições aprendidas com a pesquisa.

### 4.2.1 Metodologia

Este estudo foi baseado em métodos de pesquisa em co-design – uma abordagem através da qual várias partes envolvidas em um problema (usuários, desenvolvedores, cidadãos, etc.) participam ativamente do processo de design de soluções, no qual designers atuam como facilitadores. De acordo com Sanders & Stappers, co-design pode ser definido como “a criatividade de designers e de pessoas sem treino em design trabalhando juntas em processos de desenvolvimento de design” (2008, p.6, tradução do autor). No caso deste estudo, através de sessões individuais com usuários e desenvolvedores de software científico, explorou-se o método de design proposto como ferramenta para a ideação de soluções e, também, para a discussão acerca de similaridades entre pesquisa científica e jogos. As sessões foram registradas em anotações e fotografias, além do material produzido pelos próprios participantes, e que serviram como principal fonte dos dados recolhidos. Cada seção durou entre 90 e 120 minutos. Foram arranjados encontros individuais subsequentes para discussão adicional e avaliação de protótipos.

Sessões individuais de co-design não são comuns em processos colaborativos de design. Entretanto, frequentemente o software científico é desenvolvido e utilizado por pequenos grupos (ou mesmo indivíduos) com necessidades particulares. Em alguns casos, usuários são, eles mesmos, responsáveis por desenvolver ou estender a funcionalidade do software. Por estes motivos, sessões individuais foram conduzidas, uma vez que questões de usabilidade podem estar associadas a necessidades individuais, atreladas ao domínio científico, especialização, métodos e estágio da pesquisa. Eventualmente, soluções desenvolvidas podem ser generalizadas para um público maior. Trata-se de uma abordagem que não foi aplicada à experiência anterior, e cujo potencial foi, justamente, reconhecido com base na dificuldade daquele experimento em gerar soluções e identificar problemas adequadamente.

Participantes foram recrutados entre usuários e/ou desenvolvedores de software científico que estivessem realizando pesquisa nas seguintes áreas denominadas STEM (ciência, tecnologia, engenharia e matemática). Foram selecionados participantes de diversos ramos da ciência, bem como em diferentes estágios de desenvolvimento acadêmico e profissional. O processo de recrutamento envolveu

o envio de chamada para participação – em mensagem enviada para diversos departamentos da Universidade de Leeds – e, também, o envio de e-mails diretamente a pesquisadores individuais. Todos os participantes colaboraram de forma puramente voluntária.

#### 4.2.1.1

##### Materiais e ferramentas

Foi especialmente criado para o método um kit de ferramentas consistindo de um conjunto de 77 cartas. Três destas cartas representam os estágios típicos da pesquisa baseada em software (modelagem, simulação e análise de resultados). Seis cartas apresentam diretrizes baseadas na transposição dos elementos definidores do jogo para o software científico (Figura 46). As cartas diretrizes descrevem e enfatizam a importância dos elementos da dinâmica do software científico de acordo com o modelo baseado em jogos, ilustrada na Figura 47. Entretanto, embora o modelo sugira uma ordem sequencial para os elementos (esforço, regras, representação, quantificação, valorização e satisfação), optou-se por mantê-los independentes, possibilitando ao participante estabelecer as relações que preferir. O conjunto completo de cartas está disponível nos Anexos.

<b>Esforço do Usuário</b> <ul style="list-style-type: none"> <li>• O que usuários fazem?</li> <li>• Quais são seus desafios e ações?</li> </ul>	<b>Regras do Sistema</b> <ul style="list-style-type: none"> <li>• Como o sistema funciona?</li> <li>• Como ele deve ser usado?</li> </ul>
<b>Representação do Sistema</b> <ul style="list-style-type: none"> <li>• Como o objeto de pesquisa é exibido?</li> <li>• Como é a experiência com esta representação?</li> </ul>	<b>Quantificação de Resultados</b> <ul style="list-style-type: none"> <li>• Como o resultado é medido?</li> <li>• Como ele é materializado?</li> </ul>
<b>Valorização dos Resultados</b> <ul style="list-style-type: none"> <li>• O que torna um resultado valorizado?</li> <li>• Como ele é valorizado?</li> </ul>	<b>Satisfação do Usuário</b> <ul style="list-style-type: none"> <li>• Que resultados deixariam usuários satisfeitos?</li> <li>• O que poderia tornar a experiência mais agradável ou divertida?</li> </ul>

Figura 46 - Cartas diretrizes. Traduzido de Queiroz et al., submetido

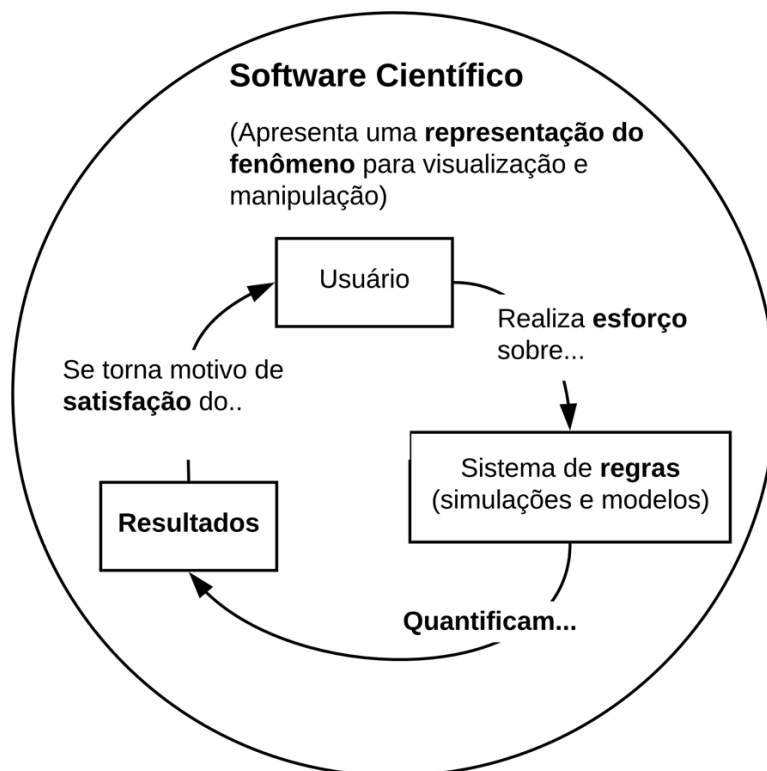


Figura 47 - Visão sistêmica. Adaptado de Queiroz et al., submetido

Finalmente, as 68 cartas restantes contêm ilustrações e palavras-chaves representando elementos lúdicos e características da ciência e software científico, tais como identificados nos capítulos 2 e 3, além de referências a elementos de design comuns a jogos. Para que se não se ultrapassasse um número de cartas razoável para os propósitos da dinâmica, nem todas as características identificadas foram incluídas – ainda que tenha se tentado maximizar o número de termos incluídos através do uso de múltiplas palavras-chave em cada carta, possibilitando múltiplas interpretações. Uma amostra das cartas está ilustrada na Figura 48.



Figura 48 - Amostra de cartas ilustradas. (Queiroz et al., submetido)

Além do conjunto de cartas, foi disponibilizado, durante as sessões, o seguinte material:

- Adesivos do tipo *Post-it*;
- Papel em tamanhos A4 e A3;
- Canetas, lápis e marcadores coloridos;
- Cola e tesoura.

#### 4.2.1.2 Procedimento

As sessões foram estruturadas como uma série de atividades curtas (uma visão geral está ilustrada na Figura 49):

*Contexto* – O participante é convidado a se apresentar, descrever sua pesquisa, seu uso de software e descrever aspectos do trabalho os quais considera agradável ou divertido.

*Problema* – O participante é perguntado a respeito de problemas que possa estar tendo com o uso do software, ou sobre funcionalidades que desejaria. Ele é convidado a sintetizar os problemas (ou funcionalidades) em um adesivo – e é solicitado priorizar apenas uma questão, caso haja mais de uma, de acordo com sua simplicidade.

*Categorização* – Pede-se ao participante para que classifique seu problema quanto à fase à qual o problema pertence (modelagem, simulação ou análise de resultados).

*Analogia* – Descreve-se ao participante a analogia entre o trabalho em software científico e o jogo com auxílio das cartas diretrizes que são, então, entregues ao

participante. A explicação da analogia deve enfatizar a natureza dos jogos enquanto atividades estruturadas e orientada a objetivos, e o modo como oferecem maneiras interessantes de interação.

*Exploração* – Posiciona-se as 68 cartas ilustradas sobre uma superfície. Pede-se ao participante para que explore as cartas e escolha aquelas que ele julgar relacionadas ao problema ou a possíveis soluções. Solicita-se ao participante que reflita e escreva, em adesivos, a relação entre cada carta e o problema. Deve-se deixar o participante à vontade para que interprete ou subverta o significado das cartas como julgar mais apropriado.

*Rascunho* – Pede-se ao participante que desenhe uma interface que reflita as soluções vislumbradas a partir das cartas selecionadas.

*Reflexão* – Pede-se ao participante que responda as perguntas das cartas diretrizes, e que escreva em adesivos como o rascunho atende a cada uma delas. Participantes podem, durante esta etapa, acrescentar novas ideias a seus rascunhos.

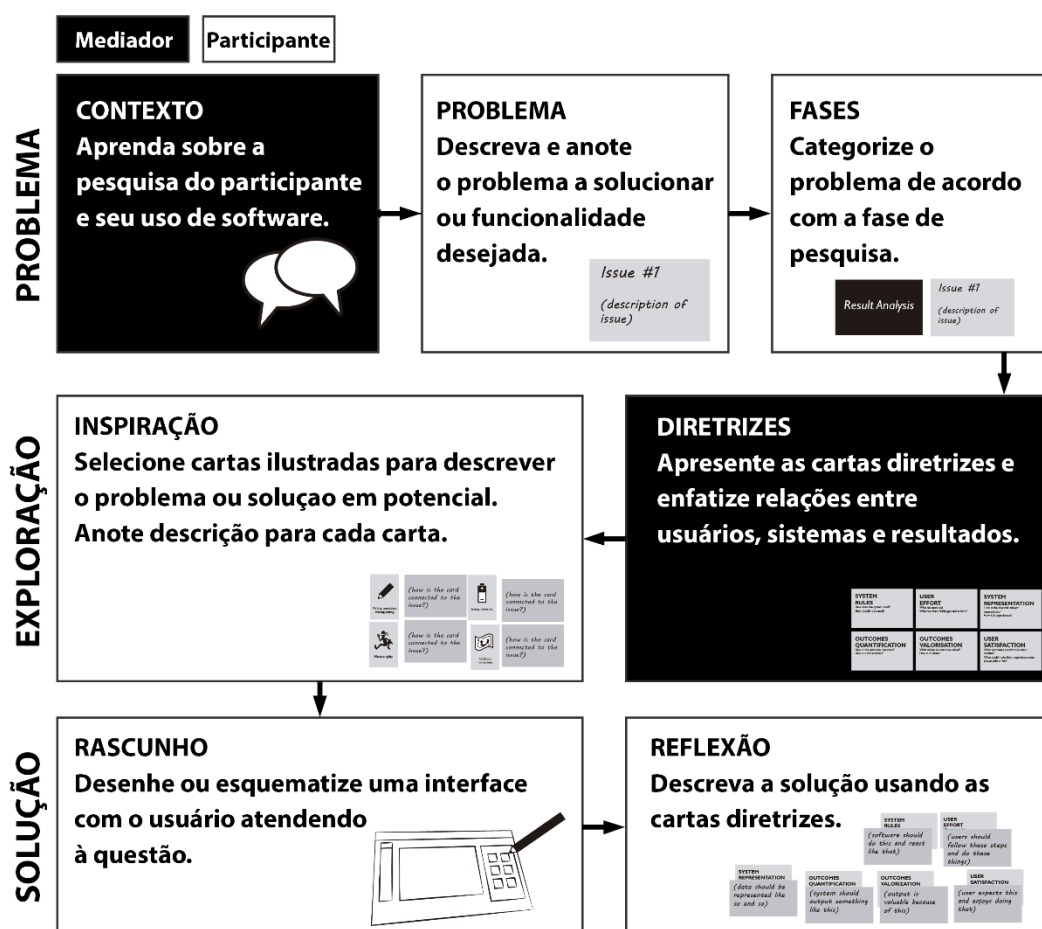


Figura 49 - Diagrama do procedimento (Queiroz et al., submetido)

#### 4.2.1.3

##### Coleta de dados

Durante as sessões, os seguintes aspectos foram observados:

- Área do conhecimento científico e grau de especialização do participante;
- Posicionamento do participante dentro do espectro usuário-desenvolvedor;
- Aspectos da pesquisa descritos como divertidos ou agradáveis;
- Descrição de problemas de usabilidade e propostas para novas funcionalidades;
- Fases da pesquisa (modelagem, simulação ou análise de resultados) ao qual pertence a questão levantada;
- Uso de cartas ilustradas na descrição de problemas;
- Uso das cartas diretrizes na descrição de soluções;
- Capacidade do participante em rascunhar solução;
- Aspectos lúdicos das soluções propostas.

#### 4.2.1.4

##### Acompanhamento

Após as sessões, o pesquisador produziu protótipos interativos para simular as funcionalidades pretendidas pelos participantes. Estes protótipos foram produzidos através de software de edição de imagens (Adobe Photoshop) e ferramenta autoral multimídia (Adobe Animate). Foram realizados, casuisticamente, pequenos encontros com participantes para esclarecimentos e discussões, a partir das quais novas iterações do protótipo foram geradas, quando julgado necessário.

#### 4.2.2

##### Estudo de caso I: plotagem de gráficos em física de matéria condensada

O Participante 1 (P1) é um pesquisador de pós-doutorado no Grupo de Física de Matéria Condensada. Em sua pesquisa sobre nanopartículas ferromagnéticas, ele utiliza o software *OriginPro 2017* para análise de resultados, examinando conjuntos de dados de carga elétrica em nanopartículas. De acordo com P1, a tarefa de se

explorar grandes quantidades de dados – e tentar compreendê-los – tem características de imersão e diversão. Entretanto, P1 acredita que seu trabalho poderia ser facilitado, e sente que a plotagem de gráficos em série requer muito trabalho manual e muitos passos desnecessários. Este um problema duplo para ele, já que (1) seu raciocínio e *insight* são atrasados até que ele gere e ajuste manualmente uma série de gráficos e (2) a preparação de material para apresentações e publicações leva um tempo longo e entediante.

#### 4.2.2.1 Problema

O problema selecionado por P1 pertence à fase de análise de resultados, e é uma combinação de dois problemas:

*“Plotagem de painéis com vários gráficos de eixo compartilhado”;*

*“Alterar partes de uma plotagem rapidamente – mudar a escala, etc.”.*

#### 4.2.2.2 Exploração

P1 selecionou seis cartas e descreveu pequenas reflexões sobre suas relações com o problema (Figura 50).

 Maestria; agilidade.	‘Deve ser ágil e rápido’	 Fluxo; imersão.	‘Criação de gráficos deveria ser intuitivo. Etapas naturais’
 Velocidade; viagem.	‘Quero a plotagem logo para poder ir à próxima fase’	 Guia; instruções	‘De novo, muitos passos para alcançar o resultado’
 Precisão; exatidão.	‘Plotagens deveriam se posicionar automaticamente em lugares sensatos’	 Conflito; concorrência.	‘o software está trabalhando contra mim!’

Figura 50 - Cartas e comentários de P1 (Queiroz et al., submetido)



### 4.2.2.3

#### Rascunho e diretrizes

P1 desenhou uma interface (Figura 51), imaginada para ser uma extensão customizada para *OriginPro 2017*. Um comando adicional – clicar o gráfico com o botão direito do mouse para selecionar fonte dos dados – não foi incluída no rascunho, mas comunicada verbalmente.

Em resposta às cartas diretrizes, P1 escreveu notas específicas e diretas conforme ilustração (Figura 52).

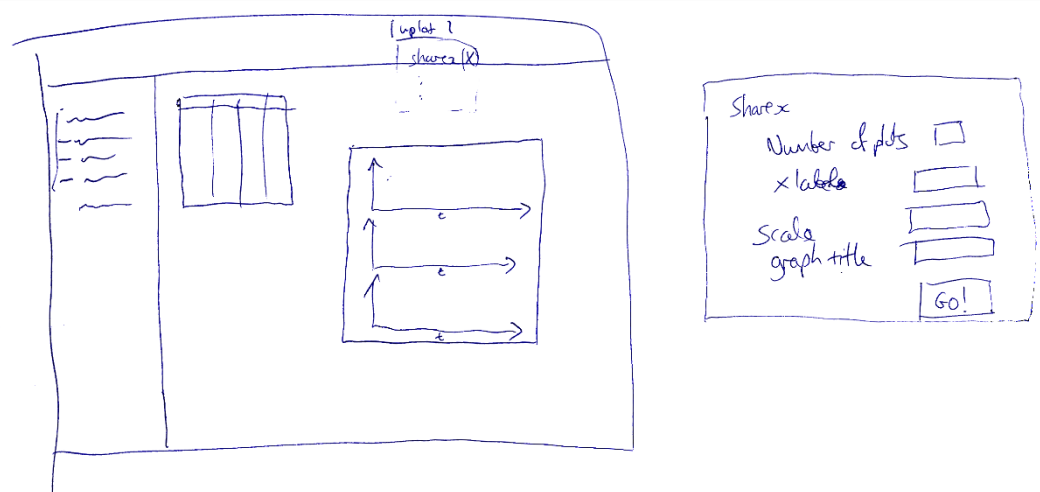


Figura 51 - Rascunho de P1 (Queiroz et al., submetido)

<p><b>Esforço do Usuário</b>          '[usa opção do menu] uplot &lt;-clica menu          compartilhar x &lt;-clicar (ou atalho teclado)          compartilhar y          multipainel'</p>	<p><b>Regras do Sistema</b>          'Pede informações da plotagem ao usuário -          rótulo/escala.          Type of presentation.          Create empty plot'</p>
<p><b>Representação do Sistema</b>          'Vantagen [pode ser de] qualquer tamanho          padronizado          Rótulos com fontes certas, etc'</p>	<p><b>Quantificação do Resultado</b>          'Resultado é plotagem de preferência do          usuário'</p>
<p><b>Valorização do Resultado</b>          'Valorizado se próximo ao que era          necessário. Informações padronizadas na          plotagem para a apresentação certa'</p>	<p><b>Satisfação do Usuário</b>          'Plotagem próxima das expectativas.          Menos botões clicados.          Plotagem parece melhor'</p>

Figura 52 - Comentários às diretrizes - P1 (Queiroz et al., submetido)

#### 4.2.2.4 Acompanhamento

Um protótipo digital (Figura 53), baseado em capturas de tela do *OriginPro 2017*, foi apresentado a P1, que solicitou a inclusão da janela de diálogo (já existente naquele software) para configuração de eixos. A funcionalidade foi incluída em uma segunda versão, apresentada no encontro seguinte. P1 se mostrou satisfeito com o funcionamento do protótipo e se ofereceu para mostra-lo a seus colegas. Durante o acompanhamento, o pesquisador comunicou a P1 sobre a inclusão de uma funcionalidade para geração de múltiplos gráficos na versão mais recente do software *OriginPro*. Este fato não reduziu o entusiasmo de P1 por sua própria solução.

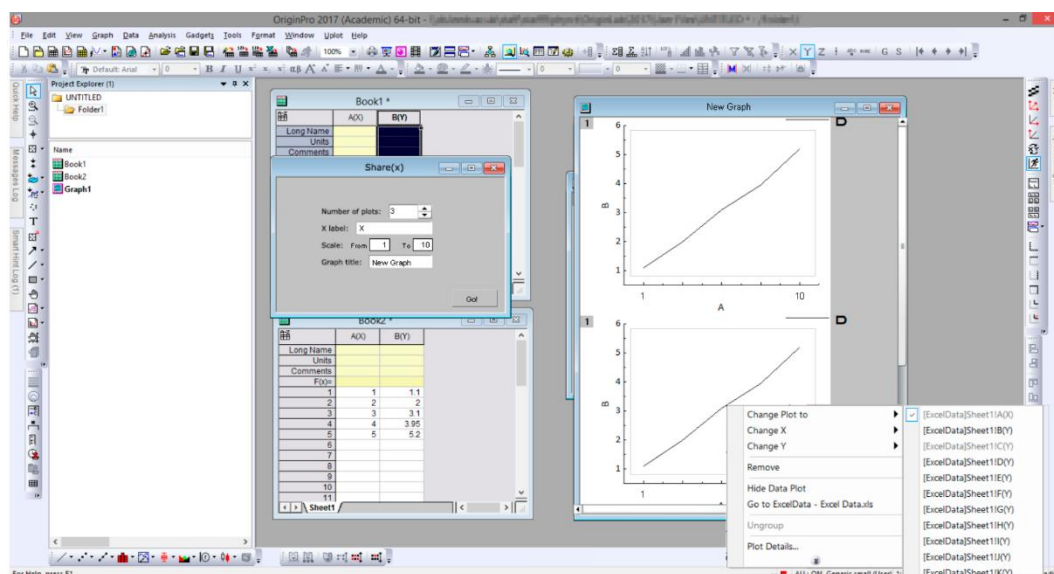


Figura 53 - Protótipo feito sobre capturas de tela (Queiroz et al., submetido)

#### 4.2.3 Estudo de caso II: visualização em matemática física

O Participante 2 (P2) é um estudante de doutorado na Escola de Matemática, onde estuda física matemática. Sua pesquisa em física de partículas elementares envolve a elaboração de algoritmos para a solução de equações diferenciais parciais (EDPs) e a visualização de representações geométricas do fenômeno pesquisado. Através de métodos de visualização que envolvem a plotagem de superfícies

tridimensionais representando um conjunto de elétrons. P2 obtém *insight* a respeito do comportamento do sistema representado.

P2 realiza seu trabalho nos softwares *SciPy* – um conjunto *open-source* de ferramentas e bibliotecas para computação científica – e *Spyder* – que inclui um ambiente para computação numérica alimentado por várias bibliotecas e, também, um ambiente de desenvolvimento interativo através dos quais usuários podem editar e executar código. De fato, a maior parte do trabalho de P2 é realizada programaticamente, pois é através de código que ele implementa algoritmos para solução das EDPs que levarão à simulação correta, quando executado. As fases de modelagem e simulação são, nesse caso, muito próximas entre si – praticamente indistinguíveis, na opinião de P2.

P2 descreve sua pesquisa e a elaboração de algoritmos como divertidas. De acordo com ele, a diversão está em não saber exatamente qual será o resultado – ou se o resultado será válido. Ele descreve como particularmente divertida uma ocasião em que pesquisou intensamente a implementação de um algoritmo para EDP – e teve como resposta de seu orientador uma explicação sobre porque sua solução era inadequada;

#### **4.2.3.1 Problema**

Questionado quanto a problemas de usabilidade, P2 se concentrou nas fases de simulação e análise de resultados:

*“Como simular campos de vetor tridimensionais relacionados a uma EDP / Interatividade com parâmetros”.*

#### **4.2.3.2 Exploração**

P2 selecionou e comentou dez cartas, como ilustrado na Figura 54.



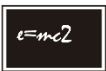







 <p>Escrever; anotação; desenhar; editar.</p>	<p>'Ciclo no meu trabalho diário: / escrever resultados parciais / anotar novas descobertas / desenhar algo significativo no problema/ corrigir erros'</p>	 <p>Imaginação; pensamento; raciocínio.</p>	<p>'Principalmente em matemática pura, os problemas interessantes são abstratos. Para pesquisar deve-se pensar e imaginar o que outra pessoa descobriu somente com artigos'</p>
 <p>Formula; teorias; equações.</p>	<p>'Tudo da literatura é descrito por equações (EPDs and EDOs na maior parte)'</p>	 <p>Visão; Visualidade</p>	<p>'Meu problema de pesquisa é geométrico. Vem da física e é uma abstração do mundo real 3D'</p>
 <p>Software; programação.</p>	<p>'Para visualizar descobertas - ou - para buscar direcionamento a respeito de qual direção seguir'</p>	 <p>Documento; arquivo</p>	<p>'Documentos armazenam a informação que preciso. Leio muitos deles, ou vejo plotagens de significado físico ou geométrico'</p>
 <p>Análise de dados; visualização de dados</p>	<p>'Porque é um problema geométrico em um "espaço infinito-dimensional" reduzido por simetrias'</p>	 <p>Base de conhecimento; biblioteca</p>	<p>'Leio muitos artigos, procurando por pistas para dar pequenos passos em um problema maior'</p>
 <p>Linha de chegada; objetivo.</p>	<p>'É a motivação para pesquisar (alcançar o objetivo)'</p>	 <p>Quebra-cabeças; solução de enigmas.</p>	<p>'Em matemática, pesquisa é como resolver um quebra-cabeças. Eu tenho pistas e divido o problema em pedaços menores. Depois devo descobrir como encaixá-las'</p>

Figura 54 - Cartas e comentários de P2 (Queiroz et al., submetido)

### 4.2.3.3 Rascunho e diretrizes

P1 concebeu uma plataforma (Figura 55) onde pudesse utilizar diferentes janelas para inserir parâmetros, visualizar campos de vetor (duas janelas diferentes, para diferentes tipos de visualização), controlar a reprodução da simulação, capturar *frames* da animação da simulação, adicionar comentários às plotagens e exportar imagens para LaTeX (editor de artigos científicos). Os comentários de P2 a respeito das cartas diretrizes estão ilustrados na Figura 56.

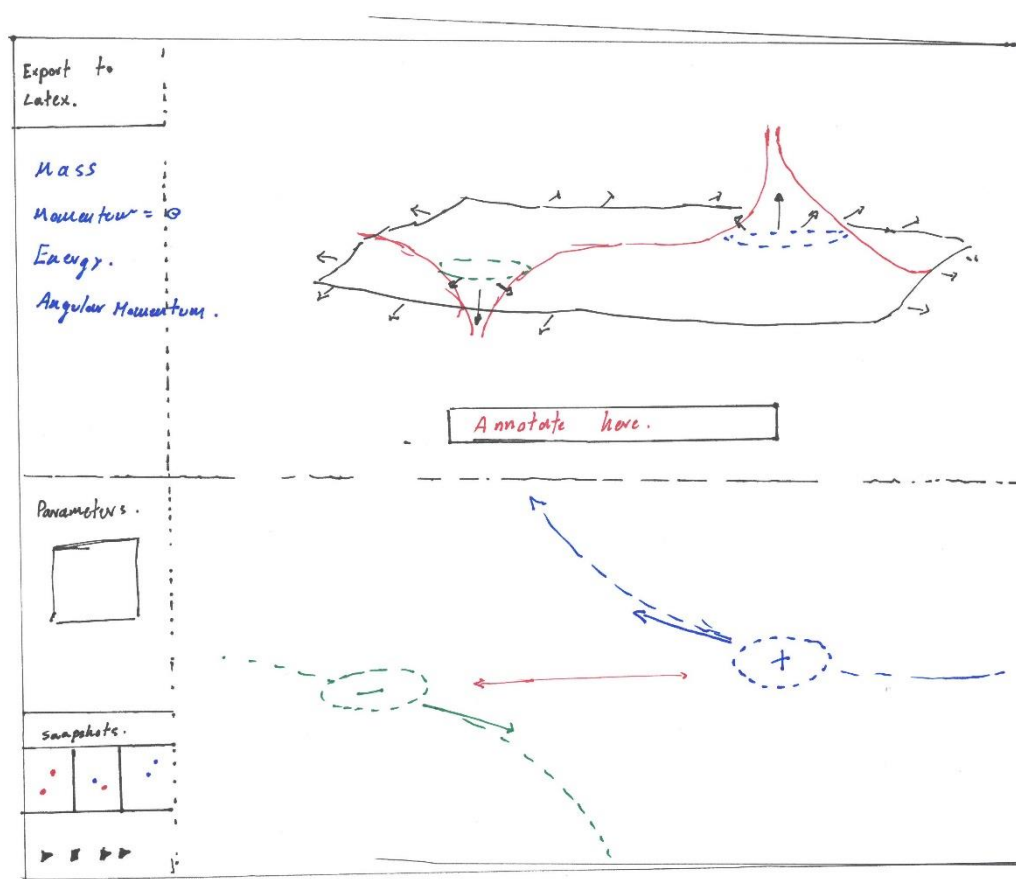


Figura 55 - Rascunho de P2. (Queiroz et al., submetido)

<b>Regras do sistema</b> 'O sistema inicia com a plotagem da última simulação. Ele lembra dos parâmetros e últimas locações visitadas para salvar capturas de tela / Deve ser usado entrando parâmetros, e deve ser interativo, para que o usuário possa rotacionar ou dar zoom-in e zoom-out no sistema'	<b>Esforço do Usuário</b> 'Usuários configuram parâmetros, depois rodam uma simulação matemática, gerando uma configuração estática do sistema e a posição inicial do par vortex-antovortex / Depois, usuários rodam a simulação da dinâmica do sistema / Desafio: refinar os parâmetros'	<b>Representação do Sistema</b> 'O objeto de pesquisa é representado com gráficos e a experiência acontece com uma interface para visualização do sistema estático e dinâmico'
<b>Quantificação do Resultado</b> 'O resultado é medido com base nos resultados conhecidos na literatura./ Se materializa com plotagens e animações.'	<b>Valorização do Resultado</b> 'Um resultado é valorizado se representa a física fielmente / É valorizado porque as imagens são claras, e a trajetória dinâmica é fácil de entender'	<b>Satisfação do Usuário</b> 'Belas plotagens, fácil de exportar em modo amigável ao LateX/ Animações da dinâmica do sistema / Interface intuitiva que "aprender" como o usuário interage tornaria mais agradável'

Figura 56 - Comentários às diretrizes – P2 (Queiroz et al., submetido)

#### 4.2.3.4 Acompanhamento

Ao ser apresentado a uma versão inicial de protótipo digital, P2 fez uma série de observações a respeito dos parâmetros da simulação, descrevendo quais deveriam ser editáveis. Ele explicou a impossibilidade de plotagem de animações em tempo real, devido ao poder de processamento necessário para o cálculo de cada *frame*. Finalmente, ao experimentar o protótipo interativo, P2 imaginou uma nova funcionalidade, através da qual partículas em um plotagem pudessem ser arrastadas para diferentes posições, a partir das quais a interação entre os elementos seria automaticamente recalculada.

#### 4.2.4 Estudo de caso III: análise de dados de fluxo em tomografia de processo

O Participante 3 (P3) é um pesquisador de pós-doutorado na Escola de Engenharia Química e de Processo. Sua área de pesquisa é a tomografia de processo, dentro da qual desenvolve um software dedicado à visualização e análise de fluxo no interior

de reatores e tubulações. O software, entretanto, é utilizado externamente – P3 não é, ele mesmo, um usuário.

Através do software, usuários podem analisar dados de fluxo e exportar imagens para relatórios. De acordo com P3, o software soluciona vários desafios de visualização: a ineficiência de imagens transversais em revelar informações sobre a dinâmica do fluxo; a necessidade de se aferir mais de uma modalidade de fluxo para se compreender seu padrão; e a necessidade de se obter resultados quantitativos e qualitativos – o segundo a depender da interpretação de um especialista.

#### **4.2.4.1**

##### **Problema**

P3 não foi capaz de descrever um problema com a usabilidade de seu software, tendo recebido apenas *feedback* positivo de seus usuários. Idealmente, P3 gostaria de *implementar uma funcionalidade para o cálculo e visualização da velocidade do fluxo*. Entretanto, ele acredita que a implementação desta funcionalidade não seria possível, dada sua complexidade, prazos e recursos disponíveis para o projeto

#### **4.2.4.2**

##### **Exploração**

P3 utilizou as ferramentas de ideação para explicar o funcionamento e as decisões de design do projeto original. Inicialmente, P3 havia selecionado dezoito cartas ilustradas – número reduzido a nove, após ser solicitado a se concentrar apenas em questões de usabilidade. As nove cartas e seus comentários estão ilustrados na Figura 57.

 <p>Dimensões; simplicidade.</p>	<p>‘(1) fusão de dados multi-dimensional; (2) reduzir dimensões para melhor exibição’</p>	 <p>Medição; conformidade.</p>	<p>‘É sobre medição de fluxo’</p>
 <p>Construção de modelos; montagem</p>	<p>‘O software é construído sobre modelos’</p>	 <p>Análise de dados; visualização de dados.</p>	<p>‘O objetivo é visualizar os dados unidos e obter mais informação dos dados’</p>
 <p>Reprodução; controle.</p>	<p>‘Precisa controlar exibição dos resultados’</p>	 <p>Imagem; fotografia.</p>	<p>‘Os resultados são apresentados em imagens e comparados às imagens de câmera de alta velocidade’</p>
 <p>Precisão; exatidão.</p>	<p>‘Resultados devem ser precisos para refletir dinâmica do escaneamento multi-fase’</p>	 <p>Medidores indicadores.</p>	<p>‘(1) medição de fluxo; (2) critério do que é um bom resultado’</p>
 <p>Caixa preta; autosuficiente; opacidade.</p>	<p>‘O software é fechado como caixa preta para os usuários’</p>		

Figura 57 - Cartas e comentários de P3 (Queiroz et al., submetido)



### 4.2.4.3 Rascunho e diretrizes

P3 desenhou a interface do software original, acrescida de um gráfico para exibição de velocidade de fluxo (Figura 58). Durante o processo, P3 declarou interesse em experimentar alternativas ao visual da interface gráfica do software. Ele exibiu o software em seu computador, providenciando uma captura de tela posteriormente (Figura 59).

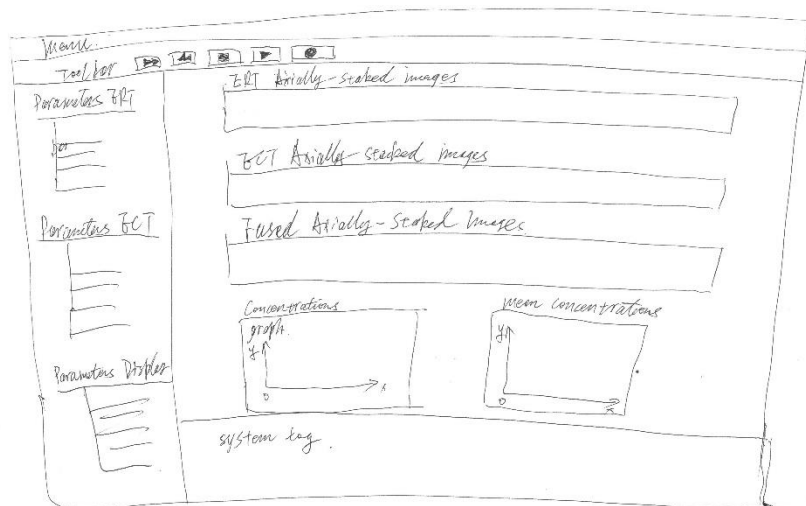


Figura 58 - Rascunho de P3 (Queiroz et al., submetido)

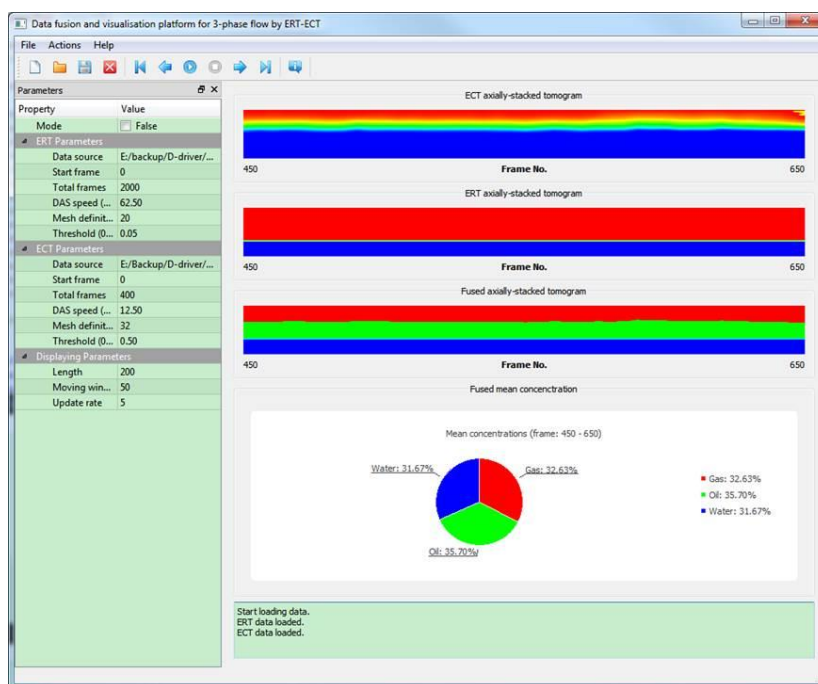


Figura 59 - Captura de tela obtida de P3. (Queiroz et al., submetido)

P3 utilizou as cartas diretrizes para descrever a proposta de uso do software, como ilustrado na Figura 60.

<b>Esforço do Usuário</b> <i>'(1) carregar dados (2) inserir parâmetros (3) executar/brincar com os resultados (4) Deve ter algum conhecimento para inserir parâmetros "válidos"</i>	<b>Regras do Sistema</b> <i>'(1) Deve apresentar bons resultados quando comparados ao vídeo de alta velocidade. (2) Para análise de dados offline e monitoramento online'</i>
<b>Representação do Sistema</b> <i>'As imagens de cada modalidade devem ser fundidas em uma única imagem, e a concentração média deve ser apresentada. Os resultados devem ser exportados.'</i>	<b>Quantificação do Resultado</b> <i>'Distribuição de fase e Decomposição de fase'</i>
<b>Valorização do Resultado</b> <i>'(1) comparado a registro de vídeo de alta velocidade; (2) comparado à referência'</i>	<b>Satisfação do Usuário</b> <i>'Se os resultados são razoáveis comparados à referência'</i>

Figura 60 - Comentários às diretrizes – P3 (Queiroz et al., submetido)

#### 4.2.4.4 Acompanhamento

Em um encontro subsequente, foi apresentado um protótipo contendo a funcionalidade para cálculo e visualização da velocidade do fluxo (Figura 61). O protótipo incluía, também, um estilo visual alternativo – incluindo ícones baseados em uma fonte iconográfica *open-source*, para maior facilidade na inclusão de futuros elementos. P3 pareceu satisfeito com o resultado e solicitou o material gerado. Questionado se acreditava que usuários descreveriam o trabalho no software como ‘divertido’, P3 afirmou que usuários deveriam considerar divertida a velocidade na qual poderiam realizar a análise dos dados através do programa.

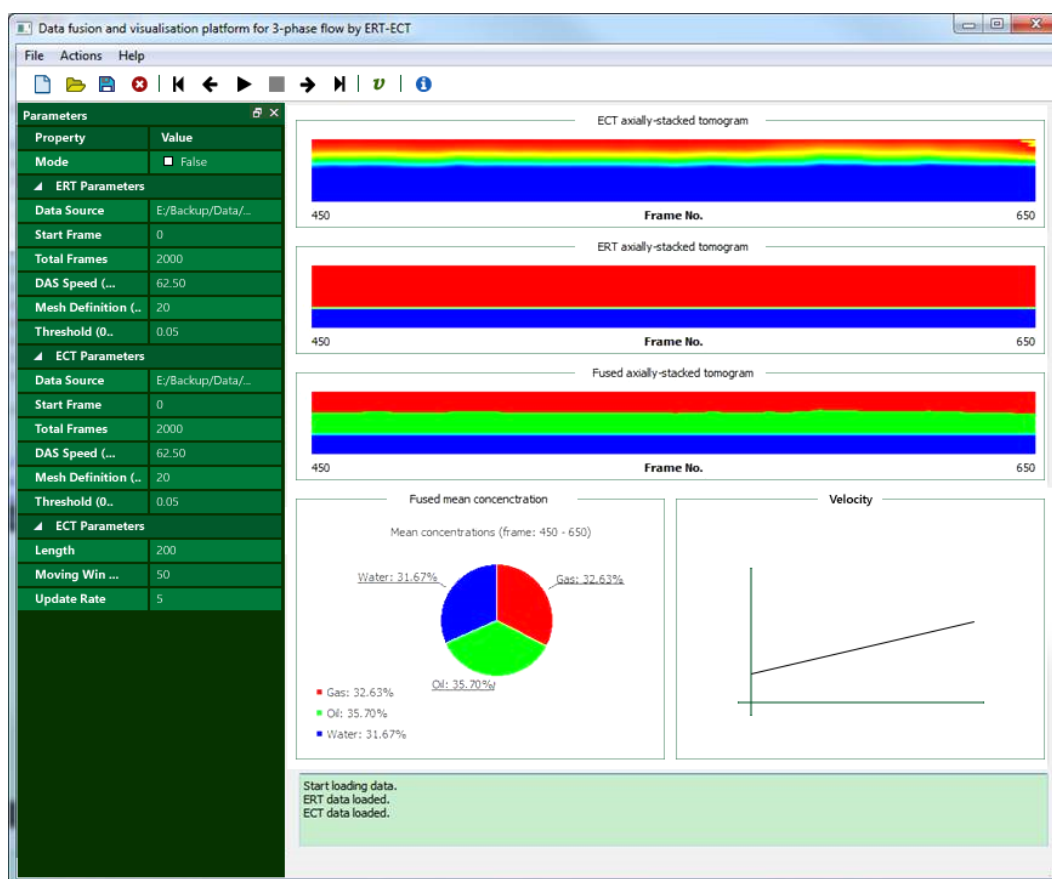


Figura 61 - Protótipo baseado em capturas de tela (Queiroz et al., submetido)

## 4.2.5 Discussão

As subseções a seguir discutem os resultados do estudo relatado a partir de duas perspectivas diferentes – ambas fundamentais para os objetivos desta tese. Em primeiro lugar, apresenta-se uma contribuição ao debate quanto as relações entre jogo e ciência – e o papel do software nesta relação. Em segundo lugar, avalia-se os resultados alcançados pelos participantes a partir do método proposto.

### 4.2.5.1 Quanto a relação entre jogo e ciência

Os estudos de caso apresentados corroboram diversas questões levantadas pela pesquisa bibliográfica e seu quadro de referência teórico, apontando para relações

entre software científico e os elementos definidores do jogo, tal como exploradas no capítulo 3 desta tese.

A respeito do *esforço*, todos estudos de caso enfatizam características relativas à solução de problemas ou de quebra-cabeças. A solução de problemas foi associada, pelos participantes, à fase de análise de resultados – particularmente através da interpretação de gráficos e figuras. Entretanto, tais características se mostraram presentes, também, no relato do Participante 2 a respeito da revisão de literatura e do desenvolvimento de algoritmos baseados em equações. No contexto do uso do software científico, a solução problemas é feita, principalmente, através da análise de resultados. Tão ou mais importante, no entanto, é perceber o quanto deste esforço é direcionado a *criar* a visualização dos dados – e não apenas interpretá-la – tanto para o *insight* quanto para a geração de publicações. De fato, todos os estudos de caso descrevem a geração de visualização de dados como sendo o resultado do trabalho com software – o que reflete a descrição da fase de análise de resultados como descrita no capítulo 2. É interessante reforçar que é são os próprios usuários que atribuem valor às visualizações e resultados obtidos, pois são eles que detém os critérios de valorização dos resultados, e são eles quem, naquele momento, podem julgar se foram ou não bem-sucedidos em suas tarefas. No entanto, trata-se de um julgamento que, mais tarde, será corroborada ou refutada pela comunidade científica na qual estão inseridos. Por isso, em um jogo da ciência, é possível afirmar que usuários não abordam o software científico como um jogo, mas como uma ferramenta ou uma etapa em um jogo que se estende a um amplo ecossistema de software, máquinas e métricas para o sucesso. Pode haver, sim, uma noção de progresso: o Participante 1 afirmou querer “avançar para a próxima fase”, e o Participante 2 declarou que sua motivação é “alcançar o objetivo”. No entanto, a avaliação da performance dos usuários e o uso de elementos de gamificação tradicionais (como pontos, placares e distintivos) pode ser problemático, não refletindo o verdadeiro estado da pesquisa. Dentre os relatos apresentados, o conceito de progresso é mais evidente na descrição feita pelo Participante 2 da literatura científica: ela pode ser o ponto de partida (revisão), o objetivo final (publicação), e – ao lado dos colegas e supervisores – o juiz que arbitra os esforços do pesquisador (referências). Entretanto, ler, escrever e organizar artigos são atividades conduzidas através de outros tipos de software. Até onde vai a análise de resultados – normalmente o último estágio no uso de software científico – os

objetivos do usuário são a obtenção de *insight* e a geração de visualização de dados.

O papel do cientista na avaliação de seus próprios resultados reflete a relação com as *regras*. O relato do Participante 2 – que programa, ele mesmo, os algoritmos que determinam o comportamento das simulações – deixa claro que pesquisadores projetam ativamente suas metodologias, experimentos e modelos. Eles constroem as regras tanto quanto as seguem. Neste sentido, o software desenvolvido pelo Participante 3 é utilizado de modo muito mais restrito, como uma caixa preta a qual usuários não podem manipular livremente. Ainda assim, mesmo estes usuários precisam julgar os resultados que obtém baseados em seus conhecimentos do domínio.

No que se refere a *consequências negociáveis*, os casos dos Participantes 1 e 3 pressupõem condições laboratoriais para o monitoramento de fenômenos a partir de instrumentos e, também, a matematização dos dados obtidos. No entanto, a separação entre mundo real e software científico é ainda mais evidente no relato do Participante 2: em primeiro lugar, por seu nível de abstração. Em segundo lugar, por não depender da observação de fenômenos no mundo real. O terceiro motivo, no entanto, não está relacionado com a natureza de seu trabalho, mas sim com a atitude do pesquisador. Sua capacidade de se divertir ao falhar na obtenção de uma resposta correta aponta para valores de realização que vão para além de “vencer”. Jesper Juul argumenta que “a liberdade encontrada em jogos comuns só pode ser preservada se tivermos espaço para experimentar e a liberdade para fracassar, ao menos temporariamente” (Juul, 2013, p.122, tradução do autor). Esta liberdade aproxima a pesquisa científica dos jogos, contribuindo para a separação entre estas atividades e a “vida real”.

Finalmente, a respeito do *apego* aos resultados, os Participantes 1 e 2 enfatizaram o prazer em obter imagens adequadas para publicação. Por sua vez, o Participante 3 destacou a satisfação de se obter resultados adequados. Quanto a satisfação dos usuários com o *uso* do software, os participantes expressaram preocupações com a facilidade e agilidade dos softwares. O Participante 1 buscou reduzir a carga de trabalho através da automação da plotagem de múltiplos gráficos. A solução proposta por ele – a ativação sequencial de diversos menus contextuais ao longo dos vários gráficos – combinada ao *feedback* visual imediato providenciado pelo próprio gráfico – teria potencial para transformar uma tarefa repetitiva em um estado de fluxo. Através desta decisão de design, o participante

manifestou, talvez inconscientemente, os conceitos de “agilidade” e “maestria” presentes na carta que selecionou, ilustrada com um guerreiro ninja. Esta solução, junto à caixa de diálogo, buscava minimizar uma sensação de conflito (“*o software está trabalhando contra mim*”). Neste ponto, é interessante notar que enquanto video games estabelecem um tom de conflito através de obstáculos e oponentes, os relatos dos participantes descrevem interfaces complexas ou desajeitadas como a principal fonte de conflito em seus trabalhos com o software. O Participante 2, por exemplo, idealizou uma ferramenta para plotar, anotar e exportar gráficos facilmente – ações que consomem tempo excessivo através das ferramentas que de fato usa. Por outro lado, o software do Participante 3 permite a usuários controlarem a reprodução de gráficos simplificados – uma abordagem minimalista mais eficaz do que métodos de visualização mais complexos. Trata-se de uma observação interessante, pois sugere que formas de representação visualmente impactantes inspiradas por jogos eletrônicos poderiam ter, em alguns casos, um efeito negativo sobre a visualização dos dados.

#### 4.2.5.2

##### **Quanto ao método e suas ferramentas**

Ao encorajar os participantes a pensarem em suas pesquisas nos termos propostos pelas cartas diretrizes, o método pareceu bem-sucedido em estimular e valorizar uma visão sistêmica através da qual considerar as relações entre usuário e software, buscando deixar claro como interação entre os dois deveria propiciar resultados satisfatórios e experiências mais agradáveis. Além disso, enquanto abordagem à gamificação, o método foi bem-sucedido em levar em consideração as características da atividade a que se destina.

Incluir um participante que não era usuário, mas sim desenvolvedor de um software, pode parecer contraditório, dada a natureza da pesquisa. Entretanto, as fronteiras entre usuários e desenvolvedores podem ser indefinidas no software científico e, neste caso, esperava-se tomar vantagem desta proximidade, instigando o desenvolvedor a adotar a perspectiva de um usuário. Neste aspecto, a experiência não foi bem-sucedida, e os dois papéis não se mostraram intercambiáveis. Usuários se mostraram, de fato, mais eficazes em identificar problemas. Ainda assim, o

método serviu àquele participante, e à pesquisa, como ferramenta para a descrição dos desafios, decisões de design e funcionalidades do software.

Os Participantes 1 e 2, que se qualificam como usuários, foram bem-sucedidos em idealizar soluções para seus problemas. O Participante 1 projetou uma única funcionalidade que desejaria ter no software que usa em seu trabalho – uma solução realizável, consideradas as ferramentas de desenvolvimento de *plug-ins* oferecidas pelo software. No caso deste participante, as cartas ilustradas tornaram as questões claras e tangíveis, fornecendo inspiração para potenciais soluções. O Participante 1 se utilizou dos conceitos expressos nas cartas diretrizes para ditar claramente como o usuário deveria completar uma tarefa, como o software deveria se comportar e responder, e como a solução projetada providenciaria resultados e experiências mais satisfatórias. É interessante notar que, embora os conceitos expressos pelas cartas diretrizes se aplicam a jogos em suas totalidades, elas foram bem-sucedidas em oferecer um quadro para a elaboração de soluções referentes a uma tarefa específica.

O Participante 2, por sua vez, idealizou uma solução complexa que demandaria muito mais esforço e conhecimento técnico para ser desenvolvida – ou mesmo prototipada. No caso deste participante, as cartas ilustradas indicaram os objetivos e esforços mais importantes de seu trabalho como pesquisador (ler, escrever, solucionar problemas, imaginar e visualizar). Foi ao redor destes esforços e objetivos que o participante imaginou várias das funcionalidades incluídas em seu rascunho – principalmente as ferramentas para anotação e exportação de imagens – sem, no entanto, descrevê-las em mais detalhes.

Dado o consenso em privilegiar o design iterativo e a implementação de pequenas alterações no software científico, concentra-se em uma única funcionalidade, tal como fez o Participante 1, parece ser o uso mais prático e produtivo do método proposto. Ainda assim, a conceituação de soluções mais amplas e ambiciosas, tais como a do Participante 2, poderia funcionar como ponto de partida para o design de produtos futuros.

#### 4.2.6 Conclusão

Ao desenvolver um método para gamificação voltado ao software científico baseado em métodos de co-design, este estudo buscou cumprir um dos objetivos desta tese, além de expandir a discussão a respeito da similaridade entre ciência e jogo – particularmente considerando o papel do software e suas interfaces.

Através do método proposto, usuários foram encorajados a pensar em suas pesquisas e trabalho em termos de um jogo, obtendo clareza a respeito de seus objetivos e em como atingi-los de modo satisfatório através do sistema de regras em que estão inseridos. Esta abordagem reflete, de várias maneiras, as semelhanças entre ciência e jogos indicadas pelo quadro teórico de referência da tese – algumas das quais se tornaram evidentes através dos estudos de caso. Assim como o jogo, o trabalho científico é permeado de desafios, objetivos, métricas, fases, regras e diversão. Este jogo, no entanto, não é jogado somente através do software científico, se estendendo a outras fases de pesquisa, anteriores e posteriores a seu uso, em um processo cíclico, às vezes errático, envolvendo literatura especializada, colegas, supervisores, equipamentos, instituições, verbas de pesquisa, e assim por diante. Além disso, a prática científica é complexa, aberta a experimentação e difícil de avaliar. Lançar mão de elementos de design de jogos tradicionais para indicar progresso e medir performance seria restritivo e artificial, desvalorizando as qualidades lúdicas verdadeiramente presentes na ciência.

Os estudos de caso realizados sugerem que o método proposto é efetivo na comunicação de problemas e ideação de soluções através de uma perspectiva baseada no jogo, tal como descrita pelas cartas diretrizes. Por outro lado, as cartas ilustradas se mostraram úteis, ajudando participantes a estabelecer prioridades, descrever desafios e conceber soluções, tornando questões abstratas mais palpáveis. De fato, as cartas se mostraram capazes de facilitar o processo de design no estímulo e estruturação de ideias; na organização do pensamento; na percepção, compreensão e verbalização de problemas. Como indicado por outras pesquisas em design para software científico (Rampersad et al, 2017) – e também pela experiência com o SiVIEP, relatada na primeira metade deste capítulo – protótipos de baixa fidelidade desempenham ótimo papel na geração de ideias, enquanto protótipos de alta fidelidade são mais eficientes em comunicar a interatividade pretendida.



Finalmente, é importante ressaltar que o método se mostrou particularmente adequado à abordagem a problemas localizados e a processos de design iterativo, estando assim de acordo com aquelas que são consideradas melhores práticas no desenvolvimento de software científico.

### 4.3

#### **Análise comparativa e considerações**

As experiências relatadas nas seções anteriores opõem duas abordagens distintas à gamificação. A primeira delas buscou ativamente por formas de interação e interfaces em jogos eletrônicos, numa tentativa de incorporar esses padrões ao software científico. A segunda, ao contrário, concentrou seus esforços em identificar aspectos lúdicos inerentes à ciência e seus suportes computacionais que que pudessem ser materializados em suas interfaces. Embora não haja motivos para considerar tais abordagens como mutuamente excludentes, é evidente a tensão entre as duas perspectivas.

No primeiro caso, o processo de design foi orientado a uma busca por elementos novos, excitantes, que se destacassem não somente por sua funcionalidade, mas também pelo deleite e plasticidade associados ao seu meio de origem, os jogos eletrônicos. No entanto, a inclusão destes elementos no software científico foi prejudicada por motivos técnicos e gerenciais, de caráter circunstancial. No caso, a dificuldade em incorporar os elementos de design pretendidos resultaria em um alto custo de implementação, dada as limitações das ferramentas de desenvolvimento adotadas e, também, dos prazos e planos de desenvolvimento. Neste ponto, é revelador que a maior parte dos exemplos de software científico gamificados identificados pela tese sejam, de fato, concebidos para desenvolvimento através de ferramentas e tecnologias originalmente dedicadas aos video games. Em alguns casos, tais produtos são, na verdade, explorações no uso destas tecnologias em prol da visualização científica – ou seja, parte-se das tecnologias dos jogos, para as quais recursos sofisticados de interação são nativos, e desenvolve-se funcionalidades desejadas a partir daí. Em casos como estes, os critérios sugeridos para a pesquisa a elementos de design (por proximidade temática, funcionalidade pretendida ou uso de tecnologia) são altamente recomendáveis. Entretanto, o caminho sugerido pelo primeiro caso relatado –

implementação de interatividade típica de jogos através de ferramentas e linguagens não necessariamente dedicadas a este fim – é bem mais difícil de ser percorrido. O que não quer dizer que tal abordagem seja impraticável: a implementação de funcionalidades inspiradas em jogos pode não ser mais difícil do que o usual. No caso do SiVIEP, por exemplo, o controle de navegação espaço-temporal – se implementado isoladamente – não teria consumido recursos excessivos de desenvolvimento. O problema neste caso – como na totalidade das funcionalidades idealizadas para aquele projeto – é que, ainda que pudesse ter sido útil, tal recurso não foi percebido como útil o suficiente para ser de fato implementado, principalmente com vista às necessidades e pressões de desenvolvimento do software naquele momento. A questão central – evidente quando se considera as mudanças de escopo e objetivos daquele software ao longo da pesquisa – é que o custo de implementação de funcionalidades é multiplicado pela falta de definição e clareza nos objetivos do software e seus usuários. Se, como afirmam Morschheuser et al. (2018), o design de gamificação deve ser iniciado por um levantamento de requisitos e necessidades dos usuários, o software científico tem um grande obstáculo a gamificação.

A segunda experiência relatada neste capítulo propôs superar este obstáculo pela inversão: se, por um lado, o levantamento de requisitos e objetivos é essencial no design de gamificação, no software científico, é a gamificação que será útil no levantamento de requisitos e objetivos. O método proposto oferece, através de sua dinâmica e suas ferramentas, formas de se especificarem os objetivos, métodos e resultados desejados pelos usuários, tomando a ciência pelo jogo que ela realmente é. Neste caso, não são obrigatórias as interfaces sofisticadas dos video games, e nem os elementos de design de jogos tal como usualmente definidos. São os elementos do uso e desenvolvimento de software, além das manifestações do jogo na ciência, que orientam o processo de design centrado no usuário. Embora o método possa dar vazão a ideias ambiciosas e de difícil concretização – como mostra o estudo de caso em matemática física – ele também facilita e valoriza soluções factíveis, voltadas a problemas pontuais – como no caso da física de matéria condensada. Em ambos os casos, a adoção do jogo como modelo para o software e pesquisa científica contribuiu para que se esclarecessem as necessidades dos usuários. Assim, as interfaces desenhadas seriam meios para suprir tais necessidades – e não, elas mesmas, ideais a serem alcançados. Neste caso, propostas de difícil realização

podem ser mais facilmente abandonadas em prol de outras, mais factíveis. Para este fim, seria interessante o envolvimento dos responsáveis pelo desenvolvimento, durante ou após a dinâmica, para orientação quanto às possibilidades reais de implementação.

Interfaces inspiradas em video games, como as projetadas durante a pesquisa-ação, têm grande potencial de aplicação em uso científico – particularmente em aplicações envolvendo inovação e o uso de novas tecnologias em visualização científica. As formas de interatividade dos jogos eletrônicos oferecem soluções criativas e prazerosas e, ao que indica sua popularidade, podem vir a se tornar o modelo a ser seguido pelo software, científico ou não. Entretanto, a exemplo do que Kuhn declarou sobre a evolução da ciência, o design e desenvolvimento de software científico é um processo “movido pelo que já passou, e não puxado pelo que virá”. Neste caso, o método de co-design se mostra como mais adequado em sua dedicação às características da ciência, aos estados atuais de desenvolvimento de software científico e aos objetivos dos cientistas.

## 5

### Conclusão e considerações finais

Esta tese recorreu a diferentes métodos de pesquisa e campos do conhecimento para desenvolver um método de gamificação adequado ao software científico. Dentro dos objetivos propostos, o método para gamificação apresentado pela segunda experiência relatada – e tanto quanto sabemos, o primeiro dedicado àquele tipo de software – consolida o resultado da pesquisa, incorporando (1) o conhecimento reunido sobre as condições de desenvolvimento e uso do software científico, (2) as características lúdicas da ciência, e (3) os métodos de design de interfaces e gamificação mais apropriados. Estes três fatores foram articulados através um modelo do software científico (e da ciência) baseado nas características do jogo tal como definidas por Jesper Juul. Como resultado, obteve-se uma nova abordagem à gamificação, que se distingue das demais abordagens investigadas por motivos que merecem ser apontados. Ao longo das próximas seções, serão discutidas as diferentes abordagens à gamificação, contribuições, limitações, sugestões para pesquisas futuras e considerações finais a respeito do desenvolvimento da tese.

#### 5.1

##### Abordagens à gamificação

A gamificação, como explicado no capítulo 2, busca proporcionar experiências mais prazerosas a partir da transposição de elementos de design de jogos às suas atividades alvo. Em seu formato mais típico, ela lança mão de recursos como sistema de pontuação, medidores de progresso, premiações e concursos para estimular a participação – em intensidade e frequência – dos seus usuários. Trata-se de uma abordagem reducionista que, no entanto, poderia ser candidata à aplicação nas ciências – afinal, as noções de competição e busca por resultados não são estranhas ao meio científico, sendo a corrida de James W. Watson pela descoberta da estrutura do DNA – e pelo Prêmio Nobel – o melhor exemplo dentre os discutidos nesta tese. Entretanto, mesmo que a ênfase em conquistas, pontuações e condições de vitória fosse alinhada aos interesses dos cientistas – e que essa

abordagem à gamificação fosse considerada produtiva e ética – ainda assim sua implementação no software científico seria arriscada ou impraticável, a não ser a partir de critérios arbitrários, pela dificuldade em se julgar o progresso e valor do trabalho em andamento do cientista. Portanto, o descarte desta abordagem, além de apoiado por uma visão crítica da gamificação, tem caráter prático.

Uma segunda abordagem à gamificação se baseia na incorporação de formas de interatividade e estéticas próximas às dos jogos. Trata-se de uma perspectiva que pode ser interessante ao software científico: manifestações culturais – como as artes, a literatura, os video games – podem revelar novas visões e apontar novos caminhos e tecnologias para a ciência. Em *Flatland- a romance of many dimensions*, por exemplo, Edwin Abbott descreveu um mundo bidimensional em termos que anteciparam a visualização transversal rotineiramente empregada em tomografias (Banchoff, 2015). No entanto, estas novas formas de visualizar objetos só se materializaram muito após 1884, ano de publicação da obra de Abbott. A distância entre as tecnologias dos jogos e o software científico é, claro, menos dramática – especialmente quando as primeiras dão suporte ao segundo. Neste caso em particular, a tese apresenta, em sua revisão de literatura, alguns exemplos de aplicações bem-sucedidas e, para aplicações do gênero, recomenda os critérios de pesquisa a design de jogos desenvolvidos durante a pesquisa-ação relatada. No entanto, aquela mesma experiência demonstra que a adoção de interatividade sofisticada, além de difícil e custosa implementação em software científico, corre o risco de ignorar os objetivos da pesquisa a que se destina.

Apesar de distintas entre si, as duas abordagens descritas são baseadas na transposição, ou ainda, na importação de elementos de design de jogos para outras atividades. O método desenvolvido pela tese, no entanto, se baseia na identificação de elementos lúdicos inerentes à ciência. Além disso, o método não recorre à gamificação somente para propor soluções a problemas, mas para defini-los. A gamificação, neste caso, é um *meio*, e não um fim. De fato, quando examinadas, nenhuma das soluções projetadas pelos participantes dos estudos de caso apresenta interfaces inspiradas em video games, e tampouco apresentam elementos de design de jogos. Trata-se de um reflexo da proposta da pesquisa – repetida ao longo da tese – de se utilizar o jogo como base para um modelo conceitual para a ciência. Neste sentido, a explicação de Ian Hacking pode ser esclarecedora:

"Eu poderia fazer um modelo da economia a partir de roldanas, alavancas, rolamentos e pesos. Cada redução em um peso  $M$  (a 'oferta de dinheiro') produz uma redução no ângulo  $I$  (a 'taxa de inflação') e um aumento no número  $N$  de roldanas no recipiente (o número de trabalhadores desempregados). Teríamos as variáveis de entrada e saída corretas, mas ninguém está sugerindo que isso é o que a economia é" (Hacking, 1983, p.22, tradução do autor)

Métodos de gamificação tradicionais tentam reforçar mecanismos e engrenagens – às vezes artificialmente – materializando-os como elementos de design de jogos que aplicam às suas atividades-alvo. O método proposto, ao contrário, compreende a dinâmica do sistema sem tentar torná-lo demasiadamente literal. Tal como a persuasão na literatura científica, o jogo no software científico parece melhor aplicado quando implícito. Esta abordagem, no entanto, dá margem à seguinte pergunta: o método proposto nesta tese é, de fato, gamificação? A resposta para esta questão passa pela interpretação do termo gamificação tal como apresentada pela tese. Se, por um lado, a visão predominante se apoia na definição de Deterding et al. para condicionar a gamificação à adoção de elementos estruturais do design de jogos, por outro, a tese estende o termo a iniciativas baseadas na adoção de elementos estéticos e tecnologias oriundas dos jogos. Neste caso, o método proposto reflete ainda outra abordagem: a adoção de elementos oriundos das teorias dos estudos de jogos na interpretação das atividades alvo. Desta forma, o método depende de certa liberdade interpretativa para ser considerado gamificação – fato que não enfraquece a tese e seus resultados, já que esta liberdade se mostrou essencial no cumprimento dos objetivos estabelecidos pela tese.

## 5.2 Contribuições

Dentre as contribuições oferecidas pela tese, destacam-se, principalmente:

- Os modelos esquemáticos das atividades de pesquisa e trabalho em software científico – baseados na definição clássica de Juul e apoiado na revisão de literatura e pesquisa bibliográfica – tais como apresentados nas seções 3.6.4, e 4.2.1.1.
- O método para co-design de interfaces de software científico tal como proposto no capítulo 4, seção 4.2 – com destaque para uma nova abordagem à gamificação integrada a processos de co-design.

Além disso, são também dignas de menção, enquanto contribuições da tese:

- A diagramação da definição de Juul como base para visões sistêmicas de atividades-alvo em processos de gamificação, tal como apresentada no capítulo 3, seção 3.2.
- Os critérios de pesquisa a jogos eletrônicos como fonte de inspiração para software científico, tal como apresentados no capítulo 4, seção 4.1.3.2.
- A discussão acerca da similaridade entre jogos e ciência apresentada no capítulo 3 – e novas observações a este respeito feitas a partir dos resultados dos estudos de caso, tal como apresentadas na seção 4.2.5.1.

### 5.3 Limitações

Como seria de se esperar, dado o escopo e diversidade metodológica do estudo, a pesquisa apresenta algumas limitações – tanto de cunho teórico quanto de caráter experimental. Em primeiro lugar, a comparação entre ciência e jogo estabelecida no capítulo 3, que serviu de base para o modelo construído, apresenta como limitação uma descrição pouco aprofundada dos valores da ciência capazes de merecer tamanha dedicação, empenho e apego por parte dos cientistas, mesmo frente à quantidade de esforço, reveses e resultados negativos que possam decorrer de seus trabalhos. Embora tais características sejam possivelmente secundárias, ou mesmo dispensáveis, na construção do modelo proposto, ainda assim poderiam ser consideradas merecedoras de maior atenção no que se refere a uma extensa comparação entre jogo e ciência. Atribuímos esta limitação, no entanto, à adoção da definição de Juul (2005) como ponto de partida para a construção de modelo. De fato, com a inclusão do *processo* da pesquisa científica como objeto de apego (em oposição a apenas *resultados*, tal como na definição original de Juul) tentou-se justamente contornar tal limitação – um esforço que, embora tenha levantado pontos importantes para o modelo e método desenvolvidos, pareceu limitado em traduzir o espaço que a ciência ocupa e o papel que ela desempenha, na vida daqueles que a praticam – papel este, poderíamos argumentar, semelhante àquele desempenhado pelo jogo na vida de seus entusiastas.

Em segundo lugar, no que se refere a limitações experimentais, teria sido desejável a realização de mais experimentos envolvendo o método proposto, particularmente tal como indicado na próxima subseção. Entretanto, a dificuldade no recrutamento de participantes mostrou-se como obstáculo a uma abordagem quantitativa à pesquisa em campo.

## 5.4

### Pesquisas futuras

Como possíveis desdobramentos da pesquisa, gostaríamos de sugerir:

- Aplicação do modelo sistêmico baseado na definição de Juul a outras atividades alvo.
- Exploração das dinâmicas de co-design em grupos multidisciplinares.
- Variações e refino do método em seus materiais e dinâmicas.
- Integração do método proposto a metodologias de desenvolvimento de software.

Além disso, no que se refere ao futuro do método desenvolvido, esperamos encontrar novas oportunidades de utilizá-lo no design de interfaces para software científico e, possivelmente, atividades correlatas (educação, ciência cidadã, etc.). Esperamos, ainda, que através do uso e aperfeiçoamento deste método, possam ser gerados novos *insights* a respeito do uso do software científico e, também, das semelhanças entre jogo e ciência.

## 5.5

### Considerações finais

Considerados, portanto, os resultados da tese, gostaríamos de tecer algumas observações a respeito de seu desenvolvimento. Mais particularmente, gostaríamos de discutir a evolução da tese sob a perspectiva de seu quadro teórico de referências. Como seria de se esperar, uma pesquisa que investigue a pesquisa científica está sujeita a ser continuamente influenciada pelas informações colhidas ao longo do processo. No caso desta tese, tal influência se refletiu, sobretudo, nas ações do pesquisador. O autor muitas vezes se viu planejando, conduzindo e analisando seu trabalho através do mosaico elaborado pela própria pesquisa. Neste caso, o estudo



da história e filosofia da ciência foi ponto de partida para reflexões e tomadas de decisão que influenciaram em muito os rumos da tese.

Em primeiro lugar, quanto à já mencionada dificuldade em se classificar como gamificação o fruto da pesquisa, esta reflete, dentre vários fatores, uma busca por um posicionamento novo e original. Da intenção inicial em se pesquisar o tema (ainda na primeira década dos anos 2000) ao fim do trabalho de pesquisa, o campo da gamificação se expandiu e se fortaleceu – mas sua novidade se esvaiu. Se o dever do cientista, como propôs Feyerabend, é mesmo “tornar mais forte o caso mais fraco” (1993, p.21, tradução do autor), parece apropriada a apresentação de uma proposta que *enfraqueça* abordagens convencionais de gamificação favorecendo uma perspectiva que, de certo modo, a incorpore de forma mais sutil no processo de design. Por outro lado, convém observar que a tese busca, também, fortalecer a gamificação – já combatida por uma visão crítica estabelecida – ao propor uma nova abordagem que responda a tais críticas. Em retrospecto, mesmo a decisão de se buscar na história e filosofia da ciência – disciplinas cuja aplicação em outros campos não é costumeira – instrumentos para a construção de métodos de design parece, também, uma busca por originalidade.

Um segundo ponto a ser ressaltado foi a busca por equilíbrio entre dois aspectos da tese: um teórico, presente no capítulo 3, e outro, experimental, mais pronunciado no capítulo 4. A tensão entre esses dois conjuntos pode ser explicada a partir da perspectiva apresentada por Galison (1987): por um lado, pretende-se que o experimento seja uma demonstração da teoria. Por outro, sabe-se que as preocupações envolvidas nos experimentos, por vezes, passam longe das questões teóricas que supostamente os guiam. De fato, no caso desta tese, as diferenças na abordagem a aspectos teóricos e experimentais poderiam sugerir um abismo entre os dois. No entanto, buscamos tornar a relação entre as duas partes clara a partir da discussão dos estudos de caso em co-design, onde o quadro teórico de referência dá suporte às interpretações. Além disto, é importante ressaltar a importância da teoria naquele que é um esforço fundamental na construção da tese: a definição de seus métodos e modelos. Neste caso, destaca-se não apenas a elaboração do modelo ciência-jogo e a identificação de elementos sujeitos à sua dinâmica, como também a transformação desta dinâmica em procedimentos experimentais – os quais vieram a se tornar o método proposto. Novamente, a pesquisa bibliográfica foi essencial

para que se tornasse clara, para o autor, a necessidade de se estabelecer e discutir as relações entre questões experimentais e teóricas.

Em terceiro lugar, o autor reconhece, tal como descreve Knorr Cetina (1981), o valor da exploração de saberes, técnicas, recursos e ideias disponíveis no ambiente de pesquisa. Neste caso, muito se deve à inserção do autor em diversos ambientes (Instituto Tecgraf, Laboratório de Arte Eletrônica, Escola de Design da Universidade de Leeds, etc.) e sua proximidade com profissionais, professores e pesquisadores de diversos campos. Tal inserção esteve intimamente ligada a outro aspecto destacado por Knorr Cetina: o estabelecimento de alianças com outros pesquisadores – muitas vezes reforçada pela busca daqueles que são, para Latour & Woolgar (1986), os maiores objetivos no ambiente de pesquisa: a elaboração e publicação de artigos. Neste sentido, o autor buscou construir redes de colaborações que pudessem contribuir para o avanço da a pesquisa e, simultaneamente, informá-lo de outras pesquisas para as quais ele mesmo pudesse contribuir.

Em quarto lugar, o quadro teórico de referência influenciou, em diversos momentos, a direção dos esforços do autor – o mais nítido exemplo sendo a construção de modelos híbridos, inspirados no relato de Nersessian (2008), que contribuíssem na transposição dos elementos do jogo ao software científico.

Finalmente, os autores de referência ofereceram, através de suas obras, conselhos e respostas às dúvidas do autor quanto a estar fazendo, de fato, pesquisa. Do oportunismo de Feyerabend (2010) à persuasão de Latour (1987), da proatividade de Popper (2005) à solução de problemas de Kuhn (2012), dos modelos de Nersessian (2008) à empatia de Keller (1985), ao longo deste trabalho, foi possibilitado ao autor espelhar-se na história e filosofia da ciência para reconhecer-se cientista, compreendendo, simultaneamente ao desenrolar do processo, o sentido de suas ações, tal como Feynman buscando as regras do universo em um jogo em andamento.

## 6 Referências Bibliográficas

ACADEMIA REAL DE CIÊNCIAS DA SUÉCIA. **The Nobel Prize in Physics 2010**. 2010. Disponível em: <[https://www.nobelprize.org/nobel\\_prizes/physics/laureates/2010/press.pdf](https://www.nobelprize.org/nobel_prizes/physics/laureates/2010/press.pdf)>. Acesso em: 23 abr. 2017.

ACKROYD, K. S. et al. Scientific Software Development at a Research Facility. In: **IEEE Software**, v. 25, n. 4, p. 44-51. 2008.

AFGAN, E. et al. Building and provisioning bioinformatics environments on public and private Clouds. In: 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). **Anais...**, 2015. p.223–228.

\_\_\_\_\_. The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2016 update. In: **Nucleic Acids Research**. v.44, p. w3-w10. 2016.

AHALT, S. et al. Water Science Software Institute: Agile and open source scientific software development. In: **Computing in Science & Engineering**, v. 16, n. 3, p. 18-26. 2014.

AHMED, Z.; ZEESHAN, S. Cultivating Software Solutions Development in the Scientific Academia. In: **Recent Patents on Computer Science**, v. 7, n. 1, p. 54-66. 2014.

AHMED, Z.; SAMAN, Z.; DANDEKAR, T. Developing sustainable software solutions for bioinformatics by the “Butterfly” paradigm. In: **F1000Research**, v. 3, 2014. p. 71 Disponível em: <<http://www.ncbi.nlm.nih.gov/pubmed/25383181>>. Acesso em: 17 out. 2018.

ALAČ, M. Digital Scientific Visuals as Fields for Interaction. In: COOPMAN, C; VERTESI, J.; LYNCH, M.; WOOLGAR, S. (Org.) **Representation in Scientific Practice Revisited**. Cambridge, MA: The MIT Press, 2014.

ANUPAM, V.; BAJAJ, C. L. Collaborative multimedia scientific design in SHAstra. In: Proceedings of the first ACM international conference on Multimedia - MULTIMEDIA '93. **Anais...** New York, New York, USA: ACM Press. 1993. p.447–456.

ARAGON, C. et al. Using visual analytics to maintain situation awareness in astrophysics. In: 2008 IEEE Symposium on Visual Analytics Science and Technology. **Anais...** 2008. p.27–34.

\_\_\_\_\_. The changing face of digital science: new practices in scientific collaborations. In: **Proceedings of the 27th international conference extended abstracts on Human factors in computing systems - CHI EA '09**. New York, New York, USA: ACM Press, p.4819. 2009.

ATKINS, D. E.; PHILLIPS, R. L. A user perspective on computer workstation integration. In: **Circuits and Devices Magazine**, IEEE, v. 2, n. 4, p. 22-31.1986.

AYDIN, S.; SCHNABEL, M. A. A Survey on the Visual Communication Skills of BIM Tools. In: **Rethinking Comprehensive Design: Speculative Counterculture, Proceedings of the 19th International Conference on Computer- Aided Architectural Design Research in Asia**. CAADRIA. 2014.

BACHELARD, G. O Novo Espírito Científico. In: **Os Pensadores – Bachelard**. Tradução de Remberto Francisco Kuhnen. São Paulo, SP. Editora Abril Cultural. 1978a.

\_\_\_\_\_. A Filosofia do Não. In: **Os Pensadores – Bachelard**. Tradução de Joaquim José Moura Ramos. São Paulo, SP. Editora Abril Cultural. 1978b.

\_\_\_\_\_. **A Formação do Espírito Científico: Contribuição para uma Psicanálise do Conhecimento**. Tradução de Estela dos Santos Abreu. Rio de Janeiro, RJ. Contraponto. 1996.

BANCHOFF, T. Flatland: an introduction. In: ABBOTT, E.A. **Flatland – a romance of many dimensions**. Princeton, NJ, Princeton University Press. 2015.

BASIL, V. R. et al. Understanding the high-performance-computing community: A software engineer's perspective. In: **IEEE software**, n. 4, p. 29-36. 2008.

BATEMAN, C. Playing work, or gamification as stultification. In: **Information, Communication & Society**, v. 21, n. 9. p. 1193–1203. 2018.

BAXTER, S. M. et al. Scientific software development is not an oxymoron. In: **PLoS Computational Biology**, v. 2, n. 9, p. e87. 2006.

BEG, M.; PEPPER, R. A.; FANGOHR, H. **User interfaces for computational science: a domain specific language for OOMMF embedded in Python.** 2016 Disponível em: <<http://arxiv.org/abs/1609.07432>>. Acesso em: 23 abr. 2017.

BELSKY, A. et al. New developments in the Inorganic Crystal Structure Database (ICSD): accessibility in support of materials research and design. In: **Acta Crystallographica Section B: Structural Science**, v. 58, n. 3, p. 364-369. 2002.

BERGMANN, S.; DEMUTH, B.; SANDER, V. A Web Framework for Workflow Submission and Monitoring via UNICORE 6 based on Distributable Scientific Workflow Templates. In: **Schriften des Forschungszentrums Jülich IAS Series** Volume 9, p. 45. 2011.

BERGMANN, T., BALZER, M., HOPP, T., DE KAMP, T. van, KOPMANN, A., JEROME, N. T., and ZAPF, M. 'Inspiration from VR Gaming Technology: Deep Immersion and Realistic Interaction for Scientific Visualization'. In: **IVAPP 2017: 8th International Conference on Information Visualization Theory and Applications**, Porto, Portugal, 27 Feb – 1 mar 2017. Vol.: 3. Ed.: L. Linsen. SCITEPRESS, Setúbal. p. 330–334. 2017.

BILMES, J. User-friendly neural-net design [software reviews]. In: **Spectrum**, IEEE, v. 33, n. 2, p. 63.1996.

BITTER, I. et al. Comparison of four freely available frameworks for image processing and visualization that use ITK. In: **IEEE Transactions on Visualization and Computer Graphics**. v. 13, n. 3, p. 483-493. 2007.

BJORK, S.; HOLOPAINEN, J. **Patterns in game design**. Hingham, Mass. Charles River Media, 2005.

BLANKENBERG, D.; TAYLOR, J.; NEKRUTENKO, A. Online resources for genomic analysis using high-throughput sequencing. In: **Cold Spring Harbor Protocols**, v. 2015, n. 4, p. pdb-top083667, 2015.

BOEYKENS, S. Using 3D Design software, BIM and game engines for architectural historical reconstruction. In: **Designing Together-CAADfutures 2011. Anais...** 2011.

BOGOST, I. **Unit operations: An approach to videogame criticism**. Cambridge, Massachusetts. MIT Press, 2006.

\_\_\_\_\_. Why Gamification is Bullshit. In: WALZ, S. P.; DETERDING. (Org.) **The Gameful World: Approaches, Issues, Applications**. Cambridge, Massachusetts: MIT Press, p.65. 2015.

BOWSER, A.; HANSEN, D.; PREECE, J. Gamifying citizen science: Lessons and future directions. In: **Workshop on Designing Gamification: Creating Gameful and Playful Experiences**. 2013

BOWSER, A. et al. Gamifying citizen science: a study of two user groups. In: **Proceedings of the companion publication of the 17th ACM conference on Computer supported cooperative work & social computing**. 2014

BREBNER, P. C. Software process improvement by example. (SPIE) In: **Proceedings from 1998 International Conference Software Engineering: Education and Practice**. 1998.

BREITWIESER, L. et al. **The BioDynaMo Project: Creating a Platform for Large-Scale Reproducible Biological Simulations**. arXiv preprint arXiv:1608.04967, 2016.

BROOKES, E. H. et al. The GenApp framework integrated with Airavata for managed compute resource submissions. In: **Concurrency and Computation: Practice and Experience**. 2015.

BROUWER-JANSE, M. D. AI technologies for user interfaces: knowledge-based front-ends. In: **IEE Colloquium on AI in the User Interface** .1990.

BROWN, P. An Emergent Paradigm. In: **Periphery**, n. 29, 1996.

BUNUS, P. A simulation and decision framework for selection of numerical solvers in Scientific Computing. In: **Proceedings of the 39th annual Symposium on Simulation**. 2006.

CAILLOIS, R. **Man, play, and games**. Urbana: University of Illinois Press, 2001.

CAROLI, E. et al. Characterization of a CdTe microstrip detector as a hard X ray focal plane prototype. IEEE Symposium Conference Record Nuclear Science 2004. **Anais...** v. 7. 2004. p.4378–4382.

CARTWRIGHT, N. **How the Laws of Physics Lie**. Oxford. Claredon Press, 1983.

CARVALHO, F. G; TREVISAN, D. G; RAPOSO, A. Toward the design of transitional interfaces: an exploratory study on a semi-immersive hybrid user interface. In: **Virtual Reality**, v. 16, n. 4, p. 271-288. 2009.

CHAM, J., COSTA, K. UX Design: maximising the value of scientific software in life science R&D. In: **Drug Discovery World** 18, p. 58. 2017.

CHAMBERS, C. Helping end users find and fix performance issues in visual dataflow code. In: **IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), 2013**. 2013.

CHANCELIER, J.-P.; LAPEYRE, B.; LELONG, J. Using Premia and Nsp for constructing a risk management benchmark for testing parallel architecture. In: **Concurrency and Computation: Practice and Experience**, v. 26, n. 9, p. 1654-1665. 2014.

CHANDLER, T. et al. Immersive Analytics. In: **2015 Big Data Visual Analytics (BDVA)**. p. 1-8. 2015.

CHEN, Y.; FU, T. The use of World Wide Web browsers to create graphical user interfaces (GUIs) for crystallographic software: a GUI implementation for X-PLOR. In: **Journal of applied crystallography**, v. 29, n. 2, p. 202-203. 1996.

CHEN, C.; ZHANG, J.; VOGLEY, M. S. Reflections on the Interdisciplinary Collaborative Design of Mapping the Universe. In: **Human-Computer Interaction. Interacting in Various Application Domains**. Springer, p. 693-702. 2009.

CHOI, Y. J. et al. Labgrid: Integrated problem solving environment system for high throughput computing. In: **Second IEEE International Conference on e-Science and Grid Computing, 2006. e-Science'06**. 2006.

CHURCH, P. et al. Toward exposing and accessing HPC applications in a SaaS cloud. In: **IEEE 19th International Conference on Web Services (ICWS), 2012**. 2012.

CIMIANO, P.; SAGERER, I. G. **CONQUAIRE: Continuous quality control for research data to ensure reproducibility: an institutional approach**. 2015.

CLANCEY W. J. Becoming a Rover. In: **Simulation and its Discontents**, MIT Press, Cambridge, MA, 2009.

CLARK, A. M. Cheminformatics: Mobile Workflows and Data Sources. In: **The Future of the History of Chemical Information**. pp.237-253. 2014.

COHEN, J. et al. Nekkcloud: A software environment for high-order finite element analysis on clusters and clouds. In: **IEEE International Conference on Cluster Computing (CLUSTER) 2013**. 2013.

COOK, D. **The Chemistry of Game Design**. Disponível em: <[https://www.gamasutra.com/view/feature/129948/the\\_chemistry\\_of\\_game\\_design.php](https://www.gamasutra.com/view/feature/129948/the_chemistry_of_game_design.php)>. Acesso em: 06 jun. 2018. 2007.

COOPER, S. Massively Multiplayer Research: Gamification and (Citizen) Science. In: WALZ, S. P.; DETERDING, (Org.) **The Gameful World: Approaches, Issues, Applications**, Cambridge, Massachusetts: MIT Press, p. 487. 2015.

COOPER, S. et al. The challenge of designing scientific discovery games. In: **Proceedings of the Fifth international Conference on the Foundations of Digital Games**. ACM, 2010

COOPER, S.; KHATIB, F.; BAKER, D. Increasing public involvement in structural biology. In: **Structure**, v. 21, n. 9, p. 1482-4. 2013.



COOPMANS, C. et al. (Org.) **Representation in Scientific Practice Revisited**. Cambridge, MA: The MIT Press, 2014.

COOPMANS, C. Visual Analytics as Artful Revelation. In: COOPMAN, C; VERTESI, J.; LYNCH, M.; WOOLGAR, S. (Org.) **Representation in Scientific Practice Revisited**. Cambridge, MA: The MIT Press, 2014.

CORT, G. et al. A development methodology for scientific software. In: **IEEE Transactions on Nuclear Science**. v. 32, n. 4, p. 1439-1443. 1985.

COULTER, E. et al. Cyberinfrastructure as a platform to facilitate effective collaboration between institutions and support collaboratories. In: **Proceedings of the 2016 ACM on SIGUCCS Annual Conference**. 2016

CRABTREE, C. et al. An empirical characterization of scientific software development projects according to the Boehm and Turner model: A progress report. In: **ICSE Workshop on Software Engineering for Computational Science and Engineering, 2009. SECSE'09**. 2009

CSIKSZENTMIHALYI, M. **Flow: The Psychology of Optimal Experience**. Nova Iorque: Harper Perennial, 2008.

\_\_\_\_\_. **Creativity: the psychology of discovery and invention**. Nova Iorque. Harper Perennial, 2013.

CUNNINGHAM, A. Getting the game right: Some plain words on the identity and invention of science. In: **Studies in History and Philosophy of Science Part A** 19, No.3, p.365-389.1989.

CURTIS, R. E.; KINNAIRD, P.; XING, E. P. GenAMap: visualization strategies for structured association mapping. In: **IEEE Symposium on Biological Data Visualization (BioVis) 2011**. 2011.

DANILUK, A. Visual modeling for scientific software architecture design. A practical approach. In: **Computer Physics Communications**, v. 183, n. 2, p. 213-230. 2012.

DAREJEH, A.; SALIM, S. S. Gamification solutions to enhance software user engagement—a systematic review. **International Journal of Human-Computer Interaction**, v. 32, n. 8, p. 613-642, 2016.

DASTON, L.; GALISON, P. **Objectivity**. New York, NY: Zone Books, 2010.

DE MATOS, P. et al. The Enzyme Portal: a case study in applying user-centred design methods in bioinformatics. In: **BMC bioinformatics**, v. 14, n. 1, p. 103. 2013.

DE ROURE, D.; GOBLE, C. Software design for empowering scientists. In: **Software**, IEEE, v. 26, n. 1, p. 88-95. 2009.

DETERDING, S. Eudaimonic Design, or: Six Invitations to Rethink Gamification In: **Rethinking Gamification**. Ed. by M. Fuchs, S. Fizek, P. Ruffino, e N. Schrape. Lüneburg, Alemanha: Meson Press. pp. 305–331. 2014.

\_\_\_\_\_. The lens of intrinsic skill atoms: A method for gameful design. In: **Human–Computer Interaction**, v. 30, n. 3-4, p. 294-335. 2015.

DETERDING, S.; DIXON, D.; KHALED, R; NACKE, L. et al. From game design elements to gamefulness: defining gamification. In: **Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments**. ACM. 2011.

DEVLIN, S. et al. Game intelligence. In: **Conference on Computational Intelligence and Games (CIG), 2014**. IEEE. 2014.

DONG, X.; WILD, D. An automatic drug discovery workflow generation tool using semantic web technologies. In: **IEEE Fourth International Conference on eScience, 2008. eScience'08**. 2008

DONGARRA, J. et al. Netlib and NA-Net: Building a scientific computing community. In: **IEEE Annals of the History of Computing**, v. 1, n. 2, p. 30-41. 2008.

DOS SANTOS, J. R. L.; WERNER, H.; RIBEIRO, G.; BELMONTE, S. L. Combination of non-invasive medical imaging technologies and virtual reality systems to generate immersive fetal 3D visualizations. In: **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**. **Anais...** v. 9745, Springer. 2016. p.92–99.

DRUSKAT, S. Lightning Talk: A Proposal for the Measurement and Documentation of Research Software Sustainability in Interactive Metadata Repositories. In: Proceedings of the Fourth Workshop on Sustainable Software for Science: Practice and Experiences (WSSSPE4). **Anais...**, 2016.

DWYER, T. et al. Immersive Analytics (Dagstuhl Seminar 16231). In: **Dagstuhl Reports**. v.6, n.6. 2016.

ELICEIRI, K. W. et al. Biological imaging software tools. In: **Nature methods**, v. 9, n. 7, p. 697-710. 2012.

ELLIS, H. J. et al. A Pipeline Software Architecture for NMR Spectrum Data Translation. In: **Computing in science & engineering**, v. 15, n. 1, p. 76-83. 2013.

ERNST, T. et al. DLR's VirtualLab: scientific software just a mouse click away. In: **Computing in Science and Engineering**, v. 5, n. 1, p. 70-79. 2003.

EVELEIGH, A. et al. I want to be a captain! I want to be a captain!: gamification in the old weather citizen science project. In: **Proceedings of the First International Conference on Gameful Design, Research, and Applications**. 2013.

FDEZ-RIVEROLA, F. et al. A JAVA application framework for scientific software development. In: **Software: Practice and Experience**, v. 42, n. 8, p. 1015-1036. 2012.

FEIBUSH, E.; GAGVANI, N.; WILLIAMS, D. Visualization for situational awareness. In: **IEEE Computer Graphics and Applications**. v. 20, n. 5, p. 38-45. 2000.

FERGUSON, H. T.; GESING, S.; NABRZYSKI, J. Measuring Usability in Decision Tools Supporting Collaborations for Environmental Disaster Response. In: 2016 49th Hawaii International Conference on System Sciences (HICSS). **Anais...** 2016. p.2872–2881.

FERNANDES, J. et al. iThink: A Game-Based Approach Towards Improving Collaboration and Participation in Requirement Elicitation. In: **Procedia Computer Science**, v. 15, p. 66-77. 2012.

FERRARA, J. **Playful Design**. Brooklin, NY. Rosenfeld Media, 2012.

FEYERABEND, P. **Against Method**. New York, NY. Verso. 1993.

\_\_\_\_\_. **Adeus à Razão**. Tradução de Vera Joscelyne. São Paulo, SP. Editora Unesp, 2010.

\_\_\_\_\_. **A Ciência em uma Sociedade Livre**. Tradução de Vera Joscelyne. São Paulo, SP. Editora Unesp, 2011.

FEYNMAN, Richard P.; LEIGHTON, Robert B.; SANDS, Matthews. **The Feynman Lectures in Physics – Mainly Mechanics, Radiation, and Heat**. Reading, Massachusetts. Addison Wesley, 1966.

FINE, Steven S et al. Improving model development with configuration management. In: **Computing in Science & Engineering**, n. 1. p. 56-65. 1998.

FISCHER, F. et al. Integrating segmentation methods from different tools into a visualization program using an object-based plug-in interface. In: **IEEE Transactions on Information Technology in Biomedicine**. v. 14, n. 4, p. 923-934. 2010.

FOSTER, K. R. Software tools [Technology 1998 analysis and forecast]. In: **Spectrum**, IEEE, v. 35, n. 1, p. 52-56, 1998.

FOULSER, D. E.; GROPP, W. D. CLAM and CLAMShell: A system for building user interfaces. In: **Pre-proceedings Second Int'l. Conf. Expert Systems Numer. Comput.** 1990. p. 22-25.

FOURQUET, J.-Y.; HUE, V.; CHIRON, P. OLARGE : on kinematic schemes and regularization for automatic generation of human motion and ergonomic evaluation of workplaces. In: IECON 2007 - 33rd Annual Conference of the IEEE Industrial Electronics Society. **Anais....** 2007. p.2835–2840.

FRANK, A.; STOTZKA, R.; JEJKAL, T.; et al. GridIJ - A Dynamic Grid Service Architecture for Scientific Image Processing. In: 33rd EUROMICRO Conference on Software Engineering and Advanced Applications (EUROMICRO 2007). **Anais...** 2007. p.375–384.

FRANZONI, C.; SAUERMANN, H. Crowd science: The organization of scientific research in open collaborative projects. In: **Research Policy**, v. 43, n. 1, p. 1-20. 2014.

GALISON, P. **How Experiments End**. Chicago. University of Chicago Press, 1987.

GALLOPOULOS, S.; HOUSTIS, E. N.; RICE, J. R. **Future research directions in problem solving environments for computational science**. Purdue University. 1992. Disponível em: <<https://docs.lib.purdue.edu/cstech/954>>. Acesso em: 06 jun. 2018. 2007.

GARCIA ESQUIROL, Ó. Futuro de la enseñanza médica: inteligencia artificial y big data. In: **FEM: Revista de la Fundación Educación Médica**, v. 18, p. s60-s61. 2015.

GARDNER, H.; CSIKSZENTMIHALYI, M.; DAMON, W. **Good Work – When Excellence and Ethics Meet**. New York, NY. Basic Books, 2002.

GEIMER, M.; HOSTE, K.; MCLAY, R. Modern Scientific Software Management Using EasyBuild and Lmod. In: First International Workshop on HPC User Support Tools. **Anais...** 2014. p.41–51.

GERTZ, M. E.; STEWART, D. B.; KHOSLA, P. K. A human machine interface for distributed virtual laboratories. In: **IEEE Robotics & Automation Magazine**. v. 1, n. 4, p. 5-13, 1994.

GESING, S. et al. Using Science Gateways for Bridging the Differences between Research Infrastructures. In: **Journal of Grid Computing**, p. 1-13, 2016.

GILLON, M. et al. Seven temperate terrestrial planets around the nearby ultracool dwarf star TRAPPIST-1. In: **Nature**, 542(7642), p.456-460. 2017.

GLASAUER, S.; JAHRMANN, M. **Science as Game – Art as Play**. Palestra. LASER Zurich. Zurique, 6-12 2016.

GLEZ-PEÑA, D. et al. AIBench: a rapid application development framework for translational research in biomedicine. In: **Computer methods and programs in biomedicine**, v. 98, n. 2, p. 191-203, 2010.

GOBLE, C. Lightning talk: A Simple Profiling Framework for Software User-Producer Reciprocity Review. In: Proceedings of the Fourth Workshop on Sustainable Software for Science: Practice and Experiences (WSSSPE4). **Anais...** 2016.

GOMES, A. T. A. et al. Experiences of the Brazilian national high-performance computing network on the rapid prototyping of science gateways. In: **Concurrency and Computation: Practice and Experience**, v. 27, n. 2, p. 271-289, 2015.

GOOD, B. M.; LOGUERCIO, S.; NANIS, M.; SU, A. I. genegames.org: High-Throughput Access to Biological Knowledge and Reasoning through Online Games. 2012 IEEE Second International Conference on Healthcare Informatics, Imaging and Systems Biology. **Anais...** 2012. p.145–145.

GORTON, I. Cyberinfrastructures: Bridging the Divide between Scientific Research and Software Engineering. **Computer**, v. 47, n. 8, p. 48-55, 2014.

GORTON, I.; SIVARAMAKRISHNAN, C.; BLACK, G.; et al. Velo: A knowledge-management framework for modeling and simulation. **Computing in Science & Engineering**, v. 14, n. 2, p. 12-23, 2012. AIP Publishing.

GREENHILL, A.; HOLMES, K.; LINTOTT, C.; et al. **Playing with science: gamised aspects of gamification found on the Online Citizen Science Project-Zooniverse**. Eurosis. 2014.

GROSS, L. et al. A new design of scientific software using Python and XML. In: **Pure and Applied Geophysics**, v. 165, n. 3-4, p. 653-670. 2008.

HACKING, I. **Representing and Intervening: Introductory Topics in the Philosophy of Natural Science**. Cambridge, Reino Unido. Cambridge University Press. 1983.

HAMARI, J.; KOIVISTO, J.; SARSA, H. Does Gamification Work? -- A Literature Review of Empirical Studies on Gamification. 2014 47th Hawaii International Conference on System Sciences. **Anais...** 2014. p.3025–3034

HANNAY, J. E.; MACLEOD, C.; SINGER, J.; et al. How do scientists develop and use scientific software? 2009 ICSE Workshop on Software Engineering for Computational Science and Engineering. **Anais...** 2009. p.1–8.

HARMS, J. et al. Gamification of Online Surveys: Conceptual Foundations and a Design Process Based on the MDA Framework. Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational. **Anais...**, NordiCHI '14.. p.565-568, 2014. New York, NY, USA: ACM.

HATTON, L.; ROBERTS, A. How accurate is scientific software? In: **IEEE Transactions on Software Engineering**, v. 20, n. 10, pp.785-797, 1994.

HEATON, D.; CARVER, J. C. Claims about the use of software engineering practices in science: a systematic literature review. In: **Information and Software Technology**, v. 67, p. 207-219, 2015.

HELMREICH S. Intimate Sensing. In: **Simulation and its Discontents**. MIT Press, Cambridge, MA, 2009.

HETTRICK, S. et al. **UK Research Software Survey 2014**. Disponível em: <<https://doi.org/10.5281/zenodo.14809>>. Acesso em: 06 jun. 2018. 2014.

HINSEN, K. A glimpse of the future of scientific programming. In: **Computing in Science & Engineering**, v. 15, n. 1, p. 84-88, 2013.

\_\_\_\_\_. **Scientific notations for the digital era**. 2016. Disponível em: <<http://arxiv.org/abs/1605.02960>>. Acesso em: 20 out. 2018.

HOLTHOUSE, M. A.; GREENBERG, S. G. Software technology for scientific and engineering applications. The IEEE Computer Society's Second International Computer Software and Applications Conference. **Anais....** 1978. p.814–818.

HOU, Z.; ZHOU, X.; GU, J.; WANG, Y.; ZHAO, T. ASAAS: Application Software as a Service for High Performance Cloud Computing. 2010 IEEE 12th International Conference on High Performance Computing and Communications (HPCC). **Anais...** 2010. p.156–163.

HOUSTIS, E. N.; RICE, J., **On the Future of Problem-Solving Environments**. Purdue University. 2000. Disponível em: <<https://docs.lib.purdue.edu/cstech/1487/>>. Acesso em: 08 set. 2018.

HOWISON, J.; HERBSLEB, J. D. Scientific software production: incentives and collaboration. In: Proceedings of the ACM 2011 conference on Computer supported cooperative work - CSCW '11. **Anais....** 2011. p.513.

HU, B.; LILL, M. A. PharmDock: a pharmacophore-based docking program. In: **J. Cheminformatics**, v. 6, n. 1, p. 14, 2014.

HUIZINGA, J. **Homo Ludens: o jogo como elemento da cultura**. Tradução de João Paulo Monteiro. São Paulo, Editora Perspectiva, 2004.

HUNICKE, R.; LEBLANC, M.; ZUBEK, R. MDA: A Formal Approach to Game Design and Game Research. Proceedings of the Challenges in Games AI Workshop, Nineteenth National Conference of Artificial Intelligence. **Anais...**, 2004.

HUTSON, H.; JOHNSON, D.; CHUA, X.-Y.; et al. Can Videogame Players Inform Better Scientific Visualization'. In: Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts. **Anais...** 2016. p.181–187.

HWANG, L. J. Lighting Talk - ASPECT: Hackathons as an Example of Sustaining an Open Source Community. In: Proceedings of the Fourth Workshop on Sustainable Software for Science: Practice and Experiences (WSSSPE4). **Anais...**, 2016.

IACOVIDES, I. et al. Do games attract or sustain engagement in citizen science? In: CHI '13 Extended Abstracts on Human Factors in Computing Systems on - CHI EA '13. **Anais...** 2013. p.1101.

ISBISTER, K. Emotion and motion: games as inspiration for shaping the future of interface. In: **Interactions**, v. 18, n. 5, p. 24-27, 2011.

JACOB, C. R. How Open Is Commercial Scientific Software? In: **The Journal of Physical Chemistry Letters**, v. 7, n. 2, p. 351-353, 2016.

JARVIS, R. M. et al. PYCHEM: a multivariate analysis package for python. In: **Bioinformatics**, v. 22, n. 20, p. 2565-2566, 2006.

JAVAHERY, H.; SEFFAH, A.; RADHAKRISHNAN, T. Beyond power: making bioinformatics tools user-centered. In: **Communications of the ACM**, v. 47, n. 11, p. 58-63, 2004.

JENNETT, C.; COX, A. L. Eight guidelines for designing virtual citizen science projects. In: Second AAAI Conference on Human Computation and Crowdsourcing. **Anais...**, 2014.



JONES, M.; SCAFFIDI, C. Obstacles and opportunities with using visual and domain-specific languages in scientific programming. In: **IEEE Symposium in Visual Languages and Human-Centric Computing (VL/HCC)** .2011.

JOPPA, L. N. et al. Troubling trends in scientific software use. In: **Science**, v. 340, n. 6134, p. 814-815, 2013.

JOSHI, A. et al. **Agent Based Systems to Support Multi-disciplinary Problem Solving Environments**. Disponível em: <<http://docs.lib.purdue.edu/cstech/1368>>. Acesso em: 08 set. 2018.1997.

JULVEZ, J.; MATCOVSCHI, M. H.; PASTRAVANU, O. MATLAB tools for the analysis of Petri net models. In: **Emerging Technology and Factory Automation (ETFA), 2014**. IEEE. 2014

JUUL, J. **Half-real: video games between real rules and fictional worlds**. Cambridge, Mass.: MIT Press, 2005.

\_\_\_\_\_. **The art of failure an essay on the pain of playing video games**. Cambridge, MA: MIT Press.2013.

KAPPEN, D. L.; JOHANNSMIEIER, J.; NACKE, L. E. Deconstructing “gamified” task-management applications. Proceedings of the First International Conference on Gameful Design, Research, and Applications - Gamification '13. **Anais....** 2013. p.139–142.

KARAMANIS, N.; PIGNATELLI, M.; CARVALHO-SILVA, D.; et al. Designing an intuitive web application for drug discovery scientists. In: **Drug Discovery Today**, v. 23, n. 6, p. 1169–1174, 2018.

KATZ, D. S. **Building Scientific Software Communities**. Palestra. HEP Software Foundation Workshop 2015. Menlo Park, CA. Disponível em: <<http://pt.slideshare.net/danielskatz/software-communities>>. Acesso em: 20 out. 2018.

KEEFE, D. F. Integrating visualization and interaction research to improve scientific workflows. In: **Computer Graphics and Applications, IEEE**, v. 30, n. 2, p. 8-13, 2010.

KEENAGHAN, G.; HORVATH, I. State of the Art of Using Virtual Reality Technologies in Built Environment Education. In: I. Horváth; Z. Rusák (Orgs.); Proceedings of TMCE 2014. **Anais...**, 2014.

KELLER, E. F. **Reflections on Gender and Science**. New Haven, Conn. Yale University Press, 1985.

\_\_\_\_\_. **Making Sense of Life: Explaining Biological Development with Models, Metaphors, and Machines**. Cambridge, Mass. Harvard University Press. 2003.

KELLER, R. M.; RIMON, M. A knowledge-based software development environment for scientific model-building. In: Proceedings of the Seventh Knowledge-Based Software Engineering Conference. **Anais...** .1992. p.192–201.

KELLER, R. M.; RIMON, M.; DAS, A. A knowledge-based prototyping environment for construction of scientific modeling software. **Automated Software Engineering**, v. 1, n. 1, p. 79-128, 1994. Springer.

KELLY, D. A Software Chasm: Software Engineering and Scientific Computing. In: **IEEE Software**, v. 24, n. 6, p. 120-119, 2007.

\_\_\_\_\_. Determining factors that affect long-term evolution in scientific application software. In: **Journal of Systems and Software**, v. 82, n. 5, p. 851-861, 2009.

\_\_\_\_\_. An analysis of process characteristics for developing scientific software. In: **Innovative Strategies and Approaches for End-User Computing Advancements**. IGI Global, 2013.

\_\_\_\_\_. Scientific software development viewed as knowledge acquisition: Towards understanding the development of risk-averse scientific software. In: **Journal of Systems and Software**, v. 109, p. 50-61, 2015.

KELLY, D.; HOOK, D.; SANDERS, R. Five Recommended Practices for Computational Scientists Who Write Software. In: **Computing in Science & Engineering**, v. 11, n. 5, p. 48–53, 2009.

KELLY, D.; SANDERS, R. Assessing the quality of scientific software. In: First International Workshop on Software Engineering for Computational Science and Engineering. **Anais...**, 2008a.

\_\_\_\_\_. The Challenge of Testing Scientific Software. In: CAST 2008: Beyond the Boundaries. **Anais...**, 2008b.

KELLY, D. F; SKORDAKI, E. M. A medical computing lab and a model of learning. In: 2015 IEEE 28th Canadian Conference on Electrical and Computer Engineering (CCECE). **Anais...** 2015. p.1561-1566.

KELLY, D.; SMITH, S. 3rd CASCON Workshop on Software Engineering for Science. In: **Proceedings of the 2010 Conference of the Center for Advanced Studies on Collaborative Research**. IBM Corp., 2010.

KELLY, D.; THORSTEINSON, S.; HOOK, D. Scientific software testing: analysis with four dimensions. In: **Software, IEEE**, v. 28, n. 3, p. 84-90, 2011.

KENDALL, R. P. et al. **A proposed taxonomy for software development risks for high-performance computing (HPC) scientific/engineering applications**. Disponível em: <<https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=8013>> Acesso em: 12 ago. 2017. 2007.

KILLCOYNE, S.; BOYLE, J. Managing chaos: lessons learned developing software in the life sciences. In: **Computing in science & engineering**, v. 11, n. 6, p. 20-29, 2009.

KIM, T. W. Gamification Ethics: Exploitation and Manipulation. In: **CHI 2015 proceedings**. Seul, Coréia do Sul. 2015

KNORR CETINA, K. **The Manufacture of Knowledge: An Essay on the Constructivist and Contextual Nature of Science**. Oxford, Reino Unido, Pergamon Press. 1981.

\_\_\_\_\_. **Epistemic cultures: How the sciences make knowledge**. Cambridge, Massachusetts. Harvard University Press, 2009.

KODOSKY, J.; MACCRISKEN, J.; RYMAR, G. Visual programming using structured data flow. In: Proceedings 1991 IEEE Workshop on Visual Languages. **Anais....** 1991. p.34–39.

KOLBERG, S.; COURIVAUD, D.; OZBEK, M. E. LMS and Interactivity - Technical Issues for Remote Laboratories. 2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications. **Anais....** 2007. p.1–4.

KORNBLUH, K. Engineering software-seeing data in action. In: **Spectrum**, IEEE, v. 30, n. 11, p. 60-64, 1993.

KOSMADOUDI, Z. et al. Engineering design using game-enhanced CAD: The potential to augment the user experience with game elements. In: **Computer-Aided Design**, v. 45, n. 3, p. 777-795, 2013.

KOTESKA, B.; MISHEV, A. Software engineering practices and principles to increase quality of scientific applications. In: **ICT Innovations 2012**. Springer, 2013. p. 245-254.

KOVALCHUK, S. V. et al. Virtual Simulation Objects concept as a framework for system-level simulation. 2012 IEEE 8th International Conference on E-Science. **Anais....** 2012. p.1–8.

KRAKER, P; LEX, E. A critical look at the ResearchGate score as a measure of scientific reputation. In: **Proceedings of the quantifying and analysing scholarly communication on the web workshop (ASCW'15), Web Science conference**. 2015

KRYUKOV, A.; DEMICHEV, A.; POLYAKOV, S. Web platforms for scientific research. In: **Programming and Computer Software**, v. 42, n. 3, p. 129-141, 2016.

KUHN, T.S. **The Essential Tension. Selected Studies in Scientific Tradition and Change**. Chicago. University of Chicago Press, 1977.

\_\_\_\_\_. **The Road Since Structure**. Chicago. University of Chicago press, 2000.

\_\_\_\_\_. **The structure of scientific revolutions**. Chicago. University of Chicago press, 2012.

LA RUE, M.; SHEN, C.; DOIZER, A.; BECK, M. Enhanced GUI environment for Pymatgen in material science. 2014 IEEE 5th International Conference on Software Engineering and Service Science. Anais... 2014. p.1186–1190.

LANDAU, E. **Earth Sized Worlds**. NASA. Disponível em: <nasaviz.gsfc.nasa.gov/12541>. Acesso em: 30 jul. 2018.

LANDE, D. R. **Implementation of an XML-based user interface with applications in ice sheet modeling**, 2008. University of Montana. Disponível em: <https://scholarworks.umt.edu/etd/554/>. Acesso em: 30 jul. 2018.

LARKIN, N.; RAFFUSE, S.; PRYDEN, D.; et al. **Conversion of the BlueSky Framework into collaborative web service architecture and creation of a smoke modeling application**. U.S. Joint Fire Science Program. 2009. Disponível em: <http://digitalcommons.unl.edu/jfspresearch/143>. Acesso em: 30 jul. 2018.

LATOUR, B. **Science in Action: How to Follow Scientists and Engineers Through Society**. Cambridge, Mass. Harvard University Press, 1987.

\_\_\_\_\_. **Cogitamus: Seis cartas sobre as humanidades científicas**. Tradução de Jamille Pinheiro Dias. São Paulo, SP. Editora 34, 2010.

LATOUR, B.; WOOLGAR, S. Laboratory life: **The construction of scientific facts**. Princeton, NJ. Princeton University Press, 1986.

LETONDAL, C. Participatory programming: Developing programmable bioinformatics tools for end-users. In: **End user development**. Springer, 2006. p. 207-242.

LETONDAL, C.; MACKAY, W. E. Participatory programming and the scope of mutual responsibility. Proceedings of the eighth conference on Participatory design Artful integration: interweaving media, materials and practices - PDC 04. **Anais....** v. 1. 2004. p.31.

LETONDAL, C.; ZDUN, U. Anticipating Scientific Software Evolution as a Combined Technological and Design Approach. Proceedings of Second International Workshop on Unanticipated Software Evolution. **Anais....** 2003. p.1–15.

LI, W.; GROSSMAN, T.; FITZMAURICE, G. Gamicad: a gamified tutorial system for first time autocad users. In: **Proceedings of the 25th annual ACM symposium on User interface software and technology**. ACM, 2012

LI, Y. Reengineering a scientific software and lessons learned. Proceeding of the 4th international workshop on Software engineering for computational science and engineering - SECSE '11. **Anais....** 2011. p.41.

LIST, M.; EBERT, P.; ALBRECHT, F. Ten Simple Rules for Developing Usable Software in Computational Biology. In: PLOS Computational Biology, v. 13, n. 1, p. e1005265, 2017.

LO, T. T. et al. Using Gamification for Collaborative Mass-Housing Design Process. In: **Across: Architectural Research through to Practice: 48th International Conference of the Architectural Science Association**. 2014.

LOYNTON, S.; SLOAN, D.; BUREL, J.-M.; MACAULAY, C. Towards a project community approach to academic scientific software development. 2009 5th IEEE International Conference on E-Science Workshops. **Anais...** 2009. p.120–124.

LUNA, D. R., RIZZATO LEDE, D. A., OTERO, C. M., RISK, M. R., GONZLEZ BERNALDO DE QUIRÓS, F. User-centered Design Improves the Usability of Drug-drug Interaction Alerts. In: **J. of Biomedical Informatics** 66.C, pp 204-213. 2017.

LUNDSTROM, M.; KLIMECK, G. The NCN: Science, Simulation, and Cyber Services. 2006 IEEE Conference on Emerging Technologies - Nanoelectronics. **Anais....** 2006. p.496–500.

LYRIO, G. H. S. D. O.; SEIXAS, R. B. Didactic Games System: Fifteen years of development in military simulation. Proceedings of The North American Simulation and AI in Games Conference - GAMEON-NA. **Anais...**, 2011.

MACAULAY, C. et al. Usability and User-Centered Design in Scientific Software Development. In: **IEEE Software**, v. 26, n. 1, p. 96-102, 2009.

MACDONALD, E. et al. Aurorasaurus: A citizen science platform for viewing and reporting the aurora. In: **Space Weather**, v. 13, n. 9, p. 548-559, 2015.

MACLEOD, R. S.; JOHNSON, C. R.; MATHESON, M. A. Visualization of cardiac bioelectricity-a case study. In: Proceedings Visualization '92. **Anais....** 1992. p.411–418.

MAEDA, T.; AOKI, Y.; MURATA, T. A problem-solving environment that assists model development for reinforcement learning algorithms. In: 5th International Conference on Computer Sciences and Convergence Information Technology. **Anais....** 2010. p.260–265.

MALINA, R. F. Intimate science and hard humanities. In: **Leonardo**, v. 42, n. 3, p. 184-184, 2009.

MANJUNATHA, A.; ANDERSON, P.; RANABAHU, A.; SHETH, A. Identifying and Implementing the Underlying Operators for Nuclear Magnetic Resonance Based Metabolomics Data Analysis. In: 3rd International Conference on Bioinformatics and Computational Biology (BICoB-2011). **Anais....**, 2011.

MAO, K.; CAPRA, L.; HARMAN, M.; JIA, Y. A survey of the use of crowdsourcing in software engineering. In: **Journal of Systems and Software**, v. 126, p. 57–84, 2017.

MARASCO, E. A.; BEHJAT, L.; ROSEHART, W. D. Integration of Gamification and Creativity in Engineering Design. In: Proceedings of 2015 ASEE Annual Conference & Exposition. **Anais...** 2015. p.26.997.1-26.997.9,

MARINOVICI, C.; KIRKHAM, H.; GLASS, K. The Hidden Job Requirements for a Software Engineer. In: 2014 47th Hawaii International Conference on System Sciences. **Anais....** 2014. p.4979–4984.

MASON, A. D.; MICHALAKIDIS, G.; KRAUSE, P. J. Tiger Nation: Empowering citizen scientists. In: 2012 6th IEEE International Conference on Digital Ecosystems and Technologies (DEST). **Anais....** 2012. p.1–5,

MCCAIN, G.; SEGAL, E. M. **The Game of Science**. Belmont, California: Brooks/Cole Pub Co. 1989.

MCFADDIN, H. S.; RICE, J. R. **Architecture of the RELAX problem solving environment**. Purdue University - Department of Computer Science. 1992. Disponível em: <<https://docs.lib.purdue.edu/cstech/1001>> Acesso em: 08 set. 2018.

MCGONIGAL, J. **Reality is broken: Why games make us better and how they can change the world.** Penguin, 2011.

MCKIERNAN, E. C. et al. How open science helps researchers succeed. In: **eLife**, v. 5, p. e16800, 2016.

MCKINEY, K. M. Thinking Inside the Box. In: **Scientific Computing and Instrumentation** 20 (SUPP. LAB INF.): 18-21. 2003.

MENDEZ, D.; VILLAMIAZR, M.; CASTRO, H. e-Clouds: Scientific Computing as a Service. In: 2013 Seventh International Conference on Complex, Intelligent, and Software Intensive Systems. **Anais...** 2013. p.481–486.

MIELKE, A. M. et al. Independent sensor networks. In: **Instrumentation & Measurement Magazine, IEEE**, v. 8, n. 2, p. 33-37, 2005.

MILES, S. **Creating an immersive Mars experience using Unity3D.** Jet Propulsion Laboratory, National Aeronautics and Space Administration. 2011. Disponível em: <<https://trs.jpl.nasa.gov/handle/2014/43730>> Acesso em: 08 set. 2018.

MILLS, J. K. et al. Development of a robot control test platform. In: **Robotics & Automation Magazine, IEEE**, v. 2, n. 4, p. 21-28, 1995.

MOHAMMAD, A. F. A new perspective in Scientific Software Development. In: **Innovations and Advances in Computer Sciences and Engineering.** Springer, 2010. p. 129-134.

MOLONEY, J. Videogame Technology Re-Purposed: Towards Interdisciplinary Design Environments for Engineering and Architecture. In: **Procedia Technology**, v. 20, p. 212-218, 2015.

MORA, A.; RIERA, D.; GONZALEZ, C.; ARNEDO-MORENO, J. A Literature Review of Gamification Design Frameworks. In: 2015 7th International Conference on Games and Virtual Worlds for Serious Applications (VS-Games). **Anais...** 2015. p.1–8,

MORAIS, H. et al. SOSPO-SP: Secure operation of sustainable power systems simulation platform for real-time system state evaluation and



control. In: **IEEE Transactions on Industrial Informatics**, v. 10, n. 4, p. 2318-2329, 2014.

MORRIS, C.; SEGAL, J. Lessons Learned from a Scientific Software Development Project. In: **Software, IEEE**, v. 29, n. 4, p. 9-12, 2012.

MORRIS, B. J. et al. Gaming science: the “Gamification” of scientific thinking. In: **Frontiers in Psychology**, v. 4, 2013.

MORSCHHEUSER, B.; HASSAN, L.; WERDER, K.; HAMARI, J. How to design gamification? A method for engineering gamified software. In: **Information and Software Technology**, v. 95, p. 219-237, Elsevier. 2018.

MURALI, P.; DUTTA, D.; BISWAS, R. N. An integrated framework for quality scientific software development. In: **ACM SIGPLAN Fortran Forum**, v. 12, n. 3, p. 19–25, 1993.

MURPHY, M. J. **Utility of coupling nonlinear optimization methods with numerical modeling software**. Lawrence Livermore National Laboratory. 1996. Disponível em: <<https://digital.library.unt.edu/ark:/67531/metadc688067/>> Acesso em: 08 set. 2018.

NACKE, L. E.; DETERDING, S. The maturing of gamification research. In: **Computers in Human Behavior**, 71, pp. 450–454. 2017.

NANTHAAMORNPHONG, A.; CARVER, J. C. Test-Driven Development in scientific software: a survey. In: **Software Quality Journal**, p. 1-30. 2015.

NERSESSIAN, N. J. **Creating Scientific Concepts**. Cambridge, Mass. The MIT Press, 2008.

NGUYEN, T.; TRIFAN, L.; DESIDERI, J.-A. Resilient workflows for cooperative design. In: Proceedings of the 2011 15th International Conference on Computer Supported Cooperative Work in Design (CSCWD). **Anais...** 2011. p.69–75.

NGUYEN-HOAN, L.; FLINT, S.; SANKARANARAYANA, R. A survey of scientific software development. Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement - ESEM '10. **Anais...** 2010. p.1.

NOVOSELOV, K. S. et al. A roadmap for graphene. In: **Nature**, v. 490, n. 7419, p. 192-200, 2012.

NTOMBELA, M.; KABERERE, K. K.; FOLLY, K. A.; PETROIANU, A. I. An investigation into the capabilities of MATLAB power system toolbox for small signal stability analysis in power systems. Proceedings of the Inaugural IEEE PES 2005 Conference and Exposition in Africa. **Anais....** 2005. p.242–248.

OPRESCU, F.; JONES, C.; KATSIKITIS, M. I play at work - ten principles for transforming work processes through gamification. In: **Frontiers in psychology**, 5, 2014.

OVASKA, P.; ROSSI, M.; SMOLANDER, K. Filtering, negotiating and shifting in the understanding of information system requirements. In: **Scandinavian Journal of Information Systems**, v. 17, n. 1, p. 7, 2005.

PANCAKE, Cherri M. 'Improving the Usability of Numerical Software through User-Centered Design. In: **Quality of Numerical Software: Assessment and Enhancement**, ed. B. Ford and J. Rice, Chapman & Hall, 1996.

PAPADIMITRIOU, S. et al. Scientific scripting for the Java platform with jLab. In: **Computing in Science & Engineering**, v. 11, n. 4, p. 50-60, 2009.

\_\_\_\_\_. Exploiting Java scientific libraries with the Scala language within the ScalaLab environment. In: **Software, IET**, v. 5, n. 6, p. 543-551, 2011.

PAWLIK, A.; SEGAL, J.; PETRE, M. Documentation practices in scientific software development. In: CHASE '12 Proceedings of the 5th International Workshop on Co-operative and Human Aspects of Software Engineering. **Anais...** 2012. p.113-119.

PAWLIK, A. et al. Crowdsourcing scientific software documentation: a case study of the NumPy documentation project. In: **Computing in Science & Engineering**, v. 17, n. 1, p. 28-36, 2015.

PEREIRA JUNIOR, M. **Concepção de um Processo de Desenvolvimento Específico para Software Científico**. 2007.

Dissertação (Mestrado em Computação). Centro Federal de Educação Tecnológica de Minas Gerais. 2007.

PHILLIPS, R. L. et al. A bridge from full-function to reduced-function workstations. In: **Computer Graphics and Applications**, IEEE, v. 6, n. 5, p. 53-57, 1986.

PICON, A. et al. 3D High Precision Tube Bevel Measurement using laser based Rotating Scanner. In: 2006 IEEE Conference on Emerging Technologies and Factory Automation. **Anais....** 2006. p.1190–1197.

PINTO, G. d. R. B. et al. A framework to support scientific knowledge management: a case study in agro-meteorology. In: The 7th International Conference on Computer Supported Cooperative Work in Design. **Anais...** 2002. p.320–324.

PLATZ, J. Project management in the development of scientific software. In: **Computer Physics Communications**, v. 41, n. 2, p. 217-225, 1986.

POINCARÉ, H. Science and Method. In: **The foundations of science: Science and hypothesis, the value of science, science and method**. Tradução de George Bruce Halstead. New York, NY. The Science Press. 1921.

\_\_\_\_\_. Science and Hypothesis. In: **The foundations of science: Science and hypothesis, the value of science, science and method**. Tradução de George Bruce Halstead. New York, NY. The Science Press. 1921b.

\_\_\_\_\_. **O Valor da Ciência**. Tradução de Maria Helena Franco Martins. Rio de Janeiro, RJ. Contraponto Editora. 1995.

PONCE, M. et al. **Designing Advanced Research Computing Academic Programs**. SciNet HPC Consortium, University of Toronto. Toronto, Ontario, 2016. Disponível em: <  
[https://www.researchgate.net/publication/301878014\\_Designing\\_Advanced\\_Research\\_Computing\\_Academic\\_Programs](https://www.researchgate.net/publication/301878014_Designing_Advanced_Research_Computing_Academic_Programs) > Acesso em: 12 ago. 2018.

PONTI, M.; HILLMAN, T.; STANKOVIC, I. Science and Gamification. In: Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play - CHI PLAY '15. **Anais....** 2015. p.679–684.

POPA, D. M. **Design Case: Gamification of ERP – A user centered design approach**. 2013. Disponível em: <<http://gamification-research.org/wp-content/uploads/2013/03/Popa.pdf>> Acesso em: 12 ago 2018.

POPPER, K. **Conjectures and Refutations**. London. Routledge, 2002.

\_\_\_\_\_. **The Logic of Scientific Discovery**. London. Routledge, 2005.

\_\_\_\_\_. **All Life is Problem Solving**. London. Routledge, 2010.

PREGO, J. J. G.; SEISDEDOS, L. V. Tailor-made small simulator for a drum boiler control based on linear techniques. In: ETFA2011. **Anais...** .2011. p.1–4.

PRESTOPNIK, N. R.; TANG, J. Points, stories, worlds, and diegesis: Comparing player experiences in two citizen science games. In: **Computers in Human Behavior**, v. 52, p. 492-506, 2015.

QUEIROZ, F.; LONSDALE, M. D. s.; SPITZ, R. [Artigo em revisão]. Submetido.

QUEIROZ, F.; SPITZ, R. The Lens of the Lab: Design Challenges in Scientific Software. In: **International Journal of Design Management and Professional Practice**. v.10, n.3. pp.17-45. Common Ground Publishing. 2016.

\_\_\_\_\_. Position Paper: Collaborative Gamification Design for Scientific Software. In: Proceedings of the Fourth Workshop on Sustainable Software for Science: Practice and Experiences (WSSSPE4). **Anais...** 2016b.

QUEIROZ, F. et al. Video Games as Inspiration for Scientific Software. Proceedings of SBGames 2016. **Anais....** 2016. p.387–396,

\_\_\_\_\_. Good Usability Practices in Scientific Software. In: Proceedings of the Workshop on Sustainable Software for Science: Practice and Experiences (WSSSPE5.1). **Anais....** 2017.

RAMAKRISHNAN, N.; RICE, J. R. **Knowledge Discovery in Computational Science: A Case Study in Algorithm Selection**. Department of Computer Science Technical Reports. Purdue University. 1996. Disponível em: <<https://docs.lib.purdue.edu/cstech/1335>> Acesso em: 12 ago. 2018.

RAMAKRISHNAN, L.; GUNTER, D. Ten Principles for Creating Usable Software for Science. In: 2017 IEEE 13th International Conference on e-Science (e-Science). **Anais....** 2017. p.210–218.

RAMPERSAD, L. et al. Improving the usability of scientific software with participatory design. In: Proceedings of the South African Institute of Computer Scientists and Information Technologists on - SAICSIT '17. **Anais....** 2017; p.1–9.

RAPOSO, A. B. et al. A system for integrated visualization in oil exploration and production. In: **IEEE Computer Graphics and Applications**, 36(3):10–16. 2016.

RECIO-GARCIA, J. A.; DIAZ-AGUDO, B.; GONZÁLEZ-CALERO, P. A. The COLIBRI Open Platform for the Reproducibility of CBR Applications. In: **Case-Based Reasoning Research and Development**. Springer. p. 255–269. 2013.

RICE, J. R. **Scalable Scientific Software Libraries and Problem Solving Environments**. Department of Computer Science Technical Reports. Purdue University. 1996. Disponível em: <<https://docs.lib.purdue.edu/cstech/1257>> Acesso em: 12 ago 2018.

RICE, J. R.; BOISVERT, R. F. **From scientific software libraries to problem-solving environments**. In: Computing in Science & Engineering, n. 3, p. 44-53, 1996.

RIJNDERS, F. M.; SPOELDER, H. J. W.; GROEN, F. C. A. Distributed visual programming environment: Applications within data-acquisition. In: 1993 IEEE Instrumentation and Measurement Technology Conference. **Anais...** 1993. p.690–693.

RIOS, F. **Preserving and Sharing Software for Transparent and Reproducible Research: A Review**, v1.2. 2016. Disponível em: <[osf.io/d4kef](https://osf.io/d4kef)>. Acesso em: 23 abr. 2017.

ROCHA, A. C. et al. Statistical processing of radiometer data using MATLAB. In: 11th International Conference on Antennas and Propagation (ICAP 2001). **Anais...** 2001. p.177–180.

ROUSE III, R. Interview: Will Wright. In: Rouse III, R. **Game Design: Theory and Practice**. Plano: Wordware Publishing, 2001.

RUIVENKAMP, M.; RIP, A. Nanoimages as Hybrid Monsters. In: **Representation in Scientific Practice Revisited**. Cambridge, Mass.; The MIT Press, 2014.

SALEN, K. S.; ZIMMERMAN, E. **Rules of play: game design fundamentals**. Cambridge, Mass: MIT Press, 2004.

SANDERS, E. B. N.; STAPPERS, P. J. Co-creation and the new landscapes of design. In: **Co-design**, v.4 n.1, p.5-18. 2008.

SANDERS, R. **The development and use of scientific software**. Dissertação (Mestrado em Computação) - Queen's University of Kingston, Ontario, Canadá, 2008.

SANDERS, R.; KELLY, D. Dealing with Risk in Scientific Software Development. In: **IEEE Software**, v. 25, n. 4, p. 21-28. 2008.

SAVELLI, R.; S O LYRIO, G.; DE, R.; SEIXAS, R. Operations Other Than War: Modeling and Simulation. In: Proceedings of The North American Simulation and AI in Games Conference - GAMEON-NA. **Anais...**, 2010.

SCHNEIDEWIND, L.; HOROLD, S.; MAYAS, C.; et al. How personas support requirements engineering. In: 2012 First International Workshop on Usability and Accessibility Focused Requirements Engineering (UsARE). **Anais...** 2012. p.1–5.

SCHRAEFEL, M. C. et al. Breaking the book. In: Proceedings of the 2004 conference on Human factors in computing systems - CHI '04. **Anais...** 2004. p.25–32.

SCHROPE, M. Solving tough problems with games. In: **Proceedings of the National Academy of Sciences**, v. 110, n. 18, p. 7104-7106, 2013.

SEGAL, J. Some Problems of Professional End User Developers. In: **IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC 2007)**. 2007.

\_\_\_\_\_. Some challenges facing software engineers developing software for scientists. In: 2009 ICSE Workshop on Software Engineering for Computational Science and Engineering. **Anais....** 2009. p.9–14.

SEGAL, J.; MORRIS, C. Developing Scientific Software. In: **IEEE Software**, v. 25, n. 4, p. 18-20, 2008.

\_\_\_\_\_. Developing software for a scientific community: some challenges and solutions. In: **Handbook of Research on Computational Science and Engineering: Theory and Practice**, p. 177-196, 2012.

SHENOY, S.; JAYARAM, B.; LATHA, N.; et al. From Gene to Drug: A Proof of Concept for a Plausible Computational Pathway. In: Sixth International Conference on Intelligent Systems Design and Applications. **Anais...** 2006. p.1147–1152.

SILVA, L. A. M. DA. **Composer-science: um framework para a composição de workflows científicos**, 2010. Dissertação (Mestrado em Modelagem Computacional). Universidade Federal de Juiz de Fora, Juiz de Fora, Minas Gerais.

SIMMONS, B. et al. Defining and Measuring Success in Online Citizen Science: A Case Study. In: **Computing in Science & Engineering**, v. 17, p. 28, 2015.

SKIDMORE, E. et al. iPlant atmosphere. Proceedings of the 2011 ACM workshop on Gateway computing environments - GCE '11. **Anais...** 2011. p.59.

SKORDAKI, E.-M., BAINBRIDGE, S. Blended Training on Scientific Software: A Study on How Scientific Data are Generated. In: **International Review of Research in Open and Distributed Learning**. v.19, n.2. 2018.

SLETHOLT, M. T. et al. What do we know about scientific software development's agile practices? In: **Computing in Science & Engineering**, v. 14, n. 2, p. 24-37, 2012.

SLOAN, D. et al. User research in a scientific software development project. In: Proceedings of the 23rd British HCI Group Annual Conference on People and Computers: Celebrating People and Technology. **Anais...** 2009. p.423–429.

SMITH, S.; JEGATHEESAN, T.; KELLY, D. Advantages, disadvantages and misunderstandings about document driven design for scientific software. In: Proceedings of the Fourth International Workshop on Software Engineering for HPC in Computational Science and Engineering. **Anais...** 2016. P.41-48.

SPENCER, M. Brittleness and Bureaucracy: Software as a Material for Science. In: **Perspectives on Science**, v. 23, n. 4, p. 466–484, 2015.

SPRINGMEYER, R. R. Applying observations of work activity in designing prototype data analysis tools. In: Proceedings Visualization '93. **Anais...** 1993. p.228–235.

STEWART, M. E. M. Automated analysis of scientific and engineering semantics. In: Proceedings 9th International Workshop on Program Comprehension. IWPC 2001. **Anais....** 2001. p.113–114.

SWINK, S. **Game feel: a game designer's guide to virtual sensation**. Burlington, MA. Morgan Kaufmann Publishers, 2009.

TAKATSUKA, M.; GAHEGAN, M. N. Exploratory geospatial analysis using GeoVISTA Studio: from a desktop to the Web. In: Proceedings of the Second International Conference on Web Information Systems Engineering. **Anais....** 2001. p.92–101.

TALBOTT, T. et al. Adapting the electronic laboratory notebook for the semantic era. In: Proceedings of the 2005 International Symposium on Collaborative Technologies and Systems, 2005. **Anais....** 2005. p.136–143.

TAUNAY, H. et al. A spatial partitioning heuristic for automatic adjustment of the 3D navigation speed in multiscale virtual environments. In: 2015 IEEE Symposium on 3D User Interfaces (3DUI). **Anais....** 2015. p.51–58.

TAWHEEL, A.; DELANEY, B.; ZHAO, L. Knowledge Management in Distributed Scientific Software Development. In: 2009 Fourth IEEE



International Conference on Global Software Engineering. **Anais...** .2009 p.299–300,

TERRANOVA, N.; MAGNI, P. TGI-Simulator: a visual tool to support the preclinical phase of the drug discovery process by assessing in silico the effect of an anticancer drug. In: **Computer methods and programs in biomedicine**, v. 105, n. 2, p. 162-174, 2012.

THOMER, A. K. et al. Co-designing Scientific Software: Hackathons for Participatory Interface Design. In: Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '16. **Anais....** 2016. p.3219–3226.

TRAINER, E. IHackathons, Codefests, and Sprints: A Case for Attention in CSCW on Community Coding Eventse. In: Human Computer Interaction Consortium Workshop (HCIC 2016). **Anais...**, 2016.

TRIPP, David. Pesquisa-ação: uma introdução metodológica. In: **Educação e Pesquisa**, São Paulo, v. 31, n. 3. 2005. Disponível em <[http://www.scielo.br/scielo.php?script=sci\\_arttext&pid=S1517-97022005000300009&lng=en&nrm=iso](http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1517-97022005000300009&lng=en&nrm=iso)>. Acesso em: 13 abr. 2015.

TRLICA, C. Software tools [technology analysis and forecast]. In: **Spectrum**, IEEE, v. 34, n. 1, p. 60-64, 1997.

TURK, M. Fostering Collaborative Computational Science. In: **Computing in Science & Engineering**, v. 16, n. 2, p. 68-71, 2014.

TURKLE, S. **The Second Self: Computers and the Human Spirit – Twentieth Anniversary Edition**. MIT Press, Cambridge, MA, 2005.

\_\_\_\_\_. Simulation and its Discontents. In: **Simulation and its Discontents**, MIT Press, Cambridge, MA, 2009.

VANSCHOREN, J. et al. OpenML: networked science in machine learning. In: **ACM SIGKDD Explorations Newsletter**, v. 15, n. 2, p. 49-60, 2014.

VENTERS, C. et al. The blind men and the elephant: Towards an empirical evaluation framework for software sustainability. In: **Journal of Open Research Software**, v. 2, n. 1, 2014.

VERIGAN, A. **Improving Pediatric Cardiology Consultation Methods by Introducing Digital Interactive 3-D Heart Models: A Proof of Concept Study**. Tese (Doutorado em Engenharia Química) – University of South Florida. 2007

VERSEK, C. W. **Charge transport studies of proton and ion conducting materials**. Tese (Doutorado em Física). University of Massachusetts Amherst. 2013.

VERTESI, J. Drawing as: Distinctions and Disambiguation in Digital Images of Mars. In: **Representation in Scientific Practice Revisited**. Cambridge, Mass.; The MIT Press, 2014.

VIGDER, M. et al. Supporting the everyday work of scientists: Automating scientific workflows. In: **IEEE Software**. v. 25, n. 4. 2008.

WANG, G.; TRAN, T. N.; ANDRADE, H. A. A graphical programming and design environment for FPGA-based hardware. In: 2010 International Conference on Field-Programmable Technology. **Anais...** .2010. p.337–340.

WANG, J.; BENNETT, K. J.; GUINNESS, E. A. Virtual Astronaut for Scientific Visualization—A Prototype for Santa Maria Crater on Mars. In: **Future Internet**, v. 4, n. 4, p. 1049-1068, 2012.

WATSON, J. D. **The Double Helix: A Personal Account of the Discovery of the Structure of DNA (A Norton Critical Edition)**. Norton and Company, 1980.

WAUER, J. et al. Two software tools for plane-wave scattering on nonspherical particles in the German Aerospace Center's Virtual Laboratory. In: **Applied optics**, v. 43, n. 35, p. 6371-9, 2004.

WEERAWARANA, S. et al. **The Purdue PSE Kernel: Towards a Kernel for Building PSEs**. Department of Computer Science Technical Reports. Purdue University. 1996. Disponível em: <<https://docs.lib.purdue.edu/cstech/1336>>. Acesso em: 06 jun. 2018.

WERNER, H et al. The use of rapid prototyping didactic models in the study of fetal malformations. In: **Ultrasound in Obstetrics & Gynecology**, v. 32, n. 7, p. 955-956, 2008.

WILLSON, B.; WHITHAM, J.; ANDERSON, C. Estimating ignition timing from engine cylinder pressure with neural networks. In: Proceedings of the Intelligent Vehicles '92 Symposium. **Anais....** 1992. p.108–113.

WILSON, G. et al. Best Practices for Scientific Computing. In: **PLoS Biology**, v. 12, n. 1, p. e1001745, 2014.

\_\_\_\_\_. Good enough practices in scientific computing. In: **PLOS Computational Biology**, v. 13, n. 6, p. e1005510, 2017. Public Library of Science. Disponível em: <<http://dx.plos.org/10.1371/journal.pcbi.1005510>>. Acesso em: 23 out. 2018.

WOLFF, C. The case for teaching “tool science”; taking software engineering and software engineering education beyond the confinements of traditional software development contexts. In: 2015 IEEE Global Engineering Education Conference (EDUCON). **Anais....** 2015. p.932–938.

WOLSTENCROFT, K. et al. SEEK: a systems biology data and model management platform. In: **BMC systems biology**, v. 9, n. 1, p. 33, 2015.

WOOLLARD, D. et al. Scientific Software as Workflows: From Discovery to Distribution. In: **IEEE Software**, v. 25, n. 4, p. 37-43, 2008.

YAMAZAKI, T. et al. Simulation Platform: A cloud-based online simulation environment. In: **Neural Networks**, v. 24, n. 9, p. 927-932, 2011.

## **7**

### **Anexos**

#### **7.1**

##### **Termos de Consentimento**

As próximas páginas reproduzem os termos de consentimento recolhidos durante a pesquisa-ação relatada no Capítulo 4. À época, a tese intitulava-se “Jogando o Software Científico”.

PONTIFÍCIA UNIVERSIDADE CATÓLICA  
DO RIO DE JANEIRO



## TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Eu, **Francisco Oliveira de Queiroz**, informo que estou realizando a Tese de Doutorado "Jogando o Software Científico" no Programa de Pós-graduação em Design do Departamento de Artes & Design da Pontifícia Universidade Católica do Rio de Janeiro. A pesquisa referente à tese conta com a participação de grupos de pesquisas integrantes ou ligados ao Tecgraf/PUC-Rio, do qual você é um dos integrantes.

A pesquisa tem como objetivo geral estabelecer paralelos e critérios para a implementação de elementos de design de jogos em software científico.

Dentro dos procedimentos da primeira etapa da pesquisa (realizada entre agosto e dezembro de 2015), foram gravadas em áudio reuniões e entrevistas com profissionais envolvidos no desenvolvimento do software SIVIEP.

É importante ressaltar que os nomes dos profissionais serão omitidos, na redação final da tese, com o objetivo de preservar a confidencialidade do grupo.

Como responsável pela coleta dos dados, estou disponível para esclarecer suas dúvidas sobre o projeto ou sobre sua participação, através do email: [chicoqueiroz@gmail.com](mailto:chicoqueiroz@gmail.com) ou pelo telefone celular: (21) 99409 1350.

Se preferir, você pode contatar a minha orientadora, Profa. Dra. Rejane Spitz, pelo email: [rejane@puc-rio.br](mailto:rejane@puc-rio.br)

Desde já, agradeço à sua participação que traz benefícios para a ampliação do conhecimento no campo do design.

## DECLARAÇÃO E CONSENTIMENTO

Declaro ter entendido os objetivos da pesquisa, concordo em participar e permito a publicação das transcrições das minhas falas ou entrevistas.

Vinicius Lopes Rodrigues Vinicius L. Rodrigues  
Participante

FRANCISCO OLIVEIRA DE QUEIROZ  
Pesquisador

Rio de Janeiro, 23 de maio de 2016.



## TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Eu, **Francisco Oliveira de Queiroz**, informo que estou realizando a Tese de Doutorado "Jogando o Software Científico" no Programa de Pós-graduação em Design do Departamento de Artes & Design da Pontifícia Universidade Católica do Rio de Janeiro. A pesquisa referente à tese conta com a participação de grupos de pesquisas integrantes ou ligados ao Tecgraf/PUC-Rio, do qual você é um dos integrantes.

A pesquisa tem como objetivo geral estabelecer paralelos e critérios para a implementação de elementos de design de jogos em software científico.

Dentro dos procedimentos da primeira etapa da pesquisa (realizada entre agosto e dezembro de 2015), foram gravadas em áudio reuniões e entrevistas com profissionais envolvidos no desenvolvimento do software SIVIEP.

É importante ressaltar que os nomes dos profissionais serão omitidos, na redação final da tese, com o objetivo de preservar a confidencialidade do grupo.


Como responsável pela coleta dos dados, estou disponível para esclarecer suas dúvidas sobre o projeto ou sobre sua participação, através do email: [chicoqueiroz@gmail.com](mailto:chicoqueiroz@gmail.com) ou pelo telefone celular: (21) 99409 1350.

Se preferir, você pode contatar a minha orientadora, Profa. Dra. Rejane Spitz, pelo email: [rejane@puc-rio.br](mailto:rejane@puc-rio.br)

Desde já, agradeço à sua participação que traz benefícios para a ampliação do conhecimento no campo do design.

## DECLARAÇÃO E CONSENTIMENTO

Declaro ter entendido os objetivos da pesquisa, concordo em participar e permito a publicação das transcrições das minhas falas ou entrevistas.

Thiagy Maioli Azevedo 

Participante

FRANCISCO OLIVEIRA DE QUEIROZ 

Pesquisador

Rio de Janeiro, 23 de Maio de 2016.



PONTIFÍCIA UNIVERSIDADE CATÓLICA  
DO RIO DE JANEIRO



## TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Eu, **Francisco Oliveira de Queiroz**, informo que estou realizando a Tese de Doutorado "Jogando o Software Científico" no Programa de Pós-graduação em Design do Departamento de Artes & Design da Pontifícia Universidade Católica do Rio de Janeiro. A pesquisa referente à tese conta com a participação de grupos de pesquisas integrantes ou ligados ao Tecgraf/PUC-Rio, do qual você é um dos integrantes.

A pesquisa tem como objetivo geral estabelecer paralelos e critérios para a implementação de elementos de design de jogos em software científico.

Dentro dos procedimentos da primeira etapa da pesquisa (realizada entre agosto e dezembro de 2015), foram gravadas em áudio reuniões e entrevistas com profissionais envolvidos no desenvolvimento do software SIVIEP.

É importante ressaltar que os nomes dos profissionais serão omitidos, na redação final da tese, com o objetivo de preservar a confidencialidade do grupo.

Como responsável pela coleta dos dados, estou disponível para esclarecer suas dúvidas sobre o projeto ou sobre sua participação, através do email: [chicoqueiroz@gmail.com](mailto:chicoqueiroz@gmail.com) ou pelo telefone celular: (21) 99409 1350.

Se preferir, você pode contatar a minha orientadora, Profa. Dra. Rejane Spitz, pelo email: [rejane@puc-rio.br](mailto:rejane@puc-rio.br)

Desde já, agradeço à sua participação que traz benefícios para a ampliação do conhecimento no campo do design.

## DECLARAÇÃO E CONSENTIMENTO

Declaro ter entendido os objetivos da pesquisa, concordo em participar e permito a publicação das transcrições das minhas falas ou entrevistas.

RODRIGO BRAGA PINHEIRO *Rodrigo Braga Pinheiro*  
Participante

FRANCISCO OLIVEIRA DE QUEIROZ *[assinatura]*  
Pesquisador

Rio de Janeiro, 23 de MAIO de 2016.

PONTIFÍCIA UNIVERSIDADE CATÓLICA  
DO RIO DE JANEIRO



## TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Eu, **Francisco Oliveira de Queiroz**, informo que estou realizando a Tese de Doutorado "Jogando o Software Científico" no Programa de Pós-graduação em Design do Departamento de Artes & Design da Pontifícia Universidade Católica do Rio de Janeiro. A pesquisa referente à tese conta com a participação de grupos de pesquisas integrantes ou ligados ao Tecgraf/PUC-Rio, do qual você é um dos integrantes.

A pesquisa tem como objetivo geral estabelecer paralelos e critérios para a implementação de elementos de design de jogos em software científico.

Dentro dos procedimentos da primeira etapa da pesquisa (realizada entre agosto e dezembro de 2015), foram gravadas em áudio reuniões e entrevistas com profissionais envolvidos no desenvolvimento do software SIVIEP.

É importante ressaltar que os nomes dos profissionais serão omitidos, na redação final da tese, com o objetivo de preservar a confidencialidade do grupo.

Como responsável pela coleta dos dados, estou disponível para esclarecer suas dúvidas sobre o projeto ou sobre sua participação, através do email: [chicoqueiroz@gmail.com](mailto:chicoqueiroz@gmail.com) ou pelo telefone celular: (21) 99409 1350.

Se preferir, você pode contatar a minha orientadora, Profa. Dra. Rejane Spitz, pelo email: [rejane@puc-rio.br](mailto:rejane@puc-rio.br)

Desde já, agradeço à sua participação que traz benefícios para a ampliação do conhecimento no campo do design.

## DECLARAÇÃO E CONSENTIMENTO

Declaro ter entendido os objetivos da pesquisa, concordo em participar e permito a publicação das transcrições das minhas falas ou entrevistas.

PABLO CARNEIRO ELIAS - Pablo Carneiro Elias  
Participante

FRANCISCO OLIVEIRA DE QUEIROZ  
Pesquisador

Rio de Janeiro, 23 de MAIO de 2016.





## TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Eu, **Francisco Oliveira de Queiroz**, informo que estou realizando a Tese de Doutorado "Jogando o Software Científico" no Programa de Pós-graduação em Design do Departamento de Artes & Design da Pontifícia Universidade Católica do Rio de Janeiro. A pesquisa referente à tese conta com a participação de grupos de pesquisas integrantes ou ligados ao Tecgraf/PUC-Rio, do qual você é um dos integrantes.

A pesquisa tem como objetivo geral estabelecer paralelos e critérios para a implementação de elementos de design de jogos em software científico.

Dentro dos procedimentos da primeira etapa da pesquisa (realizada entre agosto e dezembro de 2015), foram gravadas em áudio reuniões e entrevistas com profissionais envolvidos no desenvolvimento do software SIVIEP.

É importante ressaltar que os nomes dos profissionais serão omitidos, na redação final da tese, com o objetivo de preservar a confidencialidade do grupo.

Como responsável pela coleta dos dados, estou disponível para esclarecer suas dúvidas sobre o projeto ou sobre sua participação, através do email: [chicoqueiroz@gmail.com](mailto:chicoqueiroz@gmail.com) ou pelo telefone celular: (21) 99409 1350.

Se preferir, você pode contatar a minha orientadora, Profa. Dra. Rejane Spitz, pelo email: [rejane@puc-rio.br](mailto:rejane@puc-rio.br)

Desde já, agradeço à sua participação que traz benefícios para a ampliação do conhecimento no campo do design.

## DECLARAÇÃO E CONSENTIMENTO

Declaro ter entendido os objetivos da pesquisa, concordo em participar e permito a publicação das transcrições das minhas falas ou entrevistas.

RENATO CHERULLO DE OLIVEIRA

Participante

FRANCISCO OLIVEIRA DE QUEIROZ

Pesquisador

Rio de Janeiro, 25 de MAIO de 2016.

PONTIFÍCIA UNIVERSIDADE CATÓLICA  
DO RIO DE JANEIRO



## TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Eu, **Francisco Oliveira de Queiroz**, informo que estou realizando a Tese de Doutorado "Jogando o Software Científico" no Programa de Pós-graduação em Design do Departamento de Artes & Design da Pontifícia Universidade Católica do Rio de Janeiro. A pesquisa referente à tese conta com a participação de grupos de pesquisas integrantes ou ligados ao Tecgraf/PUC-Rio, do qual você é um dos integrantes.

A pesquisa tem como objetivo geral estabelecer paralelos e critérios para a implementação de elementos de design de jogos em software científico.

Dentro dos procedimentos da primeira etapa da pesquisa (realizada entre agosto e dezembro de 2015), foram gravadas em áudio reuniões e entrevistas com profissionais envolvidos no desenvolvimento do software SIVIEP.

É importante ressaltar que os nomes dos profissionais serão omitidos, na redação final da tese, com o objetivo de preservar a confidencialidade do grupo.

Como responsável pela coleta dos dados, estou disponível para esclarecer suas dúvidas sobre o projeto ou sobre sua participação, através do email: [chicoqueiroz@gmail.com](mailto:chicoqueiroz@gmail.com) ou pelo telefone celular: (21) 99409 1350.

Se preferir, você pode contatar a minha orientadora, Profa. Dra. Rejane Spitz, pelo email: [rejane@puc-rio.br](mailto:rejane@puc-rio.br)

Desde já, agradeço à sua participação que traz benefícios para a ampliação do conhecimento no campo do design.

## DECLARAÇÃO E CONSENTIMENTO

Declaro ter entendido os objetivos da pesquisa, concordo em participar e permito a publicação das transcrições das minhas falas ou entrevistas.

LUCIANO PEREIRA DOS REIS, Lucas Pereira  
Participante

FRANCISCO OLIVEIRA DE QUEIROZ  
Pesquisador

Rio de Janeiro, 10 de JUNHO de 2016.



## TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Eu, **Francisco Oliveira de Queiroz**, informo que estou realizando a Tese de Doutorado "Jogando o Software Científico" no Programa de Pós-graduação em Design do Departamento de Artes & Design da Pontifícia Universidade Católica do Rio de Janeiro. A pesquisa referente à tese conta com a participação de grupos de pesquisas integrantes ou ligados ao Tecgraf/PUC-Rio, do qual você é um dos integrantes.

A pesquisa tem como objetivo geral estabelecer paralelos e critérios para a implementação de elementos de design de jogos em software científico.

Dentro dos procedimentos da primeira etapa da pesquisa (realizada entre agosto e dezembro de 2015), foram gravadas em áudio reuniões e entrevistas com profissionais envolvidos no desenvolvimento do software SIVIEP.

É importante ressaltar que os nomes dos profissionais serão omitidos, na redação final da tese, com o objetivo de preservar a confidencialidade do grupo.


Como responsável pela coleta dos dados, estou disponível para esclarecer suas dúvidas sobre o projeto ou sobre sua participação, através do email: [chicoqueiroz@gmail.com](mailto:chicoqueiroz@gmail.com) ou pelo telefone celular: (21) 99409 1350.

Se preferir, você pode contatar a minha orientadora, Profa. Dra. Rejane Spitz, pelo email: [rejane@puc-rio.br](mailto:rejane@puc-rio.br)

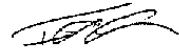
Desde já, agradeço à sua participação que traz benefícios para a ampliação do conhecimento no campo do design.

## DECLARAÇÃO E CONSENTIMENTO

Declaro ter entendido os objetivos da pesquisa, concordo em participar e permito a publicação das transcrições das minhas falas ou entrevistas.

EDUARDO GASPAR 

Participante

FRANCISCO OLIVEIRA DE QUEIROZ 

Pesquisador

Rio de Janeiro, 10 de JULHO de 2016.

## **7.2**

### **Atas das reuniões da pesquisa-ação**

As próximas páginas reproduzem as atas geradas durante a pesquisa-ação relatada no Capítulo 4.

## Ata de Reunião | Estudo em IHC – Siviep

Reunião 1 – 27/08/2015 - 16h15 | Tecgraf - 7º Andar

Solicitada por: Francisco Queiroz

Tipo de Reunião: Início de Pesquisa-Ação / Identificação de questões em design

Facilitador: Alberto Raposo

Anotador: Registro gravado em vídeo / notas de Francisco Queiroz

Presentes: Francisco Queiroz, Pablo Elías, Renato Cherullo, Rodrigo Pinheiro, Vinícius Rodrigues

## TEMAS DISCUTIDOS E NOTAS TOMADAS

**USOS DO SIVIEP:** Visão geral do projeto, formas de uso atuais, possíveis e futuras.

- O Siviep é um software desenvolvido para simulação integrada de empreendimentos.
- O principal uso do Siviep, atualmente, são demonstrações e passeios virtuais em empreendimentos.
- Programa tem, atualmente, praticamente um único usuário- que faz as exibições - além da própria equipe.
- No momento, é possível detectar problemas de tratamento de dados através do Siviep – quando óbvios o suficiente
- Para o futuro, projeta-se o uso do Siviep como software de apoio à tomada de decisões.
- ‘Insight’ a ser gerado pelo Siviep: problemas de produção em um nível macro.

**QUESTÕES/DIFICULDADES DE DESENVOLVIMENTO:** Num contexto geral, quais as maiores dificuldades no desenvolvimento do software

- Dificuldade na validação dos dados e acesso a especialistas que apresentem requisitos e dados.
- Dois casos de uso e duas ‘trilhas’ praticamente isoladas (modos desktop e imersivo), com desenvolvimento simultâneo.
- Visão da equipe de desenvolvimento nem sempre em sintonia com visão dos POs.
- Inovações são implementadas e, depois, apresentadas e – a partir daí, retrabalhadas em cima de feedback recebido.

**QUESTÕES/DIFICULDADES EM IHC:** Quais as maiores dificuldades no uso do produto

- Problemas de navegação foram reduzidos com solução multiscale, mas esquema de controle ainda é problemático para alguns usuários.
- Programa conta com dois modos de uso: **Desktop** e **Imersivo**. O modo Imersivo não contém diversas funcionalidades do modo desktop.
- Modo de imersão é menos eficiente na detecção de inconsistência de dados.
- Exibição de dados no modo imersão é um desafio.

- Project Owners resistem a propostas de renovação da interface inspiradas na linguagem de videogames. No entanto, gostam e aprovam resultados finais, quando implementados e apresentados.

### QUESTÕES PARA A PRESENTE PESQUISA IHC: Como esta pesquisa em IHC poderia enriquecer o produto?

- Trazer algumas funcionalidades do modo desktop para o modo imersivo; buscar unificação.
- Solucionar problemas como a visualização de elementos complexos (listas, tabelas, timelines) em modo imersivo.
- Manter acessibilidade a itens (menu contextual foi criticado por exigir mais cliques, ainda que seja mais adequado ao ambiente imersivo).
- Funcionalidades a serem pesquisadas: **Timeline** e/ou **Grafo**

### CONCLUSÕES E OBSERVAÇÕES

Diversas questões foram levantadas – algumas delas passíveis de tratamento através de pesquisa em IHC.

Concordou-se que a melhor aplicação da pesquisa seria na busca por soluções para usabilidade e visualização de dados em modo imersivo – particularmente no que se refere às funcionalidades de **Timeline** e **Grafo**.

Observou-se que uma melhora substancial nestas áreas poderia ajudar na divulgação do Siviep e sua aplicação em simulação integrada, atraindo possíveis colaboradores para obtenção / validação de dados e requisitos.

Concordou-se que a linguagem de jogos, particularmente aqueles pertencentes aos gêneros de simulação e estratégia, poderão oferecer insights para questões de funcionalidade e visualização de dados – e que melhorias obtidas através desta pesquisa seriam bem recebidas pelos POs e usuários.

### PRÓXIMOS PASSOS

- Instalar Siviep na máquina de Francisco Queiroz (feito).
- Elaborar e apresentar plano de pesquisa (incluindo avaliação de títulos selecionados) (Resp. Francisco Queiroz).

## Ata de Reunião | Estudo em IHC – Siviep

Reunião 2 – 16/09/2015 - 10h30 | Tecgraf - 7º Andar

Solicitada por: Francisco Queiroz

Tipo de Reunião: Acompanhamento / Brainstorm

Facilitador: Alberto Raposo

Anotador: Francisco Queiroz

Presentes: Francisco Queiroz, Pablo Elias, Renato Cherullo, Rodrigo Pinheiro, Vinícius Rodrigues

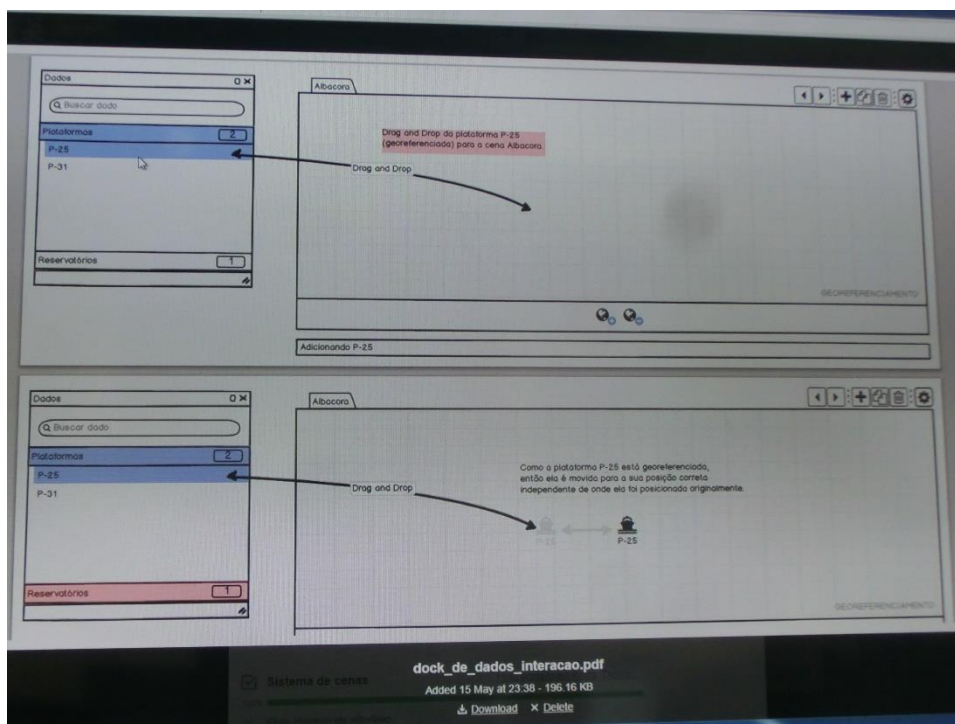
## TEMAS DISCUTIDOS E NOTAS TOMADAS

### RUMOS DO SIVIEP: Dificuldades e possibilidades

- Durante os últimos dias, a equipe de desenvolvimento tem questionado as chances reais de simulação integrada devido a inconsistências na relação entre dados de diferentes procedências; ausência de requisitos; e falta de trabalho em campo.
- Uma possibilidade seria utilizar o Siviep para analisar dados recebidos de fontes externas e gerar "alarmes" quando padrões de emergência fossem detectados.
- Cliente / Usuários tendem a utilizar o Siviep para visualizar atividades nos campos durante períodos de tempo específicos (ex: pressão do óleo entre setembro de 2012 e dezembro de 2014), mas ignoram relação entre dados de diferentes fontes.
- Todas as opções para desenvolvimento futuro contemplam o uso do Grafo e Timeline -- o que sugere fortemente que relevância desta pesquisa em IHC está mantida.

### REDESIGN ANTERIOR: Demonstração de protótipo produzido internamente

- Redesign do Siviep, para uso em produção, foi prototipado/documentado pela equipe internamente (em período anterior a esta pesquisa).
- Mudança radical de interface:
  - Dissociação 2D / 3D (e foco no 2D - modo imersivo quase suprimido)
  - 2 modos 2D: georeferenciado e esquemático.
  - Modo 'exame' para modelo 3D.
  - Menos 'docks', mais toolbars e abas.
  - Construção de cenários facilitada.
  - Funcionalidades inspiradas no jogo Sim City.



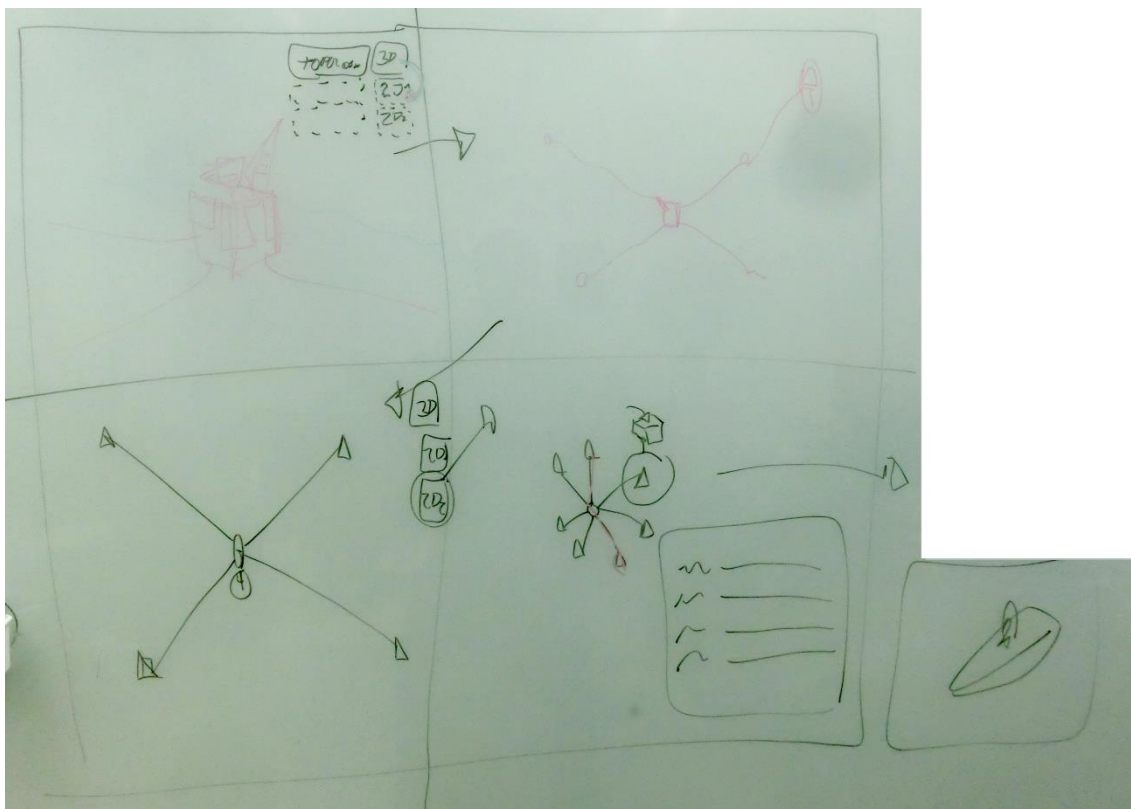
Mockup produzido por Rodrigo Pinheiro / Equipe Siviep

- A proposta de novo design foi descartada pelo cliente. Motivos para rejeição incluem:
  - Proposta de ícones fugindo ao padrão da gerência da companhia.
  - Ausência de visualização 2D incorporando elementos do universo 3D (ex: textura do terreno).
  - Retirada da ênfase sobre ambiente 3D pode ter sido um motivo adicional, embora não explicitado.

#### BRAINSTORM: Ideias geradas em cima das novas informações recebidas.

- Foram discutidas soluções para a funcionalidade do **Grafo**, apenas.
  - Menu / atalho de rápido acesso para troca de modo de visualização / ativação de visualização 2D (georeferenciada e esquemática).
  - Transição de modo 3D para 2D que valorize ambiente virtual.
  - Overlay de informações 2D sobre cenário.
  - Seleção de objeto através da visão 2D que possibilite acesso a informações e modo de exame de modelo 3D.
  - Possibilidade de uso de projetor extra como Painel de Controle para seleção de modo de visão.





Transição entre modos 3D, 2D georef., 2D esquemático e exame 3D, produzido durante reunião

## CONCLUSÕES E OBSERVAÇÕES

Dificuldades na obtenção de requisitos e impasses entre cliente e desenvolvedores no que se refere à autoria do projeto (visões, etc.) são condizentes com revisão bibliográfica em andamento. É aconselhável entrevista em separado com cliente / usuário do produto para maior entendimento de suas expectativas e visões para o produto (particularmente questões de IHC).

Modos de trabalho propostos no redesign descartado anteriormente parecem promissores, uma vez adaptados a esquemas de visualização e interação que valorizem o ambiente imersivo / virtual e respeitem convenções gráficas do ambiente de trabalho a que se destina.

Acertou-se o início da produção de protótipos (digitais ou não) para uma validação das ideias discutidas, além da pesquisa a títulos de jogos que contemplem modos de interação e temáticas aproximadas às do Siviep.

## PRÓXIMOS PASSOS

- Geração de protótipos (Resp. Francisco Queiroz)
- Entrevista com Luciano Reis (Resp. Francisco Queiroz)
- Pesquisa a esquemas de visualização 2D e 3D em jogos.

## Ata de Reunião | Estudo em IHC – Siviep

Reunião 3 – 17/09/2015 - 17h30 | Tecgraf - 7º Andar

Solicitada por: Francisco Queiroz

Tipo de Reunião: Entrevista com POs

Facilitador: Equipe Siviep

Anotador: Francisco Queiroz

Presentes: Francisco Queiroz, Luciano Reis, Eduardo Gaspar

## TEMAS DISCUTIDOS E NOTAS TOMADAS

### RUMOS DO SIVIEP: Questões de desenvolvimento e estado atual

- Foi apontada dificuldade de obtenção de dados e estudo de caso real.
- Não está claro como mostrar valor do software ao usuário – chave para este problema deve estar na visualização tridimensional de algo que ele está habituado a ver em 2D.
- Falta de produtos similares no mercado dificulta realização do projeto -- Simulação Integrada não está presente no dia-a-dia da produção da maior parte dos empreendimentos globais. O workflow atual não difere muito do mesmo de 30 anos atrás.
- Potencial a ser explorado em análise para detecção de problemas de produção.
- Intenção de não adicionar funcionalidades ao imersivo até que o desktop esteja pronto.
- Porém, pondera-se que novas funcionalidades na versão imersiva poderiam incentivar reuniões interdisciplinares e atrair usuários.

### INTERFACE E IMERSÃO: Considerações e observações

- O ideal seria unidade entre versão desktop e imersiva.
- Interface deve ser mais intuitiva para que usuários não precisem se adaptar a mais um novo programa.
- Atualmente, montar cenários é muito complicado
- Navegação entre objetos foi facilitada com grafo, mas pode melhorar ainda.
- Funcionalidade para incluir “ponto de interesse” seria bem-vinda
- Controle por voz é interessante, mas impraticável por questão de ruído em ambiente de trabalho
- Navegação por vista superior – similar a mini-mapas de jogos, seria bom, pois é como engenheiros estão habituados a ver empreendimentos. A partir desta visão 2D, explorar o 3D.
- Devices de apoio (projetor extra, tablets,ect) são interessantes, mas ideal seria ter toda interface virtual (até para facilitar implementação em headsets)
- Levar grafo para virtual é interessante, mas pode ser incipiente.
- Importância de se poder configurar opções de visualização (céu, mar, terreno, etc.) em modo imersivo.
- Menus radiais e contextuais são preferíveis à interfaces que exijam detecção de direção controle.

## JOGOS ETC. : Qualidades dos videogames e outros meios de comunicação

- Intimidade de POs com jogos eletrônicos varia:
  - Gaspar:
    - Gosta e joga jogos de estratégia e simulação como Sim City.
    - Demonstra interesse nas propriedades emergentes dos jogos.
    - Relaciona jogos a potencial criativo de crianças, que identifica como “menos limitadas” que adultos, sob alguns aspectos.
    - Capaz de jogar por horas em estado de fluxo.
  - Luciano:
    - Tem pouco contato com a área – fato que atribui à geração de que faz parte.
    - Tem crescente interesse no assunto, principalmente por influência do filho de cinco anos. Declara se interessar por Minecraft e Sim City.
    - Mesmo sem intimidade com o meio, reconhece alcance de jogos como Minecraft, sobre o qual tem pesquisado, em iniciativas de cunho científico e educativo.
    - Surpreende-se com engajamento de jogadores na construção de elementos do jogo.
    - Parece interessado em jogos com objetivos claros (fala sobre dois modos de jogo do minecraft: “o exploração e o survival. O simulação é livre, você vai e constrói, mas o objetivo mesmo é o survival”).
    - Pesquisa jogabilidade de Minecraft em vídeos e livros – atenta para o fato de que muitos jogadores recorrem a esta prática.
    - Prefere títulos em que seja possível navegar por ambientes.
    - Sem hábito de jogar (algo que acredita que ocorrerá no futuro), Luciano declara preferência a aproveitar tempo livre assistindo a filmes na televisão (atividade que relaciona a estado de fluxo).

## CONCLUSÕES E OBSERVAÇÕES

Visão dos POs parece, de modo geral, alinhada equipe de desenvolvimento – tanto nas dificuldades quanto nas necessidades do projeto. Diferença parece estar em detalhes que, uma vez observados, podem resultar em solução de interface capaz de atender a expectativas dos dois grupos (POs e Equipe de desenvolvimento). Alguns destes detalhes são:

- Acessibilidade para usuários não totalmente acostumados a convenções de videogame.
- Observar que se passa por momento de transição entre modos de interface e, também workflow.
- Grafo deve permitir, além de acesso a funções adicionais, navegação pelo cenário.

## PRÓXIMOS PASSOS

- Finalização e apresentação de protótipo (Resp. Francisco Queiroz)
- Definição de métricas (Francisco Queiroz, Alberto Raposo e Equipe Siviep)

## Ata de Reunião | Estudo em IHC – Siviep

Reunião 4 – 2/10/2015 - 17h | Tecgraf – 7º Andar

Solicitada por: Francisco Queiroz

Tipo de Reunião: Apresentação de WIP – Protótipo

Facilitador: Equipe Sivieo

Anotador: Francisco Queiroz

Presentes: Francisco Queiroz, Rodrigo Braga Pinheiro

## TEMAS DISCUTIDOS E NOTAS TOMADAS

### TESTES COM PROTÓTIPO/MOCKUP: Questões de usabilidade

Rodrigo Braga Pinheiro testou o protótipo / mockup. Ao longo do teste, os seguintes pontos foram discutidos:

- **Ativação / Exibição do Painel:** Usuário considerou confuso apertar “1” e ter o painel visível, mas não ativo.
- **Barra Lateral:**
  - **Ícones:** Usuário achou ícone de configurações do painel ambíguo, tendo dificuldade para encontra-lo
  - **Retirada:** Usuário sugeriu substituição de menu lateral por painel principal, eliminando ambiguidade do uso de setas para escolha de opção, e tornando mais óbvio o uso do botão Home do controller.
- **Painel Configurações:**
  - **Setas:** Usuário achou mais intuitivo usar setas horizontais para habilitação/desabilitação dos toggle switches, escala vertical e opções no estilo dropdown (desde que substituídos por botões no estilo < [nome da opção] >
  - **Breadcrumbs:** Usuário sugeriu adicionar breadcrumbs ao topo do painel de configurações
- **Painel Grafo**
  - **Menus contextuais:** Usuário achou número de opções em menu radial excessivo, sugerindo menus diferentes caso cursor esteja acima de unidade ou em parte vazia do Grafo
  - **Nomes de opções:** Usuário achou nome de opções do menu contextual, em alguns casos, ambíguos.
- **Modo PiP:** usuário sugeriu de modo PiP (Picture-in-Picture) para redução do viewport 3D e expansão do restante da interface.
- **Instruções na Tela:** Usuário considera que o uso de instruções na tela deveria ser, idealmente, desnecessário em um sistema intuitivo

## CONCLUSÕES E OBSERVAÇÕES

- Teste revelou inconsistências e pontos passíveis de melhoria.
- De modo geral, sugestões aproximam interface ainda mais de conceitos de SmartTv, e não, necessariamente, jogos.

## PRÓXIMOS PASSOS

- Nova versão de protótipo incorporando sugestões (Francisco Queiroz)

## Ata de Reunião | Estudo em IHC – Siviep

Reunião 5 – 23/10/2015 - 14h | NVC CENPES - Sala de Projeção

Solicitada por: Francisco Queiroz

Tipo de Reunião: Acompanhamento com POs e Equipe Siviep

Facilitador: Luciano Reies e Equipe Siviep

Anotador: Francisco Queiroz

Presentes: Francisco Queiroz, Luciano Reis, Eduardo Gaspar, Pablo Carneiro Elias,

Vinicius Rodrigues

## TEMAS DISCUTIDOS E NOTAS TOMADAS

### TESTES COM PROTÓTIPO/MOCKUP: Questões de usabilidade

Gaspar e Luciano (POs) conseguiram, de modo geral, completar as tarefas sugeridas (navegar pelo painel e utilizar funções de configuração, timeline, e grafo), com algumas observações:

- **Painel Timeline:** Bem recebido, com alguma explicação necessária para uso da função de “Marcar Início” e “Marcar Fim”.
- **Painel Configurações:** Um usuário (Gaspar) teve a reação de voltar ao ‘breadcrumbs’ do painel para voltar ao painel Home -- algo que poderia ser implementado. Adicionalmente, indicou o botão Home (opção correta original) como outra solução.
- **Indicação de Instruções:** Indicação de usos dos botões 1 e 2 (Ativar e Exibir Painel) pode ter posicionamento melhorado: Canto inferior direito se mostrou obstruído para um dos usuários (Luciano), sentado atrás de uma mesa.
- **Consistência:** Gaspar e Pablo expressaram preferência por sistemas onde “A” sempre funciona como “Ir” e “B”, como “voltar” – Nintendo, dentre outras companhias, se utiliza bastante desse método.
- **Painel Grafo:** Duas questões sobre o menu Grafo receberam destaque:
  - A – Botões ‘+’ e ‘-’ foram considerados pouco intuitivos como controle de zoom, uma vez que usuários se acostumaram a utilizar ‘A’ e ‘B’ para controlar função equivalente em navegação 3D. Como estes botões estão associados ao menu radial de contexto do Grafo, sua utilização não é possível. Melhor solução até o momento, sugerida por Vinicius, seria trocar zoom ‘contínuo’ por valores discretizados (0.5, 1, 1.5, 2 vezes), considerando que função de zoom seria pouco utilizada.
  - B – Menus contextuais diferentes para cursor sobre unidade e cursor sobre grafo vazio foram apontados como pouco intuitivos. Possível solução seria reunificar sob forma de sub-menus radiais.
- **Navegação Temporal em modo imersivo:** Inspirada no jogo Braid, esta funcionalidade foi a mais bem recebida de todas, e candidata a implementação.

### NOVOS RUMOS DO SIVIEP

Foco do Siviep para as próximas semanas / meses, inspirado em discussões prévias e sessões de demonstração para público externo, será voltado à construção de cenários de empreendimentos para simulação de alternativas. O workflow sugerido, até, o momento, seria:

- 1 – Escolher / adicionar Reservatório
- 2 – Selecionar local do Poço
- 3 – Adicionar Plataforma
- 4 – Adicionar Manifold
- 5 – Desenhar Dutos

## CONCLUSÕES E OBSERVAÇÕES

- Protótipo se mostrou, de forma geral, adequado para uso em modo imersivo. No entanto, novos rumos para o programa colocam a parte mais substancial do protótipo em uma prioridade muito baixa.
- Parece recomendável, no entanto, não descartar totalmente ideias propostas, pois podem voltar a ser vistas no futuro.
- Dado o sucesso do controle de tempo, parece ser melhor a geração/apresentação de muitas ideias por protótipo, ao invés de uma busca por refinamento.
- A natureza e ambiente de trabalho sugerem que protótipos interativos poderiam ser mais eficazes na demonstração de funcionalidades.
- No que se refere a protótipos, decisões de design tomadas em conjunto por POs e Equipe de Desenvolvimento parecem ser um campo melhor para atuação.

## PRÓXIMOS PASSOS

- Prototipagem (totalmente ou parcialmente interativo) de funcionalidades para construção de cenário (Resp. Francisco Queiroz)
- Pesquisa a jogos de construção de cenários (Resp. Francisco Queiroz)

## Ata de Reunião | Estudo em IHC – Siviep

Reunião 6 – 06/11/2015 - 15h30 | Tecgraf - 7º Andar

Solicitada por: Francisco Queiroz

Tipo de Reunião: Acompanhamento / Brainstorm

Facilitador: Alberto Raposo

Anotador: Francisco Queiroz

Presentes: Francisco Queiroz, Pablo Elias, Rodrigo Pinheiro, Thiago Maioli, Vinícius Rodrigues

## TEMAS DISCUTIDOS E NOTAS TOMADAS

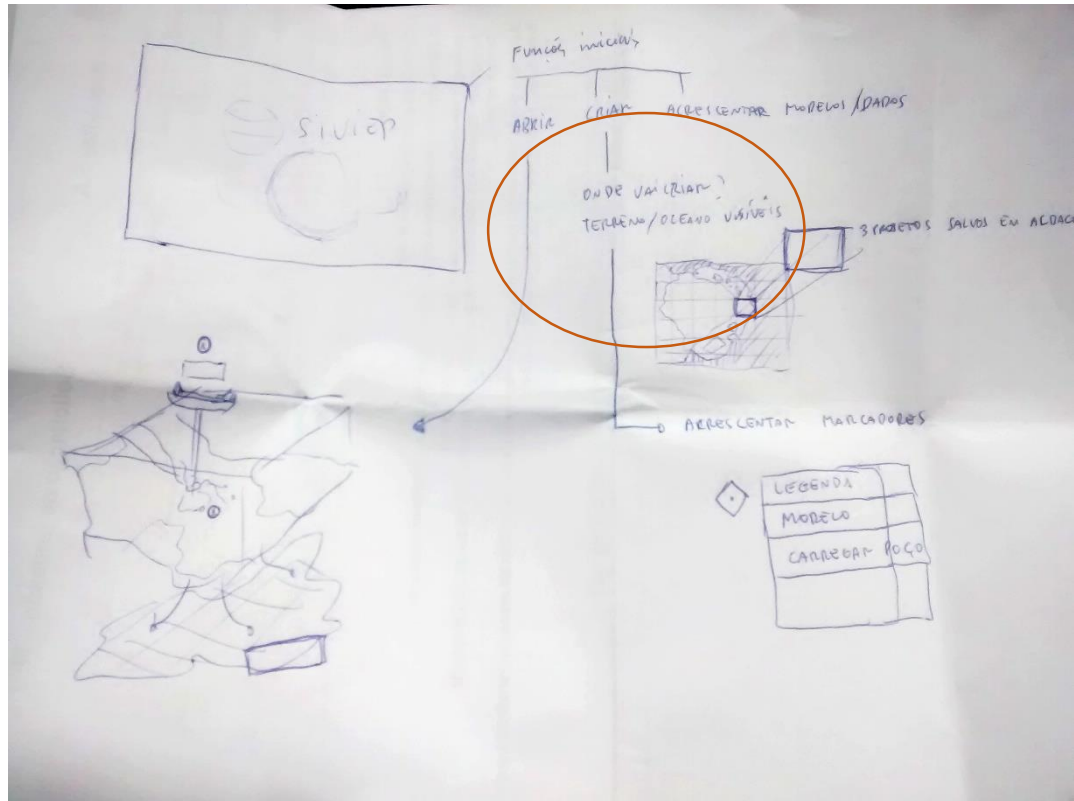
### APRESENTAÇÃO DE RECAPITULAÇÃO DA PESQUISA: Teoria e Prática

- Baseado nos slides Siviep\_recap.pptx, foram apresentadas as bases da pesquisa em seus aspectos teóricos (ganho de conhecimento na aplicação de design de jogos em software científico) e práticos (melhoria e implementação de funcionalidades no Siviep).
- Apresentação dos jogos testados para pesquisa até o momento, selecionados de acordo com critérios de relevância:
  - Proximidade temática
  - Funcionalidade pretendida
  - Similaridade de Input e hardware de suporte
- Recapitulação dos passos anteriores da pesquisa.
- Demonstração dos menus para adicionar elementos e editar elementos.
- Estabelecimento dos próximos passos a serem idealizados dentro do cenário pretendido (criação e edição de terreno).

### FERRAMENTA DE EDIÇÃO: Nova ferramenta para criação de cenário

- Maior parte da ação se dará no leito do oceano.
- Ferramenta de desenho de dutos:
  - Alternativas:
    - Desenhado à mão livre
    - Controle bezier sobre pontos de curva
    - 'Fillet'/'chamfer' sobre pontos
  - Necessidades:
    - Simplicidade
    - Similaridade com ferramentas utilizadas atualmente por profissionais
  - Funcionalidades extra interessantes:
    - Medição de distância e/ou material do duto
  - Foi observada a ferramenta de edição de dutos produzida por outra equipe do mesmo grupo de VR, implementada como complemento à engine Unity para a produção de cenas para simulação.
- Terreno:
  - Foi acertado que haverá um processo de *seleção*, e não *criação*, de terreno ao início do uso do programa.

- Foi especulado que seria preferível um sistema tipo *wizard* para seleção de terreno, poço e ANM associado, ao invés de ferramentas de criação sobre um cenário totalmente vazio.
- Uma outra ferramenta a ser possivelmente incorporada seria uma 'régua' para medição de distância (levando em consideração batimetria)



Sugestão para seleção de terreno, rascunhada previamente por Thiago Maioli

- Discutiu-se bastante, sem conclusão definitiva, se esforço na ferramenta de geração de cenários deve ser torná-la mais 'livre' (para rascunhos rápidos) ou 'precisa' e adequada a regras de negócios e requisitos (ainda não especificados). Modo 'playground' seria ponto de partida para inserir regras aos poucos.
- Outro ponto bastante discutido é se a edição de dutos deve ser feita em visão prioritariamente 2D ou 3D.

#### DESIGN E IDENTIDADE VISUAL: Manutenção e coerência

- Uma série de ícones e material gráfico produzidos anteriormente por Thiago Maioli está disponível no Confluence do grupo.



## CONCLUSÕES E OBSERVAÇÕES

O design do software está se tornando mais próximo a uma experiência do tipo ‘sandbox’, com alto grau de liberdade e poucas restrições – em grande parte devido à dificuldade na obtenção de requisitos. A equipe tenta usar isso a seu favor, desenvolvendo um ambiente onde regras de negócios não estão previamente implementadas. Neste caso, precisão e realismo serão, num primeiro momento, postos de lado em favor de simplicidade / facilidade de uso, e regras de negócios seriam inseridas aos poucos, à medida em que requisitos e casos de uso reais forem apresentados.

Curiosamente, o design inicial do Siviep (divisão em abas para “Projeto”, “Cena”) é parcialmente inspirado na engine de jogos Unity. Há, então, uma grande mudança em curso, em termos de interface.

Além disso, na ausência de requisitos ligados ao campo de trabalho, grande parte da discussão se concentra em possíveis dificuldades e limitações na implementação de ideias apresentadas durante brainstorm.

A funcionalidade sugerida para contagem de material / comprimento de dutos pode aproximar a ferramenta de um de seus ideais – suporte a decisão. A gestão de recursos como este, por sinal, é característica de processos de gamificação clássicos, onde atividades estruturadas tem resultados quantificáveis que podem ser mais ou menos desejáveis.

Ainda sobre o modo ‘sandbox’, entende-se que seria necessário, e positivo, restringir liberdade em etapas onde modo playground pudesse ser confuso, como na configuração inicial do terreno e equipamentos associados.

A presença de outro designer (Thiago Maioli, em sua primeira participação) se mostrou muito benéfica – além do rascunho apresentado, Maioli sugeriu funcionalidades (medição de dutos / material) e indicou a existência de material gráfico passível de ser utilizado. O material está armazenado no sistema *Confluence* utilizado pela equipe. No mais, parece ser essencial a presença de um designer que avalie, absorva, e dê continuidade aos métodos desenvolvidos por essa pesquisa após o encerramento desse ciclo.

## PRÓXIMOS PASSOS

A necessidade mais imediata da pesquisa é a configuração inicial do cenário (terreno). Para isso, será necessário:

- Pesquisa a esquemas de configuração de cenários em jogos (Resp. Francisco Queiroz, Thiago Maioli)
- Geração de protótipos de configuração inicial de cenário (Resp. Francisco Queiroz, Thiago Maioli)

Reunião 7 – 13/11/2015 - 15h30 | Tecgraf - 7º Andar

Tipo de Reunião: Acompanhamento / Brainstorm

Anotador: Francisco Queiroz

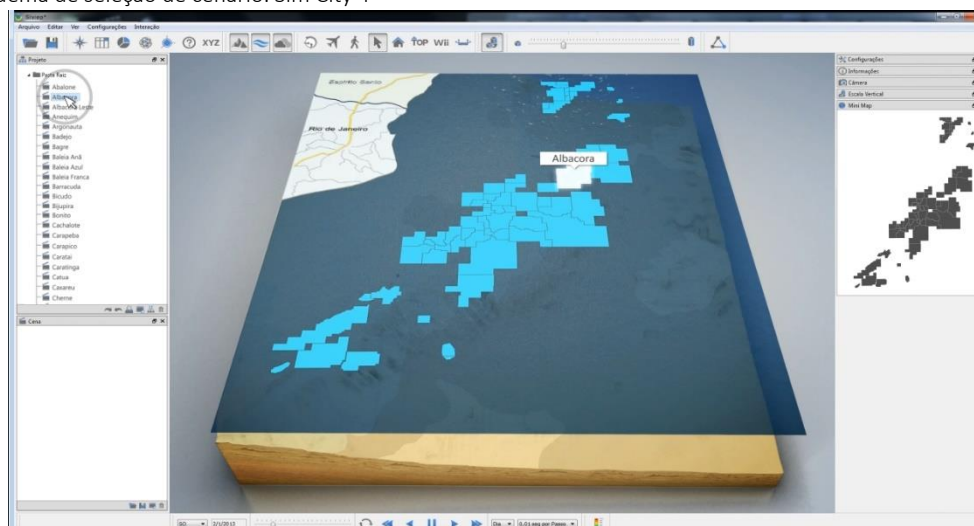
Presentes: Francisco Queiroz, Pablo Elias, Renato Cherullo, Rodrigo Pinheiro, Thiago Maioli, Vinícius Rodrigues

## TELAS DE SELEÇÃO INICIAL

- Foram debatidos modelos de tela de seleção global (bacia de campos) e visão do campo (Albacora), inspirados por soluções apresentadas pelos jogos Sim City 4 e Cities in Motion. Optou-se por adotar funcionalidade semelhante à de Sim City 4, cujo modelo de seleção global, coincidentemente, é bem semelhante a uma proposta realizada em 2014.



Esquema de seleção de cenário. Sim City 4



Conceito gerado em 2014 por Thiago Maioli

### ESCOPO: Delimitação das próximas etapas a serem produzidas

- Decidiu-se limitar a discussão a um escopo mais realista e gerenciável a curto prazo: Seleção de bloco e importação de dados (poços, reservatórios, etc) georeferenciados preexistentes.
- Serão prototipadas 5 etapas:

ETAPA 1 Bloco do terreno completo por onde usuário deverá selecionar ringfence do bloco de albacora
ETAPA 2 Bloco de Albacora por onde o usuário deverá importar seus dados georeferenciados através do menu radial
ETAPA 3 Menu radial para seleção da opção “importar dados”
ETAPA 4 Tela para seleção e importação dos dados
ETAPA 5 Elementos importados corretamente posicionados em cena

### CONCLUSÕES E OBSERVAÇÕES

A delimitação do escopo parece acertada ao não incluir elementos que, por sua complexidade, devem ser deixados para uma etapa posterior (criação e posicionamento livre de equipamentos, por exemplo).

### PRÓXIMOS PASSOS

Executar as Etapas 1 a 5 em protótipo interativo:

- Produção coleta e organização de material 2D e 3D. (Resp. Thiago Maioli, Francisco Queiroz)
- Geração de versão interativa (Resp. Francisco Queiroz)

## Ata de Reunião | Estudo em IHC – Siviep

Reunião 8 – 30/11/2015 - 14h30 | Tecgraf - 7º Andar

Solicitada por: Francisco Queiroz

Tipo de Reunião: Acompanhamento

Facilitador: Alberto Raposo

Anotador: Francisco Queiroz

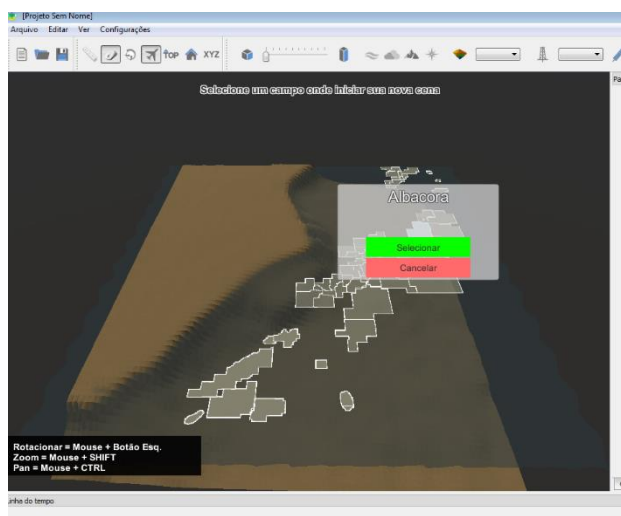
Presentes: Francisco Queiroz, Pablo Elias, Renato Cherullo, Rodrigo Pinheiro

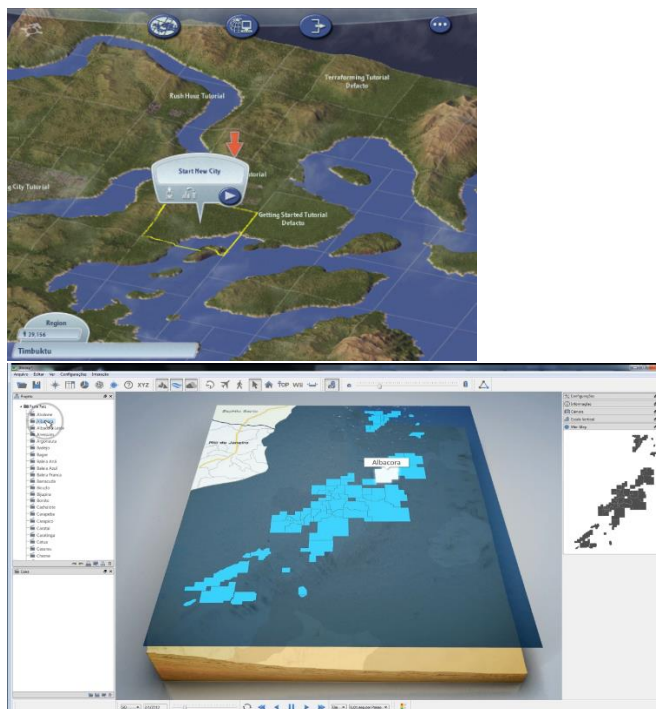
## TEMAS DISCUTIDOS

### ELABORAÇÃO / DEMONSTRAÇÃO DE PROTÓTIPO INTERATIVO

- A partir das etapas identificadas na reunião anterior, foi produzido um protótipo interativo digital, disponível na url <http://webserver2.tecgraf.puc-rio.br/~chico/siviep/>
- ETAPA 1 - Bloco do terreno completo por onde usuário deverá selecionar o *ringfence* do bloco de Albacora.

Como anteriormente discutido, esta etapa inicial foi inspirada por soluções apresentadas em jogos como *Sim City 4* e, também, propostas anteriores de *redesign* do próprio Siviep.



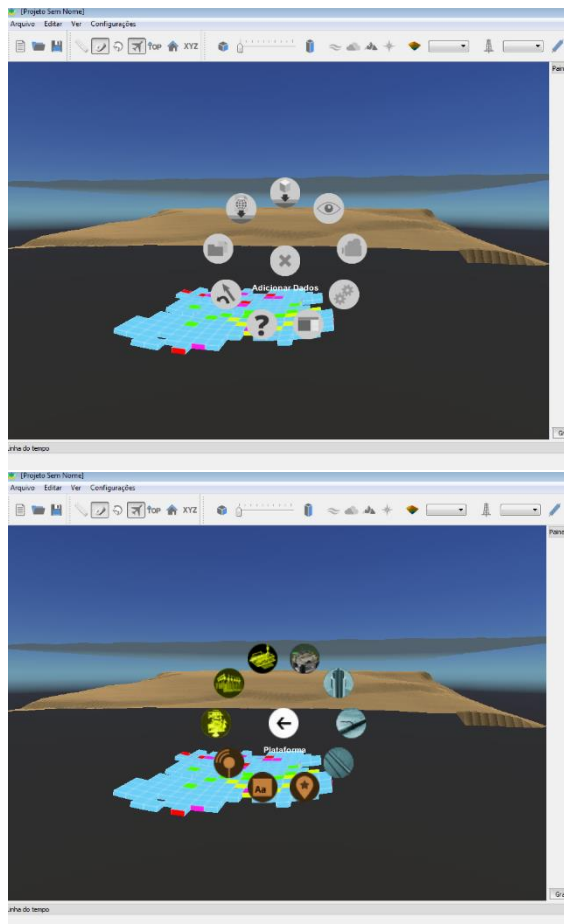


Sim City 4, da Maxis (esq.) e Tela-conceito do Siviep gerada em 2014 por Thiago Maioli (dir.)

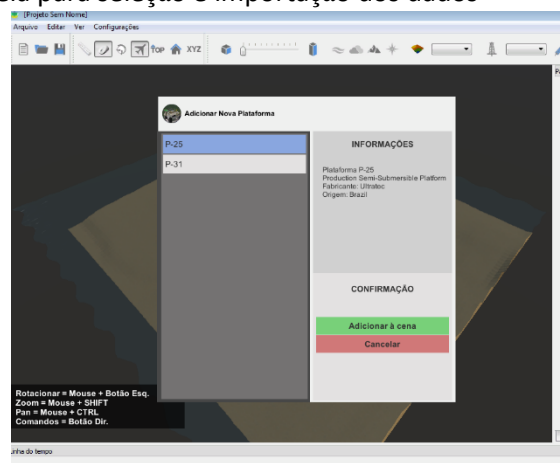
- ETAPA 2 - Bloco de Albacora por onde o usuário deverá importar seus dados georefenciados através do menu radial.

Enquanto o menu radial havia sido originalmente sugerido pelos POs e inspirada em títulos como *Mass Effect*, A tela de seleção de dados foi inspirada por soluções apresentadas em jogos como *The Elder Scrolls V: Skyrim*, porém adaptadas para necessidades do software (como a revisão dos dados selecionados).





- ETAPA 4 - Tela para seleção e importação dos dados

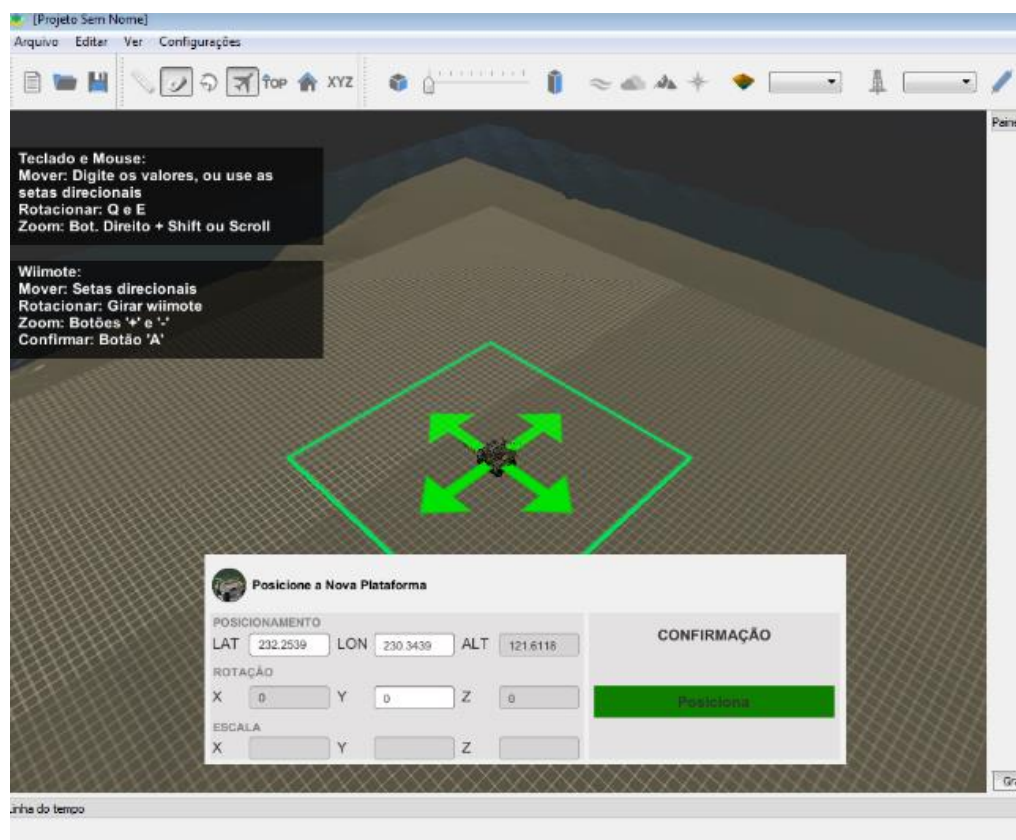


- ETAPA 5 - Tela para posicionamento dos dados em cena

A funcionalidade de posicionamento sobre um grid, enquanto inspirada em funcionalidades semelhantes de jogos como *SimCity Build It* e *Farmville*, foi adaptada para a necessidades de precisão do Siviep. Além da opção de se digitar valores específicos nos inputs de texto, uma outra solução foi elaborada para uso em modo imersivo: a proximidade da câmera com o objeto tem influência sobre a extensão do deslocamento. Por exemplo: Enquanto um toque na seta direcional moveria o objeto 100 unidades na visão panorâmica, aproximar a câmera do objeto gradualmente através da função de zoom alteraria aquele valor para 10, 1, 0.1 e



0.01 unidades. Esta funcionalidade foi, de certa forma, inspirada na solução para navegação multi-escala adotada pelo Siviep – o que sugere que poderá ser recebida com naturalidade pelos usuários.







Sistemas de posicionamento sobre grid de Sim City Build It (esq.) e FarmVille (dir.).

## CONCLUSÕES E OBSERVAÇÕES

Esta iteração conclui a etapa de elaboração de uma proposta de design inspirado e informado por conceitos e funcionalidades de jogos eletrônicos.

O protótipo interativo apresentado foi bem recebido pela equipe de desenvolvimento, restando agora sua apresentação aos POs para que as propostas sejam, ou não, aprovadas para implementação efetiva.

Do ponto de vista da investigação teórica, acredito que alguns pontos muito importantes foram levantados:

### 1 – Quanto à natureza do processo de gamificação

Costuma-se definir gamificação como ‘o uso de elementos de design de jogos em contextos de não-jogo’ (Deterding et al.). A gamificação é, portanto, primariamente associada a atividades lúdicas estruturadas - jogos, mas não necessariamente a atividades lúdicas livres – brincadeiras. Esta divisão entre jogo e brinquedo, *ludus* e *paidia* (Caillois), *game* e *nongame* (Queiroz), *Toyplay* e *Goalplay* (Madsen et al.), tem sido reportada em gamificação através dos termos **gameful** e **playful** (McGonigal). O termo *playful design*, no entanto, já foi utilizado por John Ferrara para descrever a

aplicação de práticas de gamificação (estruturada) em *software* convencional. Por outro lado, Dana Maria Popa definiu gamificação como um processo a ser baseado em **‘referências transpostas de jogos para outras mídias**, especialmente as digitais, que possam prover experiências emocionais apropriadas aos usuários’ – e não necessariamente mais, ou menos, estruturado.

No contexto desta pesquisa, a aplicação do design de jogos no software científico se enquadra na definição de gamificação proposta por Popa, uma vez que não há – até o momento – elementos de design característicos jogos enquanto atividades lúdicas estruturadas, como condições de vitória, recompensas, contagem de pontos, etc. Observa-se, no entanto, que isto ocorre, em parte, pela ausência de **regras de negócios** implementadas no software. Como exemplo, poderíamos considerar que a implementação do medidor de distância e material de dutos sugerido por Maioli (Ata de Reunião 6 – 6/11/2015) abriria novas possibilidades para *gameful* design. De fato, a **mudança do escopo do projeto** – privilegiando a livre criação de cenários e não necessariamente a simulação integrada do empreendimento – **reforçou o posicionamento deste processo enquanto playful**. Vale ainda ressaltar que o jogo *Sim City*, precursor do gênero de simulação que deu origem a diversos títulos utilizados como inspiração, foi definido por Will Wright, seu criador, não como um jogo, mas como um *software toy*.

Finalmente, cabe ressaltar que grande parte do esforço da pesquisa se deu no sentido de reproduzir, neste software científico, **mecânicas de controle, convenções de interface, e, também, uma qualidade de interação semelhante à dos jogos eletrônicos** de referência. Como aponta Katherine Isbister, melhorias em interface podem e devem ser feitas à partir do conceito de *game feel*, apresentado por Steve Swink, onde a imersão do usuário é valorizada pelo alta sensação de controle em tempo real de objetos em ambientes virtuais com alto nível de acabamento. Esta transposição de elementos e conceitos de design de jogos para design de interação é característica, também, do método de gamificação baseado em *Skill Atoms* desenvolvido por Sebastian Deterding.

## 2 – Quanto a particularidades do software científico

Como previsto, algumas características associadas ao software científico influenciaram e nortearam o processo de design. Dentre as quais, destacamos:

a) **Dificuldade na definição de requisitos** - Em software científico, requisitos são descritos como como 'voláteis' (Rebecca Sanders) e 'emergentes' (Judith Segal e Chris Morris). No caso desta pesquisa, mudanças no escopo do software exigiram, em determinado momento, que se abandonasse o redesign de um conjunto de funcionalidades para que se atendesse a outras necessidades, agora consideradas prioritárias. Isso sugere que **o processo de design deve ser flexível o bastante para abarcar mudanças** no escopo e prioridades, mas ao mesmo tempo se manter coerente para que ideias inicialmente descartadas possam ser revisitadas no futuro.

b) **Atenção para facilidade de uso** - De acordo com Jane Webster e Joseph Martocchio, **a faixa etária do usuário pode representar uma barreira** na aquisição de habilidades em

software e ambientes digitais. Ao mesmo tempo, supõe-se que cientistas mais jovens sejam mais abertos, ou mesmo ansiosos, por ferramentas que apresentem a sofisticação dos *videogames* (Javahery et al.). Apesar da média de idade dos indivíduos familiarizados com *games* ter aumentado nos últimos anos, foi preciso, nesta pesquisa, **contemplar o uso do por sistema por usuários de idade acima da média de jogadores** – compatível com o de profissionais em cargos gerenciais que venham a utilizar a ferramenta. Neste sentido, foi valiosa a presença de Luciano Reis, PO, cujo perfil serviu como contraponto ao perfil da equipe de desenvolvimento, mais jovem e altamente familiarizada com jogos eletrônicos. Dentro dos frameworks de design de jogos, a **facilidade de uso, ajuda ao usuário, e acesso por diversas faixas etárias são particularmente valorizadas em *Gamenics***, desenvolvido por Akihiro Saito na Nintendo. Não por acaso, vários jogos desenvolvidos para plataformas da Nintendo foram pesquisadas, o que se refletiu em algumas características do protótipo, como a presença constante de instruções de controle na tela.

c) **Atenção para precisão e qualidade dos dados** – Software científico deve ser preciso e correto (Hamming; Hatton; Sanders). Aos videogames, por outro lado, se permitem simplificações necessárias para a jogabilidade pretendida. Um exemplo claro encontrado durante a produção do protótipo foi a funcionalidade de posicionamento de uma plataforma. Originalmente baseado em títulos como *Sim City* e *Farmville*, o sistema de posicionamento por setas sofreu duas alterações para que se alcançasse a precisão necessária: (1) **inclusão de uma solução similar a de softwares convencionais**: campos de texto por onde digitar os dados; e (2) **inclusão de mecânica original adicional**, onde o grau de precisão da ferramenta de posicionamento aumenta de acordo com a proximidade da câmera em relação ao objeto sendo posicionado. Esta última modificação foi, de fato, inspirada na navegação multiescala implementada no próprio *Sivie* (Taunay et al.).

d) **Estímulo à geração de *insight*** – Mais do que fornecer dados à pesquisa, a função do software científico é gerar *insight* em seus usuários (Hamming; Kelly). A dificuldade na obtenção de *insight* utilizando ferramentas de software é, para Aragon et al., um desafio atualmente enfrentado por várias frentes de desenvolvimento científico. Holzinger et al. demonstraram como a visualização interativa de dados médicos permitiu a geração de *insights* por parte dos pesquisadores. No contexto desta pesquisa, contemplaram-se as formas de promover *insights* através do design:

l) **Grafos redesenhados** – Discutiu-se a hipótese de redesenhar os grafos sobrepostos às unidades para que exibissem informações relativas a um intervalo de tempo, e não somente de um único momento. Embora característica de design de visualização de dados, e não necessariamente de jogos, esta funcionalidade foi encontrada no jogo *Cities in Motion 2*.



Grafos da versão atual do Siviep (esq.) e Cities in Motion 2 (dir.)

II) **Controle do tempo** – No contexto da simulação integrada em modo imersivo, foi sugerido um sistema para controle de tempo de cena que incluísse, além do comando para executar/pausar, opções de avanço / retrocesso manual através das setas direcionais do Wiimote. Tal funcionalidade, embora inspirada no jogo de plataforma *Braid*. No entanto, como observou Pablo Elias, também é comparável à função *scrub* disponível em interfaces para execução de vídeos, como no caso da Apple Tv, por exemplo. Assim como no caso dos grafos, é interessante perceber como os jogos eletrônicos possuem um repertório variado de soluções disponíveis em outros meios – o que atesta a riqueza dos videogames como fonte de consulta.



Jogo Braid

III) **Sandbox mode** - Embora não determinado pela pesquisa, a criação de um modo de livre criação de cenários se assemelha em alguns aspectos à jogabilidade livre (*sandbox mode*) encontrado em títulos como *Sim City* e *RollerCoaster Tycoon*. No caso do Siviep, a ausência de implementação de regras de negócios torna o programa (ao

menos temporariamente) menos propenso a centralizar tomadas de decisão. Por outro lado, o usuário poderá - após internalizar as regras de negócios - experimentar soluções de forma mais irrestrita e, possivelmente, chegar a soluções criativas que se adequem, posteriormente, a todas as normas necessárias.

### 3 – Quanto ao processo inicial de geração de ideias

a) **Identificação e seleção de oportunidades** – Recomenda-se que a identificação de necessidades e oportunidades para melhorias em questões de interação seja feita coletivamente entre os diversos atores (desenvolvedores, designers, cientistas, POs, etc.). A dificuldade na definição de requisitos, combinada com diferenças entre as visões dos grupos envolvidos, exige que decisões de design sejam tomadas em comum acordo, preferencialmente em reuniões presenciais com representantes de cada grupo.

b) **Busca por referências** -- Quanto à seleção de títulos para análise, três critérios parecem especialmente importantes:

- I. **Proximidade temática** – Jogos do gênero simulação/*tycoon* foram escolhas apropriadas por simular a construção e gerenciamento de empreendimentos – algo bem próximo ao uso pretendido do software em questão. Isso **sugere que as mecânicas e funcionalidades e soluções apresentadas pelo design dos jogos serão adequadas** ao software.
- II. **Funcionalidade Pretendida** – Não havendo funcionalidade similar em jogos com proximidade temática, **títulos que não guardam semelhança com o software podem ser fontes de referência**. No caso desta pesquisa, o controle do tempo da simulação integrada inspirada pelo jogo *Braid* é um bom exemplo.
- III. **Equipamentos** – No caso de softwares que exijam o uso de equipamentos especiais ou específicos, vale a pena pesquisar **jogos que utilizem estas mesmas formas de input e output**. Para esta pesquisa, buscou-se soluções que fossem adequadas ao controle *Wii mote*, utilizado no modo imersivo, através da pesquisa a títulos lançados para plataformas compatíveis.

Vale a pena notar que os três critérios levam em consideração o valor das convenções de interação criadas por outros títulos e adotadas por similares ao longo dos anos.

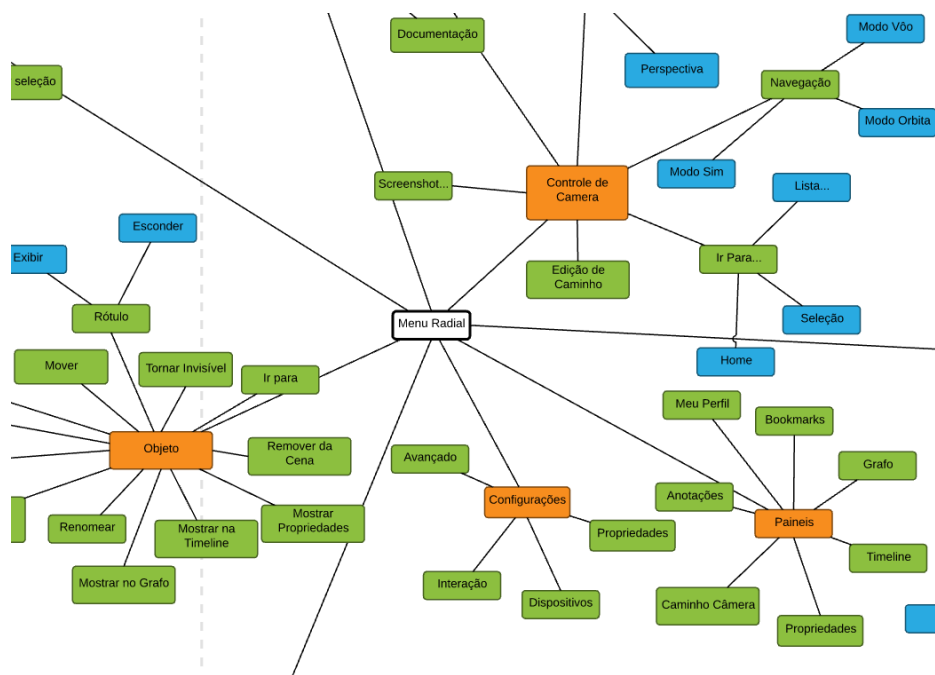
c) **Apresentação e sugestão de ideias** - Ao longo da pesquisa, foi mais produtiva a **discussão de funcionalidades em torno de referências visuais concretas**, especialmente àqueles **inspirados em jogos eletrônicos**. A equipe parece ter assimilado essa ideia – como no caso da recomendação feita por Rodrigo Pinheiro ao sistema do jogo *The Elder Scrolls V: Skyrim*.

Além disso, foi interessante perceber que **ideias descartadas no passado voltam à discussão**, atendendo muito bem a necessidades atuais. Neste caso, recomenda-se que mockups e protótipos descartados sejam arquivados para referência futura.

#### 4- Quanto ao processo de prototipagem

- a) **Interatividade** - No contexto desta pesquisa, a **apresentação de ideias através de protótipos digitais interativos foi mais produtiva** por eliminar dúvidas e ambiguidades presentes na descrição oral, esquemática, ou puramente visual das propostas. O modelo interativo permite uma **compreensão/aquisição das habilidades inerentes ao sistema** -- estes *skill atoms*, como definidos Daniel Cook, foram identificados por Sebastian Deterding como elementos de game design de grande valor, por vezes negligenciados em processos de gamificação.
- b) **Acabamento** - Para além de aspectos de interação a habilidade, recomenda-se que **o acabamento do protótipo seja o mais próximo possível ao da solução final** pretendida. Esta impressão condiz com a ênfase dada por Steve Swink na importância do acabamento (*polish*) na percepção do jogo pelo jogador, e pela busca de uma imersão de qualidade – mesmo em fases de prototipagem.
- c) **Contextualização** - Finalmente, acredita-se que **mudanças incrementais, que não afastem muito o protótipo da versão em utilização, foram melhor recebidas**. No caso do protótipo gerado para o Siviep, optou-se pela inclusão da interface original em torno do canvas para uma maior contextualização e familiaridade do usuário com a experiência apresentada. Mudanças incrementais são desejáveis em software científicos por se adequarem à flexibilidade exigida (Letondal e Zdun).

Embora um protótipo interativo, bem acabado, e contextualizado, de um conjunto de funcionalidades possa sugerir o funcionamento do sistema como um todo, achamos recomendável **esquematizar os sistemas em sua totalidade, para referência futura ou geração de novos protótipos**. No caso do Siviep, isso pode ser ilustrado no remapeamento das funções preexistentes sobre um sistema de menus radiais.



Detalhe da esquematização do sistema de menu radial

## PRÓXIMOS PASSOS

- Apresentação e avaliação do protótipo interativo.
- Teste de usabilidade comparativo.
- Elaboração de artigo, a partir da expansão dos temas apresentados nas atas de reunião, para submissão em periódicos e/ou congresso contemplando todos os envolvidos como co-autores. Sugestão de congresso: NordiCHI 2016 (aberto para papers até 14/04).

### **7.3**

#### **Conjunto de cartas elaborado para sessões de co-design**

As próximas páginas o conjunto de cartas utilizado nas sessões de co-design, conforme relatado no Capítulo 4.





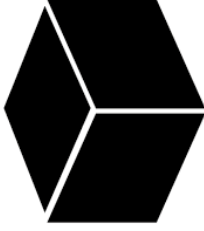
Scrutiny;  
augmentation;  
focus.



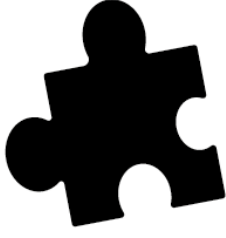
Imagination;  
thought;  
reasoning.



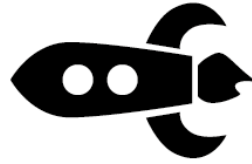
Search; inspect



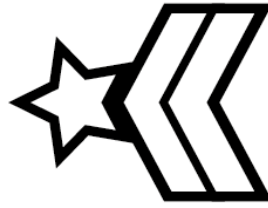
Black box;  
self-contained;  
opacity.



Puzzle;  
puzzle-solving.



Speed; travel.



Rank; achievement;  
hierarchy.



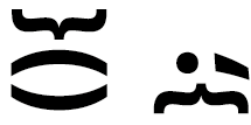
Vision; visuals.



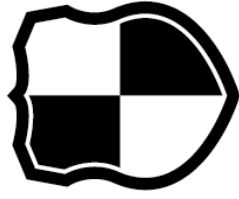
Strength; brute  
force.



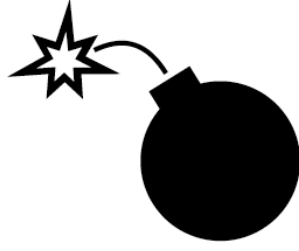
Infinity; eternity.



Programming;  
languages.



Protection; shielding.



Disruption;  
destruction.



Conflict; contest.



Magic; spell.



Customisation;  
maintenance; tools.



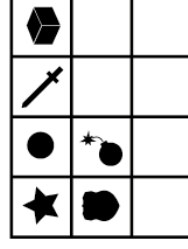
Versatility;  
improvisation;  
tools.



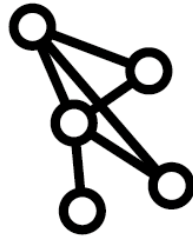
Playback.; control.



Power-up; boost



Inventory; collection;  
taxonomy.



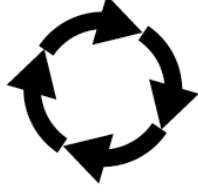
Network; graph.



Global; community.



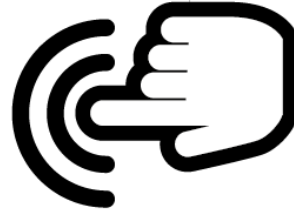
Flow; immersion.



Cycle; stages;  
repetition;  
automation.



Work environment;  
lab.



Interaction;  
manipulation.



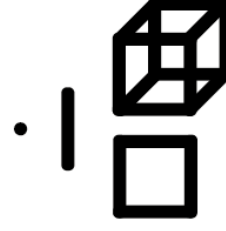
Software; coding.



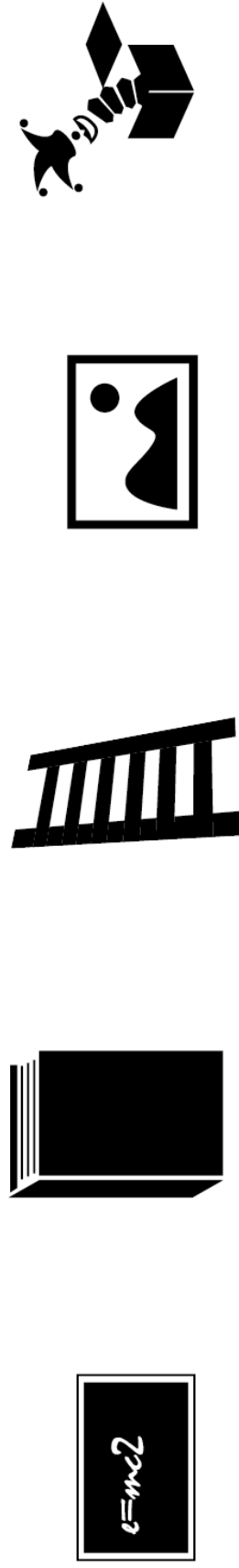
Measurement;  
indicators.



Conversation; dialog.



Dimensions;  
simplicity.



Formula; theories;  
equations.

Knowledge-base;  
library.

Support; shortcut;  
extension.

Image; photograph.

Surprise.




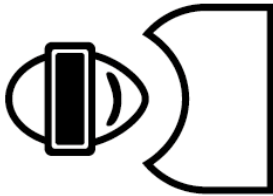

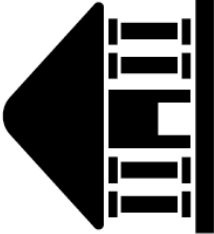

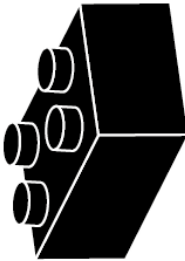
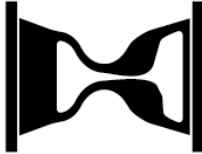



Obstacle;  
impediment; setback.

Warning.

Document; file.

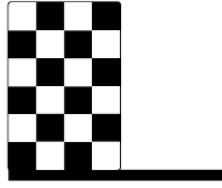
Listening; sound.

Data analysis;  
data visualization.

	Quotation; reference; statements.		Virtuality; technologies; avatar.		Recovery; reversal.		Institutions.		Investment; resources.
	Model-building; construction.		Time constraints; schedule; deadlines.		Probability; chance; randomness.		Progress; steps.		Lists; rankings.



Mastery; agility.



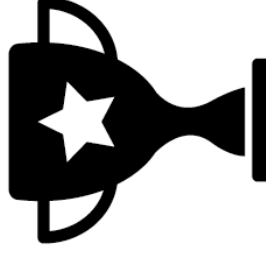
Finish line; goal.



Measurement;  
compliance.



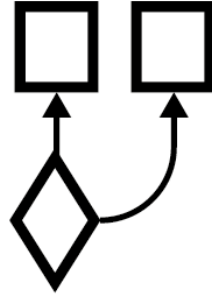
Judgement;  
validation.



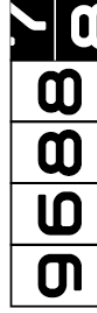
Prize; reward.



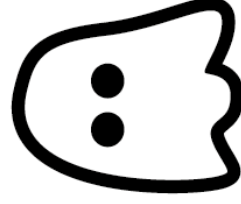
Question; doubt;  
help.



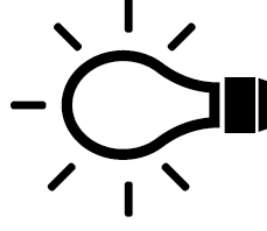
Diagram; workflow;  
logic.



Counter; score.



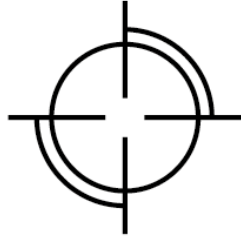
Mystery; anomaly.



Idea; enlightenment.



Milestone; flag.



Precision; accuracy.



Writing; annotation;  
drawing; editing.



Energy; resources.



Guidance;  
instructions



Portability;  
exchange;  
conversion.



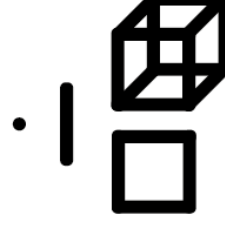
Collectivity;  
collaboration;  
users.



Tasks; checklist.



Conversation; dialog.



Dimensions;  
simplicity.

## SYSTEM REPRESENTATION

How is the scientific subject  
represented?  
How is it experienced?

## SYSTEM RULES

How does the system work?  
How should it be used?

## USER EFFORT

What do users do?  
What are their challenges and actions?

## USER SATISFACTION

What outcomes would make users  
satisfied?  
What could make their experience more  
measurable or fun?

## OUTCOMES VALORIZATION

What makes an outcome valued?  
How is it valued?

## OUTCOMES QUANTIFICATION

How is the outcome measured?  
How is it materialized?

*Simulation*

*Result Analysis*

*Modeling*