

Juliana Pereira Salles

jusalles@dcc.ufmg.br

O Modelo Fractal de Comunicação: Criando um Espaço de Análise para Inspeção do Processo de Design de Software

Tese apresentada ao Departamento de Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais, como requisito parcial para a obtenção do grau de Doutor em Ciência da Computação.

Belo Horizonte

Maio de 2001

Ao meu pai, Júlio César. Foi
convivendo com ele que
cresceu em mim o desejo de
trabalhar por um mundo
melhor.

À minha mãe, Laura, à minha
irmã, Ayesha e ao Guilherme.
Nossas interações são sempre
estimulantes e cheias de
alegria.

Agradecimentos

Ao longo dos anos em que desenvolvi este trabalho, tive o privilégio de conviver com pessoas interessantes e que me ensinaram muitas coisas que, direta ou indiretamente, contribuíram para a sua conclusão.

No meu pai, Júlio César, eu reconheci características que, para mim, são as mais essenciais em um pesquisador: uma sinceridade enorme na busca de compreender as coisas e uma grande generosidade que se manifestava através de uma preocupação de que o trabalho realizado beneficie as pessoas. Espero poder, ao longo da minha vida, praticar e entender os seus inúmeros ensinamentos. Reconheço neste trabalho e na forma como ele foi conduzido, coisas que ele antecipou para mim muito antes que eu pudesse compreendê-las.

Minha família tem sempre uma participação enorme em tudo que faço. Apesar de compartilharmos a mesma dor, minha mãe e minha irmã me mostraram como encontrar força e coragem para caminhar. Em todos os momentos, elas me supriram com o que eu precisava: carinho, ânimo, determinação e foram grandes colaboradoras deste trabalho. O Guilherme foi o companheiro de todas as horas, que dividiu pacientemente comigo as dores, alegrias, reflexões e tudo que eu experimentei ao longo desses anos.

Agradeço imensamente ao Professor Roberto da Silva Bigonha e à Professora Maria Cecília Calani Baranauskas pela oportunidade e privilégio de tê-los tido como orientadores. Foi um prazer aprender e trabalhar com pessoas tão generosas e cheias de entusiasmo.

Também tenho muito a agradecer ao Professor Antônio Alfredo Loureiro e ao Professor Jonathan Grudin. Espero que eu consiga transmitir para as pessoas que convivam comigo um pouco do que eu aprendi com eles.

Sou grata à Professora Clarisse S. de Souza, cujas sugestões contribuíram muito para o crescimento deste trabalho; ao Professor David Ribeiro Lamas e Henrique Pacca Loureiro Luna, sempre dispostos a contribuir.

Agradeço aos funcionários do Departamento de Ciência da Computação, em especial Helvécio Lopes Inácio, Belkiz Inez Costa, Gilberto Luiz Costa e Emília Soares da Silva, cujo trabalho e boa vontade tornaram o meu dia-a-dia como aluna mais produtivo.

Os períodos em que estive na Unicamp foram sempre estimulantes. Encontrei pessoas cooperativas, alegres e sempre dispostas a ajudar, entre elas o Daniel da Silva Andrade e o Alysson Prado.

Agradeço ao CNPq, que financiou parcialmente este trabalho, e à empresa em que realizei o estudo de caso, aqui nomeada ORG, que me deu a oportunidade de validar o trabalho e proveu recursos para tanto.

Resumo

Enquanto várias metodologias de design de software focalizam questões de usabilidade, aspectos relacionados à qualidade dos processos de design e desenvolvimento não têm recebido a mesma atenção. Este trabalho objetivou investigar e propor mecanismos para melhorar a usabilidade de um produto através da análise da qualidade do processo de produção de software avaliando, especificamente, a comunicação entre grupos funcionais em uma organização de produção de software.

O foco deste trabalho é, portanto, estudar os atos de comunicação envolvendo tais grupos, a natureza das mensagens trocadas e sua relevância e significado para a audiência, o impacto que elas causam e como elas são propagadas pelo processo resultando, em última instância, em melhorias para a usabilidade do produto.

Para entender a comunicação no processo de design, nós primeiramente buscamos um entendimento coerente sobre o que a própria comunicação envolve. Propusemos então um meta-modelo, o modelo fractal de comunicação, que representa a comunicação através de uma perspectiva que enfatiza: cada mensagem e seu design, os canais usados e sua influência no sucesso da propagação da mensagens e no contexto da conversação, o papel colaborativo dos agentes envolvidos na comunicação (aqueles que recebem e enviam as mensagens). Esse meta-modelo pode ser aplicado ao processo de design de um produto específico para mapear a comunicação entre seus agentes ou grupos de trabalho.

Foi proposto um método de inspeção, o percurso comunicativo, em que especialistas percorrem o meta-modelo gerado analisando a eficácia de cada ato de comunicação no processo de design.

Um estudo de caso do processo de design de uma grande organização de desenvolvimento de software e de alguns processos de produtos específicos desta organização exemplifica nossa proposta.

Abstract

While usability issues have been widely stressed in literature as a goal of the software design methodologies, quality in the design and development process has not received the same attention. Our goal is to provide means of improving the usability of the product by focusing on the quality of the software production process through the analysis of the communication among work groups in a software design organization.

The focus of this work is the investigation of the communication acts involving those groups, the nature of the changed messages and their meaning for the audience, how they impact and are propagated in the whole design process and, ultimately, promote improvements of the usability of the product.

To understand communication in the design process, we first focused on forming a coherent understanding of what communication itself involves. We then proposed a meta-model - the fractal communication model, that represents communication through a perspective that emphasizes the importance of each message and its design, the channels used and its influence on the success of the propagation of the message and on the context of the conversation, the collaborative role of those involved in the communication (those who send and receive the messages). This meta-model can be applied to a particular product design process in order to map the communication among its agents or work groups

We proposed an inspection method, the communication walkthrough in which experts walk through the generated map (meta-model) analyzing the effectiveness of each communication act concerning the design of the product.

A case study with the design process of a large software organisation and the variations for some of its specific products illustrates our proposal.

Sumário

Capítulo 1 - Introdução.....	1
1.1 Motivação.....	3
1.2 Áreas envolvidas na elaboração deste trabalho.....	5
1.3 A proposta do trabalho.....	6
Capítulo 2 - O contexto do trabalho e sua problemática.....	8
2.1 Os processos de design de interface.....	9
2.1.1 O conceito corrente de usabilidade.....	9
2.1.2 Rubinstein e Hersh.....	14
2.1.3 Gould.....	14
2.1.4 Shneiderman.....	16
2.1.5 Nielsen.....	18
2.1.6 Avaliação sobre processos de design centrados no conceito de usabilidade.....	22
2.2 A herança da Engenharia de Software.....	25
2.2.1 Os contextos de desenvolvimento de software.....	26
2.2.2 A influência da Engenharia de Software.....	28
2.3 Considerações finais.....	30
Capítulo 3 - Objetivos e Metodologia.....	36
Capítulo 4 - Modelo fractal de comunicação aplicado ao processo de design de interfaces.....	39
4.1 Introdução.....	39
4.2 Uma breve revisão sobre alguns modelos de comunicação.....	41
4.3 O modelo fractal de comunicação.....	46
4.4 Projetando a conversação do processo de design de interfaces no modelo fractal.....	50
4.5 Considerações finais.....	54
Capítulo 5 - Percurso comunicativo: um método de inspeção da comunicação no processo de design.....	58
5.1 Introdução.....	58
5.2 O percurso comunicativo.....	62
5.2.1 Visão geral.....	62
5.2.2 A formação do grupo de revisão.....	63
5.2.3 Sobre o registro.....	65
5.2.4 Quando executar um percurso.....	66
5.2.5 Descrição detalhada do percurso comunicativo.....	66
5.2.6 Escopo e limitações do método.....	70
5.3 Considerações finais.....	72
Capítulo 6 - Aplicação do percurso comunicativo: estudo de casos.....	73
6.1 O percurso comunicativo aplicado ao modelo do ciclo de vida do produto na ORG.....	74
6.1.1 Descrição do modelo do ciclo do produto.....	74
6.1.2 Aplicação do percurso comunicativo ao modelo.....	77
6.1.3 Considerações finais.....	90
6.2 Como as especificidades do produto são evidenciadas pelo meta-modelo.....	92
6.2.1 Jogos.....	93
6.2.2 Ferramentas para desenvolvedores.....	102
6.2.3 Considerações finais.....	108
6.3 Meta-modelo no nível 2 – o design de mensagens.....	110
6.3.1 Cenário de aplicação do método.....	111
6.3.2 Aplicação do percurso comunicativo.....	113
6.3.3 Considerações finais.....	120
6.4 Conclusões e discussões finais.....	121
Capítulo 7 - Discussão e conclusões.....	126
7.1 Contribuições.....	127

7.2 Perspectivas futuras.....	130
Bibliografía.....	134

Lista de Figuras

Figura 2.1: Ciclo de vida estrela para desenvolvimento da interação com os usuários. Extraído de Hix e Hartson, 1993, p. 102.....	33
Figura 4.1: Modelo de comunicação de Shannon & Weaver.....	42
Figura 4.2: Modelo de comunicação de Westley e MacLehan.....	43
Figura 4.3: Modelo de comunicação proposto por Jakobson.....	44
Figura 4.4: Locutor e ouvinte se alternam em turnos durante a comunicação. Linhas pontilhadas, ato perlocucionário; linhas sólidas, ato ilocucionário. Extraído de (Liu, 2000), p. 84.....	46
Figura 4.5: O modelo fractal de comunicação.....	47
Figura 4.6: Um exemplo de processo de design iterativo com sucessivas fases do ciclo design-implementação-avaliação (extraído da Introdução do Capítulo 2 de [Baecker et al., 1995], p. 74).....	51
Figura 4.7: Conversação entre a equipe de design (ED) e usuários (U) através de diferentes canais.....	52
Figura 4.8: Comunicação dos designers com a interface emergente.....	53
Figura 4.9: Interação usuários-sistema.....	53
Figura 4.10: A comunicação no processo de design/desenvolvimento. (a) Nível 1 - comunicação entre a equipe de criação do produto e usuários; (b) nível 2 – equipe de design e usuários; (c) nível 2 – equipe de design e toda a equipe de criação do produto.....	55
Figura 5.1: Processo de design baseado na aplicação de métodos de avaliação usabilidade que focalizam o produto; dados coletados realimentam a iteração seguinte do processo.....	59
Figura 5.2: O método proposto avalia os atos de comunicação que ocorrem durante o processo de design e desenvolvimento.....	61
Figura 5.3: Primeiro passo - identificação de todos os agentes do processo.....	67
Figura 5.4: Segundo passo – as conversações entre os agentes são explicitadas (canais e mensagens identificados).....	67
Figura 6.1 (a): Meta-modelo de comunicação aplicado ao processo de desenvolvimento da organização - comunicação entre os agentes do processo.....	79
Figura 6.1: (b) Os diferentes agentes em conversação com o produto emergente; (c) conversação entre engenheiros de usabilidade e a interface; (d) conversação entre desenvolvedores e o código.....	80
Figura 6.2: Meta-modelo de comunicação aplicado ao processo de criação de jogos.....	96
Figura 6.3: Teste de usabilidade como canal entre designers e engenheiros de usabilidade em: (a) modelo do ciclo de vida; (b) processo de criação de jogos.....	100
Figura 6.4: Meta-modelo de comunicação aplicado ao processo de criação de jogos.....	104
Figura 6.5: O meta-modelo de comunicação aplicado ao processo de design do teste de <i>help</i> das aplicações citadas.....	116

Lista de tabelas

Tabela 5.1: Visão geral do percurso comunicativo	63
Tabela 6.1: Agentes (correspondem às funções organizacionais) do processo e as conversações que estabelecem.	78
Tabela 6.2: Agentes do processo e as conversações que estabelecem.	95
Tabela 6.3: Agentes do processo e as conversações que estabelecem.	103
Tabela 6.4: Agentes envolvidos no design do teste do <i>help</i> e as conversações que estabelecem.	114
Tabela 6.4: Continuação.	115

Capítulo 1

Introdução

Apesar do imenso sucesso dos computadores pessoais nos dias atuais, a experiência de utiliza-los é ainda complexa e cheia de barreiras para muitas pessoas. Usuários aprendem o mínimo necessário para realizar seu trabalho e subutilizam produtos que exigem um esforço de produção enorme. Mitchell Kapor aponta para a importância de que “os profissionais de computação assumam a responsabilidade de criar uma experiência positiva para os usuários” (Kapor, 1996, p. 3). Ele enfatiza que, para isso, é necessário haver uma mudança conceitual no processo de criação de artefatos computacionais, através do reconhecimento do papel crítico do design como uma atividade de mesma relevância e impacto que a programação. Em um processo eficaz, essas atividades devem ser pares.

É importante notar, contudo, que a educação e o treinamento dos profissionais de computação não têm sido direcionados para questões de design, como aponta Winograd:

“The education of computer professionals has often concentrated on the understanding of computational mechanisms, and on engineering methods that seek to ensure that the mechanisms behave as the programmer intends. The focus is on the objects being designed: the hardware and software. The primary concern is to implement a specified functionality efficiently. When software engineers or programmers say that a piece of software *works*, they typically mean that it is robust, is reliable, and meets its functional specification.” (Winograd, 1996, p. xv)

Os métodos e técnicas já estabelecidas para o desenvolvimento de software não se aplicam ao design da interface, como discutem Hix e Hartson (1993). Os autores apontam que “tais técnicas são eficazes para a implementação do software que suporta a interface, mas não para o design do que tal software deve suportar – a interação com os usuários” (p. 104). O elemento humano não foi considerado pela Engenharia de Software convencional, o que torna o desenvolvimento da “interação com os usuários” um domínio com seus próprios problemas e desafios, que exige habilidades específicas e suas próprias técnicas para desenvolvimento.

A presença desse elemento humano é central para a disciplina de design. Quando um arquiteto planeja uma casa, ele determina uma série de padrões relacionados à maneira como seus habitantes irão utilizá-la. Por essa razão, o design focaliza não a estrutura do edifício em si, mas o uso que os habitantes farão do edifício. Da mesma forma, os artefatos computacionais que criamos são moldados por nós, mas, à medida que são incorporados ao nosso cotidiano, eles alteram o contexto que nos envolve e acabam moldando a nossa realidade¹. Assim como o design de uma casa, o design de software precisa ser centrado nos interesses e prioridades dos usuários.

Essa perspectiva vem crescendo progressivamente e vários pesquisadores têm discutido seu impacto na criação de artefatos computacionais.

Pelle Ehn (1997) aponta que o design de software deve ser entendido como uma atividade criativa cujo enfoque é modelar possíveis realidades futuras.

Winograd (1996) discute o conceito de design e analisa várias dimensões envolvidas. Ele aponta que a atividade de “design é intrinsecamente complexa – cada escolha dos designers tem efeitos intencionados e não intencionados” (p. xxi).

O autor coloca que o processo de design envolve uma conversação em que o artefato emergente gera interrupções e contribuições. Por outro lado, os designers veiculam mensagens através dos artefatos, comunicando aos usuários como utilizar um dispositivo. Assim como causa impacto social, o design emerge de um contexto social, em que os recursos são limitados e vários agentes, inclusive o designer, disputam valores associados às funções que desempenham em uma organização.

Neste trabalho, o processo de criação e desenvolvimento de um software é denominado processo de design do software: a definição da funcionalidade, o design da interface e o próprio projeto de software não são atividades disjuntas; elas se afetam mutuamente e se complementam para viabilizar a criação de um produto que causa impacto na vida dos usuários, individual e socialmente. Na nossa perspectiva, o design de software envolve todas as dimensões de construção de um artefato que estende e enriquece a experiência das pessoas que utilizam aquele software.

¹ Marshall McLuhan aponta que “nós moldamos nossas ferramentas e elas nos moldam”, provocando efeitos tanto para o indivíduo quanto para a sociedade.

1.1 Motivação

Dois aspectos importantes estão relacionados ao processo de design de interfaces²: (1) a obtenção de informações sobre os usuários e o contexto em que utilizam uma aplicação; (2) a utilização adequada dessas informações para suportar as decisões de design (i.e., design de uma interface que responda, de maneira apropriada, às necessidades e demandas dos usuários).

O primeiro aspecto diz respeito à conversação entre a equipe de design e os usuários; o segundo envolve a atuação da organização como um todo, as interações entre os vários grupos que trabalham em conjunto para a proposição e efetivação de um design.

Considerando a conversação entre a equipe de design e os usuários, um dos focos prioritários das metodologias de usabilidade correntes é o desempenho dos usuários ao interagir isoladamente com sistemas computacionais. Design e avaliação de interfaces de usuário são centrados em atributos relacionados ao tempo e esforço que as pessoas empregam para executar tarefas.

Tipicamente, os processos de design para a usabilidade dos sistemas incluem os seguintes passos:

- a definição de objetivos formais de usabilidade para o sistema em desenvolvimento. Nesta etapa, designers decidem os valores esperados para atributos relacionados ao desempenho dos usuários;
- sucessivas etapas do ciclo design-implementação-avaliação-modificação. O design é avaliado e, de acordo com os resultados alcançados, modificado para aproximar o desempenho que será efetivamente alcançado pelos usuários do desempenho pretendido (considerando os objetivos de usabilidade definidos).

Nesse enfoque, os testes de usabilidade têm sido usados como um dos principais canais de comunicação entre designers e usuários. Entretanto, a partir de desenvolvimento da área de Interação Humano-Computador (IHC), os canais de comunicação entre designers e usuários têm crescido em número. Abordagens como design participativo (Muller, 1997) e contextual (Beyer e Holtzblatt, 1998) podem

² Neste trabalho, o termo “design” refere-se sempre à atividade de criação de interfaces para usuários de sistemas computacionais. A comunidade de Interação Humano-Computador no Brasil vem adotando este termo em oposição a projeto, pois esse geralmente se refere ao desenvolvimento do software (i.e., código).

ser entendidas como canais adicionais de comunicação que aproximaram os usuários e a equipe de design. Além disso, outros canais de comunicação com usuários podem ser identificados na organização: a equipe de *marketing* determina as características e interesses de um dado mercado, consultores atuam como mediadores entre a organização dos usuários e a organização dos desenvolvedores, equipes de suporte orientam clientes, etc. (Grudin, 1991).

As abordagens semióticas (Nadin, 1988; de Souza, 1993; Andersen, 1997) explicitam a interface como uma mensagem enviada pelos designers aos usuários utilizando o computador como canal. O design dessa mensagem, contudo, reflete a capacidade da organização como um todo entender as necessidades e interesses dos usuários e prover uma resposta adequada.

Se, por um lado, a organização tem um conjunto claro e bem definido de objetivos, os vários indivíduos e grupos que a integram podem não compartilhar tais objetivos e interesses da mesma forma. Eles têm especialidades, formações, perspectivas diferentes sobre a criação de um artefato. Argumentamos que o processo de design será efetivo se as interações entre eles possibilitarem que suas diferentes perspectivas se complementem, de tal forma que seja possível criar uma resposta unificada da organização para os usuários. Ou seja, o produto deve consolidar as colaborações e perspectivas dos diversos grupos que cooperam para a sua criação.

Considerar a dimensão da comunicação na atividade de design envolve enxergar o processo de design/desenvolvimento por uma outra perspectiva.

Existe uma diversidade de técnicas e métodos de design e avaliação de usabilidade de interfaces de usuário. Entretanto, selecionar quais os métodos mais apropriados para um dado contexto, determinar quanto iterar design e avaliação, são decisões complexas e muito dependentes das particularidades de cada projeto (Gould, 1995; Grudin, 1993). Ao focalizar a comunicação com os usuários, a equipe de design precisa identificar quais os canais (técnicas e métodos) de que dispõe, quais pode criar (e cuja criação, em termos de custo/benefício, é viável) e como eles podem influenciar a interpretação da mensagem final (i.e., a interface).

Por outro lado, os métodos de usabilidade correntes focalizam como “obter” informações sobre os usuários, mas comunicá-las de forma efetiva à equipe de desenvolvimento exige que os designers identifiquem as formas e canais adequados e entendam as várias limitações do processo e os objetivos da organização. Isso demanda uma grande integração entre os processos de design e desenvolvimento do produto. Como resultado da atuação de diferentes áreas, principalmente IHC e ES,

com objetivos, interesses e vocabulários próprios, observa-se atualmente a existência de uma lacuna entre os processos de design e desenvolvimento do produto, lacuna essa que é foco de investigação deste trabalho. É nosso objetivo identificar deficiências de comunicação nesses processos e oportunidades para solucioná-las, aproximando assim as atividades de design e desenvolvimento.

1.2 Áreas envolvidas na elaboração deste trabalho

IHC é, tradicionalmente, uma área multidisciplinar. Sistemas computacionais e suas interfaces são desenvolvidos e aprimorados através do trabalho coordenado de especialistas de várias disciplinas (ex., ciência da computação, psicologia cognitiva, design gráfico). As metodologias, métodos, técnicas e recomendações correntes são resultantes da cooperação entre várias áreas.

Contudo, o conhecimento produzido em IHC só causará impacto na qualidade do produto se os processos de desenvolvimento de software incorporarem a atividade de design de interfaces como um elemento essencial. O enfoque deste trabalho envolve entender como se articulam os processos de design e de desenvolvimento na criação de um artefato de software. Para tanto, adotou-se a perspectiva da comunicação.

Alguns tópicos foram particularmente importantes para a elaboração deste trabalho: metodologias de design fundamentadas no conceito corrente de usabilidade (no domínio de IHC), processos de desenvolvimento da Engenharia de Software e o conceito de comunicação aplicado ao contexto da criação de software.

O conceito corrente de usabilidade é centrado no desempenho individual dos usuários interagindo com sistemas monousuário. Foi com base neste conceito que grande parte das técnicas e métodos correntemente usados (ex., diretrizes, procedimentos para testes de usabilidade em laboratórios, etc.) foram derivados. Eles focalizam características gerais do diálogo compartilhadas pela maioria dos usuários. Grande parte dos esforços de pesquisa em IHC foi direcionado para aspectos como controle motor, percepção e processos cognitivos de baixo nível – o *look and feel* do software.

Essas técnicas e métodos estão organizados em metodologias que estruturam o processo de design, se seguidos em seqüência (Baecker *et. al.*, 1995). Metodologias ou processos de design são um outro grande tópico de IHC. Resultados importantes foram derivados de pesquisas nesta linha. Ao contrário das metodologias para desenvolvimento de sistemas não interativos, os processos de design estabeleceram, desde cedo, o conceito de design iterativo, em que os sistemas são desenvolvidos e

aprimorados através de sucessivas iterações dos estágios de design, implementação e avaliação (Gould e Lewis, 1985; Gould, 1995). O design iterativo é essencialmente empírico, centrado em testes com usuários.

Essa iteratividade do design deve ser suportada por uma metodologia de desenvolvimento que permita que os resultados das avaliações resultem em modificações rápidas na interface. Por outro lado, os processos de desenvolvimento da Engenharia de Software são centrados nos desenvolvedores e na produção de software e relegam o design da interface a uma fase ou subfase do processo. Este enfoque pode ser adequado para o desenvolvimento de sistemas não interativos, mas apresenta várias limitações e conseqüências (especialmente em termos de cronograma e orçamento) quando se trata do desenvolvimento de sistemas interativos. Uma integração entre essas duas áreas de conhecimento (processos de design e processos de desenvolvimento de software) é necessária.

A partir de considerações das abordagens semióticas, os atos de comunicação no processo de design foram investigados. Para tanto, foi necessário formar um entendimento coerente sobre o próprio conceito de comunicação e o que ele envolve. Alguns modelos de comunicação foram estudados a fim de se analisar sua adequação para representar a comunicação no processo de design de software. Através desse entendimento, da identificação de elementos importantes no processo comunicativo, propusemos um novo modelo de comunicação, que permite uma representação mais apropriada dos diversos níveis de comunicação no processo de design e da participação dos vários agentes envolvidos. Com base nesse modelo de comunicação, foi proposto um método de inspeção para o processo de design de software.

1.3 A proposta do trabalho

Na perspectiva deste trabalho, para que o processo de design seja efetivo, é necessário criar um entendimento claro e integrado sobre as conexões entre as atividades de design e desenvolvimento, em todos os níveis. Isso exige que se compreenda como cada grupo na organização contribui para a criação do produto e comunica sua perspectiva para os demais, já que o produto final resulta das negociações entre todos eles.

O enfoque deste trabalho é a comunicação envolvida no processo criação de artefatos computacionais. Uma das dimensões dessa comunicação é o diálogo que ocorre entre os designers e os usuários que está implícito na própria estrutura do

equipamento [Adler e Winograd, 1992]. Esse tipo de comunicação é intrínseco a todos os tipos de artefato. Através da sua estrutura e aparência, os objetos expressam mais ou menos efetivamente o que eles são, como eles devem ser usados e como eles se integram ao contexto. Os usuários interpretam os símbolos que encontram no artefato e no contexto associado para entender o estado de um sistema, as ações que eles podem executar e os resultados dessas ações. As abordagens semióticas em IHC percebem essa estrutura e aparência – a interface – como uma mensagem enviada pelos designers aos usuários utilizando o computador como canal.

O processo de design dessa mensagem exige que todos os grupos da organização desenvolvam um entendimento comum sobre a resposta que a organização irá prover para os usuários. Isso envolve um processo de negociação, não apenas entre a equipe de design e os usuários, mas entre todos os agentes que integram a equipe de criação do produto.

Neste trabalho, é nosso objetivo entender as comunicações que se estabelecem entre esses agentes, a natureza das mensagens que eles trocam, seu significado e relevância para a audiência, como elas são propagadas pelo processo e o impacto que causam. Ou seja, é nossa intenção avaliar a eficácia da comunicação entre os vários agentes envolvidos no processo.

Para tanto, desenvolvemos um modelo de comunicação que permite que os diversos níveis de comunicação inerentes ao processo de design sejam representados (i.e., a comunicação entre a equipe de design e os usuários, entre a equipe de design e toda a equipe de criação do produto). Esse modelo, quando aplicado ao processo de criação de um produto, cria um espaço de análise, um “mapa” das comunicações entre os agentes do processo.

O modelo de comunicação proposto serviu de fundamento para a proposição de um método de inspeção que auxilia na identificação de deficiências de comunicação no processo que causam impacto na usabilidade do produto. O método avalia como a conversação entre a equipe de design e os usuários é refletida no design do produto.

Este trabalho está organizado da seguinte forma: o capítulo 2 apresenta uma revisão dos processos de design de interfaces e de desenvolvimento de software e identifica uma lacuna entre eles. O capítulo 3 discute os objetivos e metodologia envolvidos no desenvolvimento deste trabalho. O capítulo 4 apresenta uma breve revisão de alguns modelos clássicos de comunicação e o modelo proposto. O capítulo 5 apresenta o método de inspeção proposto, o percurso comunicativo. O capítulo 6 apresenta um estudo de casos que ilustra a aplicação do método. Finalmente, o capítulo 7 apresenta as contribuições, conclusões e discussões finais.

Capítulo 2

O contexto do trabalho e sua problemática

A maior parte dos trabalhos em IHC focaliza o diálogo entre os usuários e o computador (Backer *et al.*, 1995). Outras abordagens ampliaram esse enfoque, emergindo da observação de que os indivíduos trabalham em um contexto social que influencia o design de uma aplicação. Cabe ressaltar que nem todos os métodos e técnicas para design de interfaces e avaliação de usabilidade são apropriados para todos os projetos (Gould, 1995) e selecioná-los de maneira adequada depende da habilidade e experiência da equipe de design. Frequentemente, tais métodos e técnicas se concentram na conversação entre designers/engenheiros de usabilidade e usuários.

Além de serem usados em um contexto social, os sistemas interativos são planejados e construídos por grupos de pessoas e o design da interface é um dos elementos envolvidos. A maior parte das abordagens da Engenharia de Software não incorpora ou prioriza aspectos necessários para o design e desenvolvimento da interface. Por outro lado, as limitações de cada projeto (ex., técnicas, de cronograma e orçamento e até da própria equipe) impõem restrições às possíveis soluções de design.

Assim, é propício analisar os processos de design e desenvolvimento em conjunto. Os reflexos de um processo se manifestam no outro e têm impacto direto na qualidade do que é efetivamente disponibilizado para os usuários.

Na próxima seção, alguns processos clássicos de design são analisados. A seguir, uma breve revisão dos contextos de desenvolvimento e da herança da Engenharia de Software é apresentada. Nas considerações finais, as lacunas entre os processos de design e desenvolvimento de software são discutidas.

2.1 Os processos de design de interface

Na Engenharia de Software tradicional, a prática é primeiro se obter um software correto e depois avaliar sua conformidade com os requisitos e especificações funcionais. Design, implementação e avaliação são tratadas como tópicos separados. Entretanto, esse tipo de procedimento não é apropriado para o design de sistemas interativos. É difícil definir antecipadamente tanto os requisitos e especificações corretas, quanto as melhores estratégias de interação. O entendimento sobre esses aspectos surge através da observação dos usuários e do contexto que os envolve. O próprio uso de um artefato traz consequências tanto para a comunidade quanto para o indivíduo (Press, 1995) que precisam ser consideradas e entendidas.

É através de uma conversação entre designers/engenheiros de usabilidade e usuários que se alcança um entendimento progressivo sobre o que deve ser desenvolvido. Soluções são propostas pela equipe de design e sua adequação ao contexto de uso é avaliada observando-se as reações dos usuários. Esses dados, por sua vez, realimentam o processo viabilizando o refinamento da interface.

Esse processo de design, implementação, avaliação, realimentação e modificação precisa ser repetido iterativamente até que se alcance um resultado satisfatório. A avaliação se estende sobre todas as etapas deste processo: avaliação das versões correntes e de sistemas concorrentes, avaliação das práticas de trabalho, avaliação de protótipos, de versões alfa e beta, avaliação do produto final (Baecker *et al.*, 1995). Assim, o design de interface é uma atividade essencialmente iterativa e empírica.

Em IHC, o reconhecimento da necessidade de se adotar uma abordagem empírica aconteceu cedo. O ciclo típico com a realização de sucessivas etapas de design-implementação-avaliação é comum aos processos de design apresentados na literatura. Nas próximas subseções, alguns destes processos, centrados no conceito de usabilidade, são apresentados. Eles estruturaram a aplicação de técnicas e métodos de usabilidade ao longo do processo de design e objetivam, especialmente, melhorar o desempenho de usuários interagindo individualmente com um sistema. O conceito de usabilidade praticado correntemente e no qual se fundamentam tais processos será apresentado a seguir.

2.1.1 O conceito corrente de usabilidade

As metodologias correntes para design e avaliação de interfaces baseiam-se, principalmente, em testes quantitativos orientados a aspectos cognitivos do uso.

Grande parte dos métodos, técnicas e princípios que orientam o design e avaliação de interfaces focalizam o usuário interagindo individualmente com um sistema particular.

Esta orientação a aspectos quantitativos possibilitou uma argumentação objetiva para que a indústria de software investisse em usabilidade, enfatizando os benefícios que tais investimentos trazem a curto e longo prazo. Benefícios a curto prazo são aqueles alcançados durante o desenvolvimento de um projeto; benefícios a longo prazo são os observados após a liberação de um produto.

A redução do custo e do tempo de desenvolvimento são benefícios prioritários alcançáveis a curto prazo. Por exemplo, o foco antecipado nos usuários e suas tarefas são um princípio comum às metodologias de design. A aplicação desse princípio pressupõe o uso de métodos e técnicas, desde as fases iniciais do projeto, que reduzem significativamente os custos para identificar e solucionar problemas de usabilidade e o tempo de desenvolvimento (ex., análise de tarefas, questionários, design participativo) (Karat, 1994).

Benefícios a longo prazo incluem, por exemplo, aumento da produtividade e satisfação dos usuários, diminuição dos custos de treinamento e suporte e a diminuição das taxas de erros dos usuários. Para os casos de desenvolvimento interno (o cliente do sistema é alguém dentro da própria organização), o aumento da produtividade e a diminuição da taxa de erro dos usuários são benefícios diretos. Nos casos de desenvolvimento de produtos para o mercado, estes atributos os tornam competitivos.

Em geral, quando se fala em melhorar a usabilidade de um sistema, a ênfase dos designers é possibilitar o aumento da produtividade dos usuários ao utilizar tal sistema. O conceito corrente de usabilidade focaliza, basicamente, o tempo e esforço empregado pelos usuários para completarem tarefas, como pode ser visto nas definições a seguir.

Shneiderman (1992, pp. 15) aponta as seguintes métricas de usabilidade como centrais para a avaliação:

- Facilidade de aprendizado
Quanto tempo é necessário para que membros típicos da comunidade de usuários aprendam como utilizar os comandos para executar um conjunto relevante de tarefas?
- Eficiência/desempenho na execução de tarefas

Quanto tempo é necessário para que os usuários executem um *benchmark* significativo de tarefas?

- Taxa de erros
Qual a quantidade e tipos de erros ocorridos ao se executar um *benchmark* de tarefas? Apesar de o tempo necessário para corrigi-los ser incorporado pela métrica “eficiência na execução de tarefas”, a taxa de erros é um componente crítico da usabilidade.
- Período de retenção
Por quanto tempo os usuários conseguem manter seu conhecimento sobre a forma de utilização de um sistema?
- Satisfação do usuário
Quão satisfeitos os usuários se sentem ao utilizar os vários aspectos do sistema?

Mayhew (1992, p. 30) coloca que o objetivo final de um sistema interativo é melhorar o desempenho dos usuários, onde **desempenho é definido como o tempo e esforço empregados na execução de suas tarefas**. Este conceito resume o anterior, já que minimizar o tempo e esforço de execução de tarefas implica aumentar a desempenho na execução, minimizar a taxa de erros, aumentar a facilidade de aprendizado e o período de retenção.

Nielsen (1993) enfatiza que a usabilidade não é uma propriedade unidimensional da interface, mas uma combinação de múltiplos componentes tradicionalmente associados aos cinco parâmetros enumerados anteriormente (facilidade de aprendizado, eficiência, erros, retenção e satisfação). A Engenharia de Usabilidade (Nielsen, 1993) provê métodos e técnicas estruturadas que focalizam esses parâmetros e que devem ser aplicados durante o ciclo de vida do produto para melhorar a usabilidade da interface.

Tipicamente, o processo de design inclui os seguintes passos:

- definição dos objetivos formais de usabilidade (i.e., a prioridade de cada um dos cinco atributos anteriores), dependendo da natureza do sistema em desenvolvimento (ex., minimizar o tempo de execução de tarefas para sistemas de tele vendas);
- várias iterações do ciclo design-implementação-avaliação-modificação. A avaliação envolve a análise da interface quanto aos cinco parâmetros

supracitados, observando-se as prioridades determinadas no passo anterior. Por exemplo, em testes de usabilidade o desempenho dos usuários na interação é avaliado. Em função dos resultados alcançados, o design é modificado a fim de tornar a interação mais eficiente (de acordo com os objetivos de usabilidade definidos). Medidas de desempenho incluem, por exemplo, o tempo para completar cada tarefa (especialmente as críticas ou mais frequentes), número de tarefas completadas corretamente, número e a natureza dos erros, tempo para recuperação de erros, número de acessos ao manual e ao *help*. Medidas de preferência, relacionadas ao quinto atributo, dependem do julgamento subjetivo dos usuários sobre a utilidade do produto, como o produto preencheu suas expectativas, a adequação da funcionalidade às tarefas executadas, facilidade de acesso à funcionalidade, etc.

As metodologias, técnicas, princípios e recomendações estabelecidas em IHC objetivam, principalmente, obter melhorias em aspectos da interface a partir da consideração dos atributos discutidos. Os estudos realizados para justificar os investimentos em usabilidade também se concentram nesses atributos.

Segundo Hix e Hartson [1993], os custos determinantes na aquisição de softwares interativos são os de *personware*. Assim, os investimentos em usabilidade são justificados especialmente avaliando-se a economia de uma empresa em termos do número de horas/homem anuais poupadas com aspectos improdutivo da interface [Bias e Mayhew, 1994]. Os estudos sobre custo/benefício baseiam-se, essencialmente, nos cinco parâmetros tradicionais enumerados anteriormente. Um estudo típico produz análises como a seguinte.

Considere um sistema com 250 usuários que, em média, preenchem 60 telas por dia. Suponha que esses usuários trabalhem 230 dias por ano e que recebam \$25 por hora de trabalho. Se, com a introdução de atividades de usabilidade no desenvolvimento de uma nova versão do sistema, for possível reduzir o tempo de preenchimento de cada tela de 1 segundo, a empresa fará uma economia de \$23.958, de acordo com o cálculo a seguir.

$$250 \text{ usuários} \times 60 \text{ telas/dia} \times 230 \text{ dias/usuário} \times 1 \text{ seg/tela} \times 25 \text{ \$/hora} = \$23.958$$

Se, além da redução do tempo de preenchimento das telas, as atividades de engenharia de usabilidade aumentarem a facilidade de aprendizado do sistema, é

possível reduzir o tempo de treinamento dos usuários. Considerando que, tipicamente, o treinamento seja de uma semana e que a redução introduzida seja de 4 horas. O benefício alcançado será de

$$250 \text{ usuários} \times 4 \text{ hora/usuário} \times 25 \text{ \$/hora} = \$25.000 \text{ no primeiro ano.}$$

Considere ainda que, com as atividades de usabilidade introduzidas, haja uma diminuição na taxa de erros. Suponha que sejam eliminados 0.2 erros por dia de trabalho de cada usuário e que o tempo médio para recuperação de erros seja de 2 minutos. O benefício resultante será de

$$250 \text{ usuários} \times 0.2 \text{ erros/dia} \times 230 \text{ dias/usuário} \times 0,833 \text{ \$/erro} = \$9.580 \text{ por ano}$$

Em geral, os aspectos mais ressaltados nos estudos mostrando os benefícios da introdução de práticas de usabilidade no processo de desenvolvimento são:

- para os casos de desenvolvimento interno:
 - aumento da produtividade dos usuários;
 - diminuição dos erros;
 - diminuição dos custos de treinamento;
 - economias com o processo de desenvolvimento, em função das necessidades de mudanças serem detectadas mais cedo no ciclo de vida;
 - diminuição dos custos de suporte aos usuários;
- para os casos de desenvolvimento de produtos para o mercado:
 - aumento das vendas;
 - diminuição dos custos de suporte ao cliente;
 - economias com o processo de desenvolvimento, em função das necessidades de mudanças serem detectadas mais cedo no ciclo de vida;
 - diminuição dos custos de treinamento (caso o treinamento seja oferecido pela empresa de vendas).

Esses atributos são importantes para todos os tipos de software, mas são absolutamente fundamentais para alguns. Para sistemas de televendas, por exemplo, os benefícios não resultam apenas do número de horas/homem anuais poupadas, mas do aumento do número de vendas, da satisfação dos clientes, etc.

Em geral, o processo de design de interfaces envolve várias iterações em que o desempenho dos usuários, segundo os atributos citados, é verificado e o design é modificado. A seguir, alguns processos clássicos da literatura serão apresentados.

2.1.2 Rubinstein e Hersh

A metodologia de design proposta por Rubinstein e Hersh (1984), apresentada em (Baecker *et al.*, 1995, p. 77), é uma sucessão de iterações de cinco etapas:

1. **fase de coleta de informações**, incluindo os objetivos do sistema, requisitos de *marketing*, requisitos técnicos, estado da arte, padrões da indústria, produtos concorrentes, capacidade da equipe de desenvolvimento. A principal atividade nesta fase é a análise das tarefas que os usuários;
2. **fase de design do sistema**, em que são produzidos a especificação do modelo conceitual através do qual os usuários percebem e entendem o sistema, o modelo de uso descrevendo como utilizar o sistema, a documentação e estratégias para detecção e recuperação de erros;
3. **fase de implementação**, primeiramente de protótipos e depois do próprio sistema;
4. **fase de teste e avaliação do sistema**, formais e informais;
5. **fase de liberação do sistema** e avaliação das reações dos usuários e do mercado.

2.1.3 Gould

Gould³ (1995) aponta que a usabilidade é determinada por um conjunto amplo de componentes que vai muito além da interface do sistema e que são mutuamente dependentes, como desempenho do sistema, funcionalidade, documentação, treinamento e suporte a grupos, entre outros. O autor propõe, ainda, que o design bem sucedido deve seguir quatro princípios de usabilidade:

³ Este trabalho é uma revisão de um artigo histórico publicado em Martin Helander (ed.), *Handbook of Human-Computer Interaction*. North-Holland: Elsevier, 1988, pp 757-789.

- **foco antecipado e contínuo nos usuários:** a equipe de design precisa entender os usuários do sistema. Esse entendimento é alcançado pelo estudo direto de suas características cognitivas, comportamentais e antropométricas e das tarefas que eles devem executar. Gould sugere a utilização de técnicas variadas para a aplicação deste princípio, como entrevistas, observações, questionários e design participativo;
- **foco antecipado e contínuo em testes com a participação de usuários:** o autor aponta que a única abordagem factível para um design bem sucedido é uma abordagem empírica. Os usuários alvo do sistema devem executar tarefas reais utilizando simulações e protótipos desde as fases iniciais do desenvolvimento e seu desempenho e reações devem ser medidos quantitativa e qualitativamente. Os resultados das avaliações devem ser analisados cuidadosamente e a equipe de desenvolvimento deve estar comprometida e motivada para realizar rapidamente as modificações no design. É importante ressaltar que os vários componentes da usabilidade devem ser testados;
- **design iterativo:** os vários componentes de usabilidade de um sistema em desenvolvimento (ex., funcionalidade, interface, *help* do sistema, documentação, abordagens de treinamento) devem ser modificados em função dos resultados dos testes com usuários. Este processo de design, implementação, teste, avaliação e modificação deve ser repetido iterativamente para melhorar o sistema;
- **design integrado:** todos os aspectos de usabilidade devem evoluir em paralelo em vez de serem definidos seqüencialmente, e devem estar sob a responsabilidade de um único gerente.

Com base nestes princípios Gould (1995) propôs uma metodologia informal dividida em quatro fases:

1. **fase de “engrenar” o projeto:** esta é, predominantemente, uma fase de coleta de informações. Nesta etapa, pontos de referência devem ser identificados (ex., sistemas concorrentes ou relacionados). Os designers devem se familiarizar com os padrões de interface, diretrizes de design e procedimentos de desenvolvimentos existentes em suas organizações;

2. **fase inicial de design:** nesta etapa, deve-se produzir uma especificação preliminar da interface baseada nos sistemas existentes, nos padrões de interface e nas diretrizes de design, caso seja apropriado. O princípio “foco antecipado nos usuários” é fundamental nesta fase. Gould recomenda que os designers interajam diretamente com os usuários alvo do sistema para coletar informações sobre eles e sobre as tarefas que executam. Outra atividade importante é o estabelecimento de objetivos de usabilidade testáveis, o que demanda: (1) a descrição dos usuários alvo do sistema e a identificação dos participantes dos testes; (2) a identificação e descrição das tarefas a serem executadas no teste e das circunstâncias em que elas devem ser executadas; (3) a identificação e definição das medidas de interesse (ex., tempo de aprendizado, taxa de erros, etc.). É difícil estabelecer os valores que se deseja alcançar para cada uma dessas métricas e isto deve ser feito iterativamente (Gould e Lewis, 1985). Esta atividade requer que a equipe de design focalize os pontos adequados. O design integrado, em que todos os componentes de usabilidade são considerados para que evoluam conjuntamente, se inicia nesta etapa e prossegue na fase de seguinte;
3. **fase de desenvolvimento iterativo:** com os objetivos de usabilidade definidos e acesso rápido aos resultados dos testes com usuários, a modificação da interface e nova avaliação podem ser executados. Entretanto, isto só é possível com a elaboração de uma estratégia de implementação que permita a realização de testes dos elementos do design (desde as fases iniciais) e a modificação eficaz e barata da implementação em andamento. A separação entre a interface e o código funcional é fundamental;
4. **fase de instalação do sistema:** nesta etapa, os esforços se concentram em instalar o sistema nos locais de trabalho dos clientes, em utilizar os materiais de treinamento já desenvolvidos, em determinar e assegurar a aceitação pelos usuários. Suporte ao cliente e aspectos de manutenção do sistema devem ser testados nesta fase. A liberação para o cliente é, geralmente, uma outra iteração. Programas para a coleta de dados sobre o desempenho dos usuários podem ser incorporados.

2.1.4 Shneiderman

Shneiderman (1992) apresenta um “ciclo de vida para desenvolvimento de sistemas interativos” composto de oito fases. Testes são realizados em cada uma das

fases e sucessivas iterações entre elas são necessárias. Assim como Gould (1995), ele aponta que a usabilidade envolve muitos componentes além da interface propriamente dita e enfatiza a necessidade de um planejamento cuidadoso para assegurar confiabilidade, disponibilidade, segurança, integridade, padronização, portabilidade, integração e adequação ao cronograma e orçamento.

Shneiderman destaca ainda que o processo de design sugerido por ele é apenas um ponto de partida e que pode ser adaptado em função das necessidades específicas de cada projeto.

O processo de design proposto pelo autor está fundamentado, entre outras teorias, no modelo sintático-semântico de objeto-ação, também desenvolvido por ele (Shneiderman, 1992). O modelo sugere que os usuários têm conhecimento sintático sobre aspectos dependentes de dispositivos e conhecimento semântico sobre conceitos. O conhecimento semântico se divide, por sua vez, em conhecimento sobre a tarefa (domínio da aplicação) e sobre conceitos computacionais (domínio computacional).

1. **Fase de coleta de informação:** nesta etapa, deve-se coletar informações que irão direcionar o design sobre vários aspectos. O autor recomenda a execução de atividades de natureza gerencial como organizar a equipe de design, estimar custos de desenvolvimento, treinamento e manutenção, preparar cronograma com os pontos de liberação e revisão bem determinados e obter participação da gerência e dos clientes. Essas atividades, relativas à definição dos recursos disponíveis, têm impacto direto na seleção dos métodos e técnicas de design e avaliação de usabilidade. O autor recomenda a aplicação de entrevistas, questionários, análise de tarefas e da frequência com que são executadas. Sugere, ainda, que a equipe de design converse com usuários e com designers de sistemas similares, estude a literatura profissional e acadêmica sobre design e planeje uma estratégia de testes de usabilidade;
2. **definição dos requisitos e da semântica do sistema.** Os requisitos incluem, além da funcionalidade, segurança, privacidade, integridade e aderência aos padrões. Em relação à semântica, o autor recomenda a execução das seguintes tarefas:
 - 2.1. definir os objetivos gerais do sistema e dos requisitos funcionais;
 - 2.2. analisar alternativas para o seqüenciamento de tarefas;
 - 2.3. organizar as tarefas em unidades de transação;

2.4. definir os objetos e ações da tarefa (domínio da aplicação);

2.5. definir os objetos e ações computacionais.

O autor sugere ainda que se crie documentos com diretrizes e que se obtenha a concordância da gerência e dos clientes sobre os objetivos, requisitos e design da semântica;

3. **design da sintaxe e das facilidades de suporte.** Nesta etapa, o autor recomenda que se execute uma série de atividades relacionadas ao design da sintaxe do sistema, como comparar alternativas para o formato de tela, definir a sintaxe e projetar realimentação informativa para as ações, definir mensagens e tratamento de erros. A criação da sintaxe para as ações inclui o design de aspectos como tempo de resposta, *help on-line* e tutoriais. Deve-se iniciar os testes de usabilidade com protótipos e maquetes;
4. **especificação dos dispositivos físicos**, como tamanho e resolução da tela, requisitos de comunicação, etc. Deve-se realizar mais testes e revisões no design;
5. **desenvolvimento do software**, utilizando ferramentas para desenvolvimento de interfaces. Deve-se enfatizar a facilidade de modificação, a manutenibilidade, confiabilidade, segurança e documentação do software em desenvolvimento;
6. **integração do sistema e distribuição para os usuários.** Nesta fase, deve-se realizar testes de aceitação tanto para o software quanto para as facilidades de suporte;
7. **suporte à comunidade de usuários**, utilizando-se linhas telefônicas com consultores, entrevistas, *newsgroups*, caixas de sugestões, etc. Nesta etapa, deve-se conduzir avaliações objetivas e subjetivas e monitorar as frequências e padrões de uso, incluindo a ocorrência de erros;
8. **preparação de um plano de evolução.**

2.1.5 Nielsen

Nielsen (1993, pp. 72) ressalta a necessidade de se adotar a engenharia de usabilidade no desenvolvimento de famílias de produtos à medida que eles evoluem por períodos mais longos. Ele desenvolveu um modelo de ciclo de vida para

engenharia de usabilidade como um conjunto de onze estágios com o design iterativo enumerado explicitamente como um dos estágios.

1. **Conhecer os usuários.** O primeiro passo no processo de design é estudar os usuários do sistema e as características de uso do produto. Nielsen ressalta que o conceito de “usuário” inclui todas as pessoas cujo trabalho é afetado pelo uso do produto (ex., instaladores, administradores de sistema, pessoal de suporte, além dos “usuários finais”). Esta etapa se divide em quatro atividades:
 - 1.1. **identificação das características individuais dos usuários**, como experiência no domínio da aplicação, formação, idade, etc. É importante também conhecer o ambiente de trabalho e o contexto social. O autor sugere a utilização de questionários, entrevistas, estudos para observação e análises de marketing;
 - 1.2. **análise das tarefas**, o que inclui determinar tanto os objetivos gerais do sistema, quanto a maneira como os usuários abordam as tarefas, as informações de que necessitam e a maneira como tratam as exceções. As necessidades de comunicação dos usuários enquanto realizam ou se preparam para realizar as tarefas devem ser bem entendidas;
 - 1.3. **análise funcional**, que ultrapassa a observação sobre como os usuários executam correntemente as tarefas (atividade de análise de tarefas), e investiga qual a razão funcional para a execução da tarefa. A análise funcional tem por objetivo distinguir o que realmente precisa ser feito dos procedimentos superficiais que podem e até devem ser modificados no processo de execução de tarefas dos usuários;
 - 1.4. **evolução dos usuários e das tarefas que executam**. Nielsen observa que o uso do sistema modifica os usuários e que, ao se modificarem, eles também utilizam o sistema de maneiras diferentes. Para suportar novas formas de utilização, o design precisa ser flexível;
2. **análise competitiva**. Nesta etapa, deve-se avaliar as versões correntes do produto sob design e os produtos concorrentes. O autor sugere uma avaliação heurística para verificar a conformidade desses produtos às diretrizes de usabilidade e a realização de testes com usuários para analisar a funcionalidade e as estratégias de interação utilizadas;

3. **definição dos objetivos de usabilidade.** Antes de iniciar o design, é importante discutir as métricas de usabilidade de interesse para o projeto e especificar os objetivos de usabilidade da interface em termos mensuráveis. Para novas versões de sistemas ou para sistemas que tenham um concorrente bem definido no mercado, é relativamente fácil definir os objetivos de usabilidade. Nos casos de produtos completamente novos, este processo é bem mais complexo. É importante fazer uma análise do impacto financeiro que as melhorias de usabilidade trarão ao novo sistema;
4. **design paralelo,** fase em que vários designers diferentes produzem designs preliminares. O objetivo do design paralelo é explorar diferentes alternativas de design antes de optar por uma abordagem específica. Nesta etapa, o enfoque é a diversidade de soluções. Assim, os designers devem trabalhar independentemente. A solução final pode ser uma combinação dos vários designs produzidos. O autor enfatiza que o design paralelo é especialmente importante para o design de produtos completamente novos. Para os casos mais tradicionais, para os quais existem concorrentes bem conhecidos, a análise competitiva funciona como um design paralelo inicial;
5. **interação com os usuários.** Depois de iniciado o design, os designers devem manter encontros regulares com usuários representativos da comunidade alvo (pessoas que irão efetivamente usar o sistema). Essa atividade é importante para evitar discrepâncias entre o modelo real de tarefas dos usuários e o modelo de design⁴. Não se deve esperar que os usuários proponham idéias sobre o design; entretanto, observar suas reações a designs concretos que lhes são apresentados é particularmente útil. Nielsen enfatiza que é necessário apresentar as soluções propostas em uma forma que os usuários possam entender (ex., protótipos e, nas fases mais iniciais do design, maquetes de papel ou apenas alguns esquemas de telas);
6. **coordenação do conjunto da interface.** O conjunto da interface de um sistema é composto por vários elementos (ex., as telas do sistema, a documentação, o *help online*, tutoriais, materiais de treinamento) veiculados através de diferentes meios. É necessário que haja consistência entre esses vários elementos. Deve-se observar ainda a consistência entre versões sucessivas de um produto e entre produtos de uma mesma família. Nielsen

⁴ O modelo de design é o modelo conceitual que os designers têm sobre o sistema a ser construído.

sugere que, para se alcançar consistência no conjunto da interface, é necessário haver uma pessoa central no projeto para coordenar os vários aspectos da interface;

7. **avaliação heurística e aplicação de diretrizes.** As diretrizes listam um conjunto de princípios de design de interfaces bem conhecidos que devem ser seguidos durante o desenvolvimento. Técnicas de inspeção podem ser usadas para avaliar a conformidade do design a esses princípios. A avaliação heurística reúne dez princípios que explicam uma grande proporção dos problemas de usabilidade usualmente encontrados em uma interface. Esta técnica, proposta por Nielsen, orienta os designers na inspeção da interface com relação a esses princípios;
8. **prototipagem.** A avaliação de usabilidade deve, nas fases iniciais do design, basear-se em protótipos. Eles podem ser desenvolvidos rapidamente a baixos custos, o que permite modificá-los várias vezes até se alcançar um melhor entendimento sobre o design da interface. Nielsen ressalta que, para se utilizar protótipos, a interface e o código devem ser desenvolvidos separadamente. O autor observa ainda que os protótipos são uma forma de especificação da interface e são frequentemente usados como uma maneira de comunicar o design final aos desenvolvedores;
9. **avaliação da interface.** Os métodos de avaliação de interfaces são classificados segundo os seguintes critérios: se há ou não participação dos usuários e se o sistema foi ou não realmente implementado. Independentemente do método utilizado, uma avaliação de interface produz como um dos resultados uma lista dos problemas de usabilidade detectados. Nielsen recomenda que especialistas em usabilidade atribuam graus de severidade a esses problemas, a fim de que se possa determinar as prioridades para solucioná-los;
10. **design iterativo.** Com base nos resultados da avaliação (problemas detectados e prioridades de solução) deve-se produzir uma nova versão da interface. O design modificado deve ser testado novamente tão logo quanto possível, para que outros problemas que tenham sido mascarados pelos problemas iniciais possam ser detectados. Nielsen enfatiza que os usuários que colaboram no projeto na fase de interação (item 5 desta lista) não devem participar dos testes;

11. estudos subseqüentes dos sistemas instalados. Após a liberação, o principal objetivo de usabilidade é obter dados sobre a usabilidade do sistema para o design de versões futuras. Nielsen recomenda a utilização de vários métodos, como entrevistas, questionários, observação *in loco* das reações dos usuários e uso de arquivos *log* provenientes de versões instrumentadas.

Após a apresentação do ciclo de vida para a Engenharia de Usabilidade, Nielsen enfatiza dois pontos:

- na aplicação deste processo, é necessário priorizar algumas atividades de usabilidade, já que, muitas vezes, não é possível executar todas as atividades recomendadas;
- para que a aplicação dos métodos sugeridos seja bem sucedida, é necessário complementá-los com os seguintes meta-métodos:
 - elaborar um plano de trabalho explícito para aplicar o método, com a descrição (documentada) das informações necessárias e dos passos a serem seguidos. Um plano para realização de testes de usabilidade, por exemplo, deve incluir, entre outras, informações sobre o número de usuários que participarão do teste, suas características, quais as tarefas que eles deverão executar, cronograma para a realização dos testes;
 - submeter o plano de trabalho a uma revisão realizada por uma pessoa que não se inclui na equipe de design;
 - realizar uma atividade piloto empregando cerca de 10-15% dos recursos destinados à aplicação do método. Deve-se então revisar o plano e corrigir as dificuldades detectadas durante a execução do piloto.

2.1.6 Avaliação sobre processos de design centrados no conceito de usabilidade

Os processos de design apresentados têm em comum a iteratividade, o empirismo e o foco nos usuários, princípios fundamentais para o design de sistemas interativos.

Um aspecto central é a definição de objetivos de usabilidade testáveis. Cada iteração tem por objetivo verificar pontos em que o design pode ser melhorado,

considerando os objetivos de usabilidade, para aproximar o resultado final do alvo inicial. Gould aponta que a definição de valores alvo para cada uma das métricas é uma tarefa árdua que por si só pode demandar várias iterações.

A definição dos objetivos de usabilidade nesses processos é direcionada a dois elementos principais:

- tempo e esforço empregados pelos usuários na execução de tarefas;
- atributos do sistema, como tempo de resposta e disponibilidade, que têm um impacto na produtividade e reações dos usuários.

Um aspecto de particular importância ressaltado por Nielsen: a aplicação de métodos de design e avaliação de usabilidade pressupõe o uso de meta-métodos. A aplicação de um método, assim como o próprio processo de design, requer iteratividade e empirismo (i.e., execução de testes piloto e correções). Assim, a presença de ciclos de design-implementação-avaliação-modificação pode ser observada tanto em um nível macroscópico, quando envolve a interface como um todo, quanto em um nível mais local, quando se trata da aplicação de métodos (ex., questionários, testes de usabilidade).

A seguir, os processos são comentados segundo alguns aspectos particulares.

Quanto à generalidade dos processos.

Enquanto o modelo proposto por Gould (seção 2.1.3) pode ser considerado uma metodologia informal, os modelos propostos por Shneiderman (seção 2.1.4) e Nielsen (seção 2.1.5) são mais elaborados.

Gould divide o processo cronologicamente em quatro fases, sem preocupações em refinar esta divisão. Em cada fase predominam determinados objetivos e a conseqüente utilização de métodos para alcançá-los. Na fase de design inicial, por exemplo, os designers coletam informações críticas sobre os usuários e as tarefas que desempenham. Entretanto, ao longo de todo o design e até mesmo depois da instalação do sistema, os designers devem continuar coletando informações mais específicas sobre o uso, contexto de trabalho, etc. Nesta fase também se inicia a preocupação com o design integrado, em que os vários aspectos de usabilidade são considerados e planejados, o que também se estende por todo o desenvolvimento. Várias atividades ocorrem simultaneamente e se prolongam por várias fases, mas elas são mais prioritárias em algumas.

Essa mesma percepção se reflete no processo de Rubinstein e Hersh (seção 2.1.2), que propõe uma sucessão de iterações das etapas de coleta de informações, design, implementação, teste e avaliação e liberação. Como no processo proposto por Gould, essas atividades coexistem durante todo o processo.

Gould propõe uma seqüência de fases relativamente difusa, mas bastante genérica, o que reflete a percepção do autor de que não há uma prescrição geral, no estilo de “livro de receitas”, que possa ser aplicada a todas as situações. Já Shneiderman e Nielsen são mais específicos, mas ressaltam a necessidade de adaptar o processo às particularidades de cada projeto, o que também exige habilidade e prática dos designers.

Quanto à integração com as metodologias de desenvolvimento

As metodologias apresentadas destacam a necessidade de design integrado dos vários aspectos de usabilidade (ex., confiabilidade, manutenibilidade). Gould e Nielsen destacam a importância de uma pessoa central no projeto, que coordene todos os aspectos de usabilidade do sistema. Shneiderman recomenda a realização de atividades de natureza gerencial que irão definir os recursos disponíveis para o design.

Essas diretrizes revelam a preocupação dos autores em garantir que a usabilidade seja um atributo prioritário para o projeto e que os próprios gerentes estejam preocupados em assegurar a existência de recursos e condições para promovê-la.

Gould (1995) ressalta que a equipe de desenvolvimento deve estar comprometida e motivada para realizar rapidamente as modificações no design. Ele coloca, ainda, que a iteratividade “só é possível com a elaboração de uma estratégia de implementação que permita a realização de testes dos elementos do design (desde as fases iniciais) e a modificação eficaz e barata da implementação em andamento. A separação entre a interface e o código funcional é fundamental (p. 109)”. Shneiderman também enfatiza a necessidade de separação entre a interface e o código.

Essas são preocupações legítimas. Se, por um lado, os processos de design reconheceram a necessidade e se propõem a praticar uma abordagem empírica como condição necessária para assegurar a qualidade de sistemas interativos, essa prática só é possível se o processo de desenvolvimento do software estiver estruturado para isso.

Nielsen sugere o uso de protótipos como estratégia para assegurar a iteração no processo. Ele argumenta que os protótipos podem servir tanto para comunicar aos usuários sobre a interface quanto para comunicar as especificações do design para os desenvolvedores, demonstrando uma preocupação com a comunicação entre as equipes de design e desenvolvimento.

Os processos apresentados propõem o design empírico, iterativo e centralizado nos usuários como a única abordagem factível para se desenvolver software interativos, ou seja, usabilidade deve ser um aspecto a ser considerado nas metodologias de desenvolvimento da Engenharia de Software.

2.2 A herança da Engenharia de Software

Pesquisas mostram que a interface consome de 47% a 60% das linhas de código de um sistema (MacIntyre *et al.*, 1990). Pelo menos 29% do orçamento (Rosenberg, 1989) e 40% do esforço (Karat, 1994) para desenvolvimento de software são dedicados à interface.

Por outro lado, 80% dos custos do ciclo de vida de um software são consumidos na fase de manutenção (Pressman, 1995). Uma revisão no trabalho de manutenção aponta que 80% dos problemas encontrados são requisitos não detectados ou não alcançados e 20% são erros ou problemas de confiabilidade (Martin e McClure, 1983).

Pressman (1995) estima que o custo de uma alteração seja de 1 unidade na fase de definição de um projeto, de 1,5 a 6 unidades na fase de desenvolvimento e de 60 a 100 unidades na fase de manutenção. A definição criteriosa dos requisitos, a avaliação da usabilidade de protótipos e a realização de *walkthroughs* para inspecionar a usabilidade são estratégias que, se utilizadas desde as fases iniciais de um projeto, reduzem significativamente os custos para identificar e solucionar problemas de usabilidade e o tempo de desenvolvimento (Karat, 1994).

A análise desses dados revela que a engenharia de usabilidade tem potencial para fazer contribuições significativas na qualidade do software e no sucesso financeiro dos projetos, uma vez que uma parcela substancial do esforço e dos recursos de desenvolvimento de um projeto é dedicada à interface.

Contudo, os métodos de desenvolvimento de software empregados atualmente foram propostos antes que o software interativo para usuários finais se tornasse importante (Grudin, 1991). Eles são centrados nos desenvolvedores e na produção de software e reduzem o design da interface a um plano subordinado.

Na próxima seção, os contextos de desenvolvimentos de software são apresentados, mostrando as diferentes condições para a participação dos usuários. Em seguida, alguns dos processos de desenvolvimento da Engenharia de Software são discutidos.

2.2.1 Os contextos de desenvolvimento de software

Três contextos de desenvolvimento de software provêm o embasamento para se entender os projetos de desenvolvimento de software interativos: desenvolvimento por contratação, de produtos de prateleira e interno (Grudin, 1991). Grudin aborda, em cada contexto, o momento em que os usuários e os desenvolvedores são identificados.

No desenvolvimento por contratação, a organização dos usuários é conhecida desde o início do projeto, mas os desenvolvedores só são identificados após uma concorrência em que a proposta mais vantajosa é selecionada. O caso mais típico é o de licitações públicas, em que uma empresa do governo prepara uma especificação do software e licita o desenvolvedor.

No caso de desenvolvimento de produtos de prateleira, os desenvolvedores são conhecidos desde o início do projeto, mas os usuários freqüentemente permanecem desconhecidos até que o produto seja comercializado.

No caso de desenvolvimento interno, tanto os desenvolvedores quanto os usuários são conhecidos desde o início do projeto (ex., a seção de informática de um banco desenvolvendo serviços para os gerentes).

Combinações desses três paradigmas podem ocorrer. Quando um contrato é negociado sem concorrência, por exemplo, os desenvolvedores podem participar muito mais cedo do projeto, uma situação em que coexistem características de desenvolvimento por contratação e interno.

As condições para participação dos usuários no processo de design variam muito entre os três contextos de desenvolvimento. As recomendações em IHC ressaltam que os desenvolvedores devem estabelecer e manter contato com os usuários durante todo o desenvolvimento. Grudin (1991) apresenta uma revisão detalhada destes contextos, com o histórico, oportunidades e dificuldades para o design de IHC e os mediadores que podem aproximar desenvolvedores e usuários em cada contexto.

O autor aponta que cada um dos paradigmas trouxe contribuições para as práticas correntes de desenvolvimento de sistemas interativos: o desenvolvimento por contratação foi central para a evolução dos métodos de gerenciamento de projeto de software; o desenvolvimento de produtos de prateleira focalizou interfaces para

usuários individuais; o desenvolvimento interno explorou aspectos sociais e organizacionais da utilização de sistemas (Grudin, 1991, p. 61). Um breve resumo desse trabalho é apresentado a seguir.

Desenvolvimento por contratação. O modelo cascata de desenvolvimento de software, fundamentado em uma seqüência rígida de fases, se tornou a base da maioria dos padrões de aquisição de software pelo governo norte-americano⁵. Suas características o tornam uma opção natural para o desenvolvimento por contratação, já que a organização dos usuários determina a viabilidade, produz a especificação de requisitos e então celebra um ou mais contratos para o design, desenvolvimento, administração e manutenção.

Entretanto, a ênfase do modelo cascata no design inicial do software é mais apropriada para sistemas não interativos, mais previsíveis em relação aos requisitos. A rigidez na seqüência de fases inerente ao modelo torna mínimas as oportunidades para prototipagem e design iterativo, condições fundamentais para a participação dos usuários.

Desenvolvimento de produtos de prateleira. A ênfase do desenvolvimento de produtos de prateleira é produzir produtos atraentes para os usuários, especialmente porque, com um grande número de possíveis usuários, um produto pode ser bem sucedido se apenas uma fração do mercado julgá-lo aceitável. Inicialmente, os produtos de prateleira focalizavam a funcionalidade do software. À medida que o mercado foi amadurecendo e a usabilidade se tornou uma questão importante, os desenvolvedores de produtos de prateleira passaram a focalizar aspectos genéricos do diálogo entre usuários e sistemas computacionais que são comuns à maioria dos usuários (ex., aspectos relacionados ao controle motor, percepção e processos cognitivos elementares).

Desenvolvimento interno. Neste contexto de desenvolvimento, o critério para conclusão bem sucedida de um projeto é que o sistema seja aceito por um conjunto pré-definido de usuários. Em tais circunstâncias, aspectos sociais e organizacionais do uso têm grande importância. Por outro lado, este contexto apresenta uma característica que estimula a participação dos usuários no design: os desenvolvedores e usuários são conhecidos desde que o projeto é iniciado. As abordagens de design baseadas na participação ativa dos usuários ganharam força, especialmente em projetos ingleses e escandinavos.

⁵ Nas fases iniciais do desenvolvimento comercial da computação, os contratos do governo norte-americano representaram um grande estímulo para a indústria de software.

2.2.2 A influência da Engenharia de Software

Na Engenharia de Software, o design é visto como um processo formal de derivação de um sistema a partir das especificações (Denning e Dargan, 1996). Esse processo estabelece uma ordem para execução dos diversos estágios envolvidos no desenvolvimento e evolução de software e critérios de transição de um estágio para outro (Boehm, 1988).

Nas fases iniciais de desenvolvimento de software, o processo de desenvolvimento envolvia basicamente duas etapas:

- escrever um código;
- corrigir os problemas no código.

Nesse processo, denominado *code-and-fix* (Boehm, 1988), a ordem dos estágios envolvia primeiro desenvolver um código e depois pensar sobre os requisitos, design, teste e manutenção. Este modelo apresenta três dificuldades primordiais:

- após algumas correções, o código tornava-se tão mal estruturado que o custo para as correções subseqüentes era muito alto;
- freqüentemente, o software produzido não correspondia às necessidades dos usuários e era rejeitado ou desenvolvido novamente. A necessidade de uma fase de levantamento de requisitos antes do design tornou-se evidente;
- já que não havia uma preparação prévia para testes e modificação, o custo para correção do código era alto. Isso evidenciou a necessidade dessas fases.

Durante a década de 50, a contratação de desenvolvimento de software para grandes projetos pelo governo norte-americano levou ao reconhecimento desses problemas e ao desenvolvimento do modelo de estágios (*stagewise model*), que estabelecia que o software deveria ser desenvolvido em estágios sucessivos (plano operacional, especificações operacionais, especificações para codificação, codificação, testes e avaliação do sistema).

Como resultado de refinamentos sucessivos do modelo de estágios durante os anos 50 e 60, surgiu, na década de 70, o modelo cascata, de grande influência no desenvolvimento de software. O modelo cascata foi empregado para orientar o desenvolvimento por contratação de grandes sistemas não interativos. Ele organiza o processo em um conjunto de fases progressivas: estudo de viabilidade, análise de requisitos, design preliminar, codificação, instalação e manutenção. Em relação ao

modelo de estágios, o modelo em cascata apresentou duas contribuições principais (Boehm, 1988):

- o reconhecimento de ciclos de realimentação entre estágios, apesar da orientação para mantê-los apenas entre estágios adjacentes a fim de minimizar os altos custos de “retrabalho” que a realimentação entre vários estágios envolve;
- uma incorporação inicial do uso de prototipagem no ciclo de desenvolvimento, que se restringe, contudo, ao passo “*build-it-twice*” (Boehm, 1988; Grudin, 1991), realizado em paralelo com a análise de requisitos e o design.

É importante ressaltar que o modelo em cascata evoluiu inserido no contexto de desenvolvimento por contratação (Grudin, 1991). Dois aspectos importantes estão relacionados a esse contexto (Baecker *et al.*, 1995):

- os desenvolvedores e usuários estavam separados por barreiras espaciais, temporais e organizacionais;
- os sistemas desenvolvidos eram não interativos.

A abordagem apresentada pelo modelo cascata de se realizar primeiro o design do software e, depois de finalizá-lo, prosseguir no desenvolvimento, era muitas vezes a melhor opção para o desenvolvimento por contratação. Entretanto, no caso de desenvolvimento interno, os desenvolvedores preferiam manter um contato mais próximo com os usuários durante o desenvolvimento (Grudin, 1991). Surgiram então algumas abordagens que incorporavam a participação dos usuários no processo.

É importante notar ainda que, no caso de desenvolvimento de sistemas interativos, dificilmente é possível produzir um design adequado na primeira tentativa. A definição da funcionalidade e das estratégias de interação acaba demandando iteração e prototipagem. A ênfase do modelo cascata em documentos muito elaborados como critério de conclusão das fases de especificação de requisitos e design dificultava o desenvolvimento de sistemas interativos. A preparação de especificações elaboradas sobre interfaces e funções de suporte à decisão mal compreendida era seguida pelo design e desenvolvimento de um grande volume de código sem utilidade.

Boehm (1988) observou as falhas do modelo cascata, especialmente para o desenvolvimento de sistemas interativos. Ele propôs o modelo de desenvolvimento espiral, projetado especificamente para incorporar iteração e prototipagem.

O modelo espiral reflete o conceito de que cada ciclo envolve uma progressão da mesma seqüência de passos, para cada parte do produto e para cada um dos seus níveis de elaboração, desde o documento com o conceito operacional geral do sistema até a codificação de cada programa individualmente.

Apesar de incorporar a iteração e prototipagem, o modelo espiral continua sendo centrado nos desenvolvedores e no software, e não nos usuários e no contexto em que utilizam os artefatos.

2.3 Considerações finais

Os esforços para estruturar os princípios, técnicas, métodos e diretrizes de IHC em metodologias de design trouxeram grandes avanços. O conceito de design iterativo, com sucessivas etapas do ciclo design-implementação-avaliação foi reconhecido cedo. A fase de avaliação centrada na participação dos usuários também é amplamente reconhecida. Essas realizações mudaram o paradigma de design de software interativo.

Dois aspectos fundamentais estão relacionados ao sucesso de uma metodologia de design: (1) a aplicação de métodos e técnicas de design e avaliação de usabilidade para coletar dados sobre os usuários e sua interação com o sistema; (2) o uso desses dados para suportar as decisões de design, o que implica a combinação dos mesmos com todos os demais aspectos envolvidos no processo de desenvolvimento (ex., design do software, codificação, objetivos de *marketing*).

O primeiro desses aspectos, ou seja, a qualidade da coleta de dados sobre a interação, depende:

- da adequação dos métodos e técnicas de design e avaliação de usabilidade ao contexto específico de um dado projeto, que propicia oportunidades e impõe uma série de limitações. Esse fator tem impacto direto na generalidade de uma metodologia de design, pois dificulta a proposição de passos específicos, bem definidos, e que sejam igualmente genéricos. Isso pode ser observado nas diretrizes propostas pelas metodologias apresentadas. Enquanto Gould propõe um processo informal e genérico, Shneiderman e

Nielsen são mais específicos, mas ressaltam a necessidade de haver espaço para adaptações considerando as necessidades específicas de cada projeto;

- da eficácia da aplicação de cada método ou técnica de design ou avaliação de usabilidade. Nielsen enfatiza a importância da preparação de cada estudo de usabilidade ao recomendar o uso de meta-métodos, que envolve a mesma abordagem iterativa e empírica do próprio processo de design da interface. Esse aspecto é também discutido por Draper e Norman (1985), que estabelecem uma analogia entre o design do software e o design da interface. Os autores apontam que, assim como o teste de software precisa ser planejado de tal forma que os conjuntos de dados de testes exercitem cada trecho do código, o teste de usabilidade precisa verificar todos os “trechos” da interação (p. 256). Assim, é necessário desenvolver bons problemas de teste, encontrar um grupo de participantes adequado e que seja representativo do público alvo do sistema, realizar observações cuidadosas e avaliar os resultados.

O segundo aspecto, o uso dos dados coletados para suportar as decisões de design, implica determinar o que deve ser feito para solucionar um problema de usabilidade e disponibilizar os recursos necessários para corrigi-lo. Isso envolve negociações com toda a equipe, pois é necessário determinar o custo/benefício de cada decisão de design. Como foi observado na descrição dos vários processos de design, é necessário definir uma estratégia de desenvolvimento que permita que os resultados das avaliações resultem em modificações efetivas. Isto significa que o processo de design precisa focalizar a interação entre a equipe de design e os desenvolvedores, testadores, equipe de *marketing*, e todos os demais grupos que colaboram para a criação do produto final.

Por outro lado, os processos de desenvolvimento da Engenharia de Software focalizam primordialmente o design e codificação do software. O modelo cascata, fundamentado essencialmente na produção de documentos elaborados como critérios de transição para os estágios de análise de requisitos e design da interface, é muitas vezes adequado para o desenvolvimento por contratação, mas apresenta grandes dificuldades para o desenvolvimento de sistemas interativos, que exige uma iteratividade não suportada pelo modelo. O modelo espiral, por sua vez, apesar de haver incorporado iteração e prototipagem, é centrado nos desenvolvedores e na produção de software, em vez de ser centrado nos usuários e em seu contexto de trabalho. Além disso, Hix e Hartson (1993) apontam que o design de interface

requer mais iteração, em uma granularidade diferente da proposta pelo modelo espiral.

Melhorias progressivas têm ocorrido desde o surgimento do modelo espiral, mas mesmo os refinamentos mais recentes das abordagens da Engenharia de Software que têm incorporado aspectos de IHC enfatizam muito a funcionalidade e relegam a interface a um plano secundário (Baecker *et al.*, 1995). Abordagens de análise e design estruturados limitam o design da interface a uma subfase do desenvolvimento.

Se, por um lado, a Engenharia de Software perde com a falta de integração com os processos de design, até mesmo em função do impacto financeiro que a falta de entendimento sobre o trabalho dos usuários pode acarretar, por outro, as metodologias de design não são efetivas sem que haja uma estratégia de desenvolvimento que permita que os resultados das avaliações resultem em modificações rápidas e efetivas, viabilizando o design iterativo. A necessidade de integração entre esses processos é evidente.

Essa lacuna entre os processos de design e desenvolvimento e seu conseqüente prejuízo para a qualidade da interação foi observada e discutida por alguns autores.

Draper e Norman (1985) resumem as questões discutidas ao afirmarem que “é necessário apenas encarar o problema de engenharia da interface de usuários tão seriamente quanto qualquer outra parte da engenharia de software e aplicar os mesmos tipos de técnicas, adaptadas de maneira apropriada” (p. 252).

Hartson e Hix (1989) apontam a falta de conexão entre as metodologias de design de interface e as abordagens de desenvolvimento de software e a necessidade de se tratar o design e desenvolvimento da interface como parte integral do processo de Engenharia de Software. Eles propuseram uma metodologia de design, o ciclo de vida estrela (Hartson e Hix, 1989; Hix e Harston, 1993), em que o design da interação pode começar teoricamente com qualquer atividade do desenvolvimento e prosseguir com qualquer outra. Contudo, as várias atividades são altamente interconectadas através da avaliação de usabilidade no centro da estrela (Figura 2.1).

Assim como o modelo espiral da Engenharia de Software, o modelo estrela inclui as fases de teste e prototipagem. Entretanto, ao contrário do primeiro, que focaliza o desenvolvimento do software, o modelo estrela foi especificamente criado para melhorar a usabilidade do produto, já que a avaliação da interface ocorre continuamente ao longo de todo o desenvolvimento.

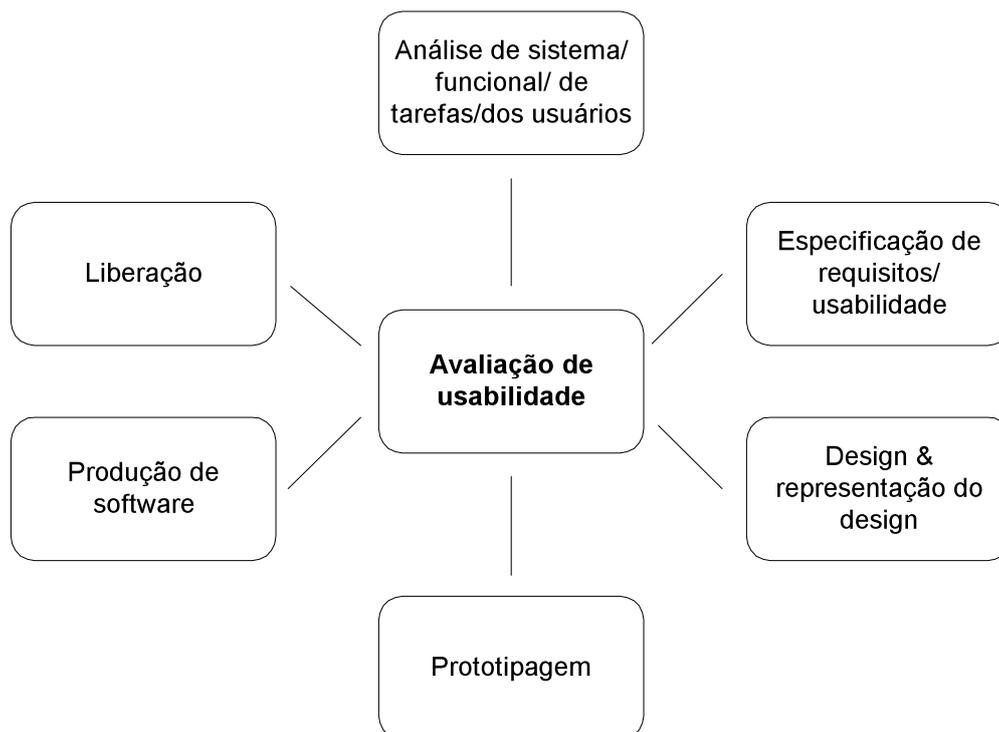


Figura 2.1: Ciclo de vida estrela para desenvolvimento da interação com os usuários.
 Extraído de Hix e Hartson, 1993, p. 102.

Hix e Hartson enfatizam a importância da comunicação para diminuir a lacuna entre os processos de design de interface e desenvolvimento de software e apontam para a necessidade de técnicas de representação – mecanismos que veiculam as especificações e design da interface em progresso (Hartson e Hix, 1989). Contudo, eles focalizam a comunicação entre os diversos agentes que integram a equipe de design, mas não a comunicação entre a equipe de design e os demais agentes do processo de desenvolvimento (ex., desenvolvedores, gerentes, testadores, equipe de *marketing*). Eles afirmam que, “apesar de tal comunicação ser essencial para manter a equipe de desenvolvimento integrada, os canais utilizados para esse fim ainda não são bem conhecidos” (Hix e Hartson, 1993, p. 112).

Em um projeto de desenvolvimento de software, apesar dos objetivos e interesses da organização desenvolvedora serem claros, os diversos indivíduos e grupos que a integram podem compartilhá-los de maneiras diferentes, considerando sua formação, perspectiva, função no processo. Assim, a cooperação entre esses agentes é fundamental para se identificar e solucionar os compromissos de custo/benefício que surgem no curso do desenvolvimento e ainda para se criar uma

solução da organização como um todo para seus clientes. Isso está intrinsecamente relacionado à eficácia da comunicação entre tais indivíduos e grupos.

Esse aspecto foi observado por Beyer e Holtzblatt, (1998):

“There are the rest of the engineers on the project who have to believe in the system enough to code it. There are the project teams working on systems that have to interface to yours. There is your manager, his manager (...) There is the marketing and the product-planning department who tend to be skeptical of ideas coming out of engineering. There is the sales force of a commercial product, which needs to understand what makes the new product worth selling. And there are the customers, who need to be convinced that the new system will improve their lives” (p. 199).

Os autores apontam que não é possível integrar, em uma equipe de design, todos os indivíduos que participam do projeto e se preocupam com seus resultados. Por outro lado, é possível criar uma equipe de design multifuncional, que inclua pessoas de diferentes grupos funcionais de uma organização (ex., desenvolvimento, teste), o que permite que a perspectiva de cada grupo seja incorporada ao design. Contudo, a participação desses representantes não garante que as soluções emergentes sejam comunicadas aos grupos de forma efetiva para que eles percebam como podem influenciá-las.

O Design Contextual (Beyer e Hotlzblatt, 1998) propõe uma série de técnicas que força os diferentes membros da equipe a interagirem com dados, a compartilharem um espaço para analisar dados sobre os clientes, a analisarem modelos de trabalho e, principalmente, a se engajarem em uma conversação em que cada membro apresente sua perspectiva e contribua para o projeto.

Apesar dessa abordagem focalizar intensamente a comunicação entre os vários agentes do processo, ela requer o uso de técnicas e mecanismos de comunicação específicos. Entretanto, assim como cada contexto de desenvolvimento (i.e., interno, de produtos de prateleira e por contratação) é mais ou menos favorável ao uso de determinadas técnicas de design e avaliação de usabilidade, cada projeto específico apresenta particularidades que facilitam ou dificultam o uso de determinadas estratégias. Por exemplo, o próprio Design Contextual demanda o uso de um espaço físico, a sala de design, em que os diversos diagramas e modelos gerados com a aplicação das técnicas propostas são afixados. Apesar da existência desse espaço ser altamente desejável e benéfica, muitos projetos têm limitações que inviabilizam a sua criação.

Em síntese, os seguintes pontos são importantes na perspectiva deste trabalho:

- existe uma lacuna entre as metodologias de design da interface e da Engenharia de Software que foi identificada na literatura. Tais metodologias se beneficiariam mutuamente de uma maior integração. Ao aproximar esses processos, a organização de criação de software estaria mais apta a prover uma resposta unificada e coerente a seus clientes;
- a integração entre tais processos depende da comunicação entre os diversos grupos de agentes que atuam em uma organização de criação de software. Para que as atividades de design da interface e do software não sejam disjuntas, essa comunicação precisa ser efetiva;
- essa comunicação se dá através de mecanismos formais e informais. Contudo, os mecanismos utilizados e o impacto dessa comunicação no produto final estão fortemente relacionados às características específicas de cada projeto.

A aproximação entre tais processos é um passo em direção ao que chamamos “processo de design do software”, no Capítulo 1, em que todas as atividades inerentes ao design e desenvolvimento se afetam mutuamente e se complementam para viabilizar a criação de um produto que causa impacto na vida dos usuários, individual e socialmente.

Capítulo 3

Objetivos e Metodologia

Grande parte dos esforços de pesquisadores e profissionais que atuam na área de IHC tem sido no sentido de criar sistemas computacionais com maior usabilidade. Foram desenvolvidos diversos métodos, técnicas, metodologias, princípios e teorias para suportar o design e avaliação de usabilidade.

As práticas correntes de desenvolvimento de software, contudo, apresentam uma série de obstáculos e oportunidades para utilização desse conhecimento. Deve-se observar, ainda, que o simples fato de um conjunto de procedimentos de design ser aplicado usualmente em um processo de desenvolvimento não assegura que os dados resultantes sejam utilizados de maneira apropriada pela equipe de criação do software, resultando em melhorias efetivas na usabilidade do produto.

Assim, um dos objetivos deste trabalho é entender os vários níveis da atividade de design no processo de criação de um software e os efeitos que eles acarretam entre si:

- o design da interface do produto. Neste nível, designers/engenheiros de usabilidade propõem um design de interface através da qual os usuários irão perceber o produto;
- o design do software, que envolve o planejamento e implementação do código. Este nível envolve a execução do que foi proposto no nível de design da interface, considerando as limitações técnicas existentes;
- o design do produto como um todo, que corresponde à solução alcançada pela organização, considerando seus objetivos e limitações, inclusive gerenciais, para as necessidades e demandas dos usuários.

Considerando-se esses níveis, é possível perceber que várias forças atuam na negociação de uma solução. Essa negociação envolve uma conversação entre os

diferentes agentes do processo de design e desenvolvimento. Na perspectiva adotada neste trabalho, ela será tanto mais eficaz quanto melhor for a comunicação entre a equipe de design, usuários e os outros agentes. Assim, é igualmente importante entender as diferentes dimensões da comunicação no processo de design:

- a comunicação entre a equipe de design e os usuários que permite que as necessidades e interesses dos usuários, o contexto de utilização do produto, os problemas de usabilidade das soluções propostas sejam identificados;
- a comunicação entre a equipe de design e toda a equipe de criação do produto. As informações obtidas na conversação entre os primeiros e os usuários precisam ser propagadas para agentes chave no processo de desenvolvimento, para que possam ser efetivamente incorporadas ao produto. Por outro lado, as soluções de design precisam refletir também as limitações e objetivos do projeto;
- a comunicação entre a equipe de criação e os usuários, através, principalmente, da equipe de design. É importante considerar também a conversação que outros agentes estabelecem com os usuários, como a equipe de suporte ao produto, o *marketing*, etc.

Através do entendimento dos diversos níveis de design no processo de design/desenvolvimento e da comunicação envolvida nesses processos, é possível identificar deficiências de comunicação que resultam em prejuízo para a qualidade do produto. A fim de entender essas questões e investigar estratégias para inspecionar o processo quanto à comunicação, o desenvolvimento deste trabalho envolveu as seguintes etapas:

- revisão dos processos de design, baseados em uma abordagem empírica, iterativa e centrada nos usuários. Isso demanda uma estratégia de implementação que permita que o design seja modificado de maneira rápida e efetiva. Revisão dos processos da Engenharia de Software, centrados no projeto e desenvolvimento de software. Foi identificada uma lacuna entre os processos derivados de IHC e de ES;
- estudo do conceito de comunicação;

- proposição de um novo modelo, o modelo fractal de comunicação, que permite que as diversas dimensões de comunicação envolvidas no processo de design/desenvolvimento sejam representadas;
- proposição de um método de inspeção do processo de design quanto aos aspectos de comunicação. O método auxilia na identificação de deficiências de comunicação entre a equipe de design, usuários, e a equipe de criação do produto como um todo;
- realização de um estudo de casos para ilustrar a aplicação do método.

Capítulo 4

Modelo fractal de comunicação aplicado ao processo de design de interfaces

O processo de design da interface de sistemas computacionais envolve muitos atos de comunicação que ocorrem através de diferentes canais. Testes de usabilidade, por exemplo, têm sido usados como um canal de comunicação fundamental entre designers/engenheiros de usabilidade e usuários. Outros canais usados nesse processo incluem análise de requisitos, construção de cenários, prototipagem, etc. Um dos objetivos deste trabalho é entender como eles atuam nas conversações entre os agentes envolvidos no processo de design/desenvolvimento.

Foram analisados alguns modelos de comunicação apresentados na literatura e foi proposto um modelo fractal, que, em função de sua natureza recursiva, permite que se amplie ou limite o foco de observação, trazendo flexibilidade à análise do processo de conversação como um todo ou de cada ato de comunicação que o integra.

O modelo fractal proposto foi, então, projetado ao contexto do processo de design/desenvolvimento de sistemas interativos.

4.1 Introdução

O enfoque deste trabalho é a comunicação envolvida no processo de design da interface e na utilização de sistemas computacionais. Um aspecto desta comunicação é o diálogo entre a equipe de design (designers e engenheiros de usabilidade) e usuários. Como apontam Adler e Winograd (1991, p. 7), “este tipo de comunicação é

intrínseco a todos os tipos de artefato. Através da sua estrutura e aparência, os objetos expressam mais ou menos efetivamente o que eles são, como eles devem ser usados e como eles se integram ao contexto”. Os usuários interpretam os símbolos que encontram no artefato e no contexto associado para entender o estado de um sistema, as ações que eles podem executar e os resultados dessas ações (Norman, 1986; Norman, 1988). De acordo com as abordagens semióticas, essa estrutura e aparência (i.e., a interface) podem ser percebidas como uma mensagem enviada pelos designers aos usuários usando o computador como canal.

Para criar tal mensagem, contudo, a equipe de design e os usuários precisam alcançar um entendimento comum sobre o que deve ser desenvolvido. De um lado, há as expectativas e demandas dos usuários; de outro, há limitações de orçamento, cronograma, tecnológicas e do próprio contexto de desenvolvimento que restringem as soluções de design possíveis. Designers/engenheiros de usabilidade e os usuários tomam parte em um processo de negociação que, neste trabalho, é percebido como um processo de conversação em que ocorrem muitos atos de comunicação.

Com a evolução da área de IHC, o número de canais entre utilizados nessa conversação (i.e., entre a equipe de design e os usuários) aumentou: designers/engenheiros de usabilidade conversam com a equipe de *marketing*, a equipe de suporte ao cliente atua como intermediária entre desenvolvedores e usuários, consultores externos auxiliam tanto as organizações dos usuários quanto a dos desenvolvedores, etc. Essa questão é amplamente discutida em (Grudin, 1991). O uso dos diferentes canais está relacionado às necessidades de cada ato de comunicação: a natureza da informação desejada, o contexto de cada projeto (incluindo características da equipe, objetivos do produto, etc.). Os testes de usabilidade, por exemplo, vêm sendo usados como um dos principais canais de comunicação entre a equipe de design e os usuários com o objetivo de localizar problemas de usabilidade avaliando o desempenho individual dos usuários.

Mais recentemente, a importância do contexto de trabalho na interação entre usuários e computadores foi amplamente reconhecida. Além de focalizar no desempenho dos usuários ao interagirem isoladamente com o artefato, a equipe de design reconheceu que também é importante analisar como a interface reflete o contexto de trabalho dos usuários e as interações sociais necessárias para que eles realizem suas tarefas. Novas abordagens, como design participativo (Muller, 1997) e contextual (Beyer e Holtzblatt, 1998), foram desenvolvidas, introduzindo canais adicionais à comunicação entre designers/engenheiros de usabilidade e usuários.

Finalmente, as abordagens semióticas explicitaram o papel da interface como um ato de comunicação entre designers e usuários, usando o computador como canal (Andersen, 1997; de Souza, 1993; Nadin, 1988). Os designers estabelecem os limites da comunicação e criam um conjunto de signos que os usuários podem ativar (Andersen, 1997), o que significa que os primeiros são os emissores neste processo de comunicação. A Engenharia Semiótica (de Souza, 1993) considera que a mensagem é um artefato de metacomunicação, já que não apenas os designers se comunicam com os usuários, mas a própria interface troca mensagens com os últimos.

Obter informações relevantes para suportar as decisões de design envolve mais do que realizar uma série de perguntas para os prováveis usuários (Gould e Lewis, 1985). A equipe de design precisa fazer perguntas, representar as informações obtidas e as soluções que propõe de uma maneira compreensível para apresentá-las aos usuários e analisar suas reações, etc. (Norman, 1986; Norman, 1988; Rheinfrank *et al.*, 1992). Esse diálogo corresponde, na verdade, às sucessivas iterações do ciclo design-implementação-avaliação (Capítulo 2) que caracteriza o processo de design.

Por outro lado, o processo de design se insere no processo de desenvolvimento e o que é efetivamente disponibilizado para os usuários depende da atuação de cada agente nesses processos.

Na perspectiva deste trabalho, para se entender a as dimensões envolvidas na construção da interface como mensagem, é necessário alcançar uma melhor compreensão sobre a conversação entre várias partes envolvidas no processo de design, como e quando ela começa e prossegue durante o processo de design/desenvolvimento. Assim, considerando o design como um processo de comunicação, o presente trabalho propõe um modelo que organiza um espaço de análise do design.

Neste capítulo apresentamos uma breve revisão sobre alguns modelos de comunicação, o modelo fractal proposto, sua projeção para o processo de design e as discussões finais.

4.2 Uma breve revisão sobre alguns modelos de comunicação

Para entender a comunicação no processo de design é preciso formar um entendimento coerente sobre o próprio conceito de comunicação e o que ele envolve.

A seguir, serão examinados alguns dos modelos clássicos apresentados na literatura para o conceito de comunicação.

Por muito tempo, o modelo dominante na teoria de comunicação foi a Teoria Matemática da Comunicação, de Shannon & Weaver (1949). Ele está fundamentado na Teoria da Informação como um sistema matemático para estudar problemas de transmissão de mensagens através de canais físicos e é representado graficamente pelo diagrama da Figura 4.1 cuja leitura é a seguinte (Coelho Neto, 1996):

- uma fonte de informação produz mensagens;
- um transmissor codifica a mensagem, transformando-a em signos;
- um canal físico transporta os signos
- um receptor decodifica os signos para recompor a mensagem;
- um destinatário recebe a mensagem;
- a transmissão pode sofrer alterações provocadas pelo canal (ruído físico) ou distorções de significado da mensagem (ruídos semânticos), provocadas tanto no processo de codificação (tradução de uma intenção para a forma) quanto no processo de decodificação (o significado atribuído pelo receptor é diferente do significado intencionado pela fonte).

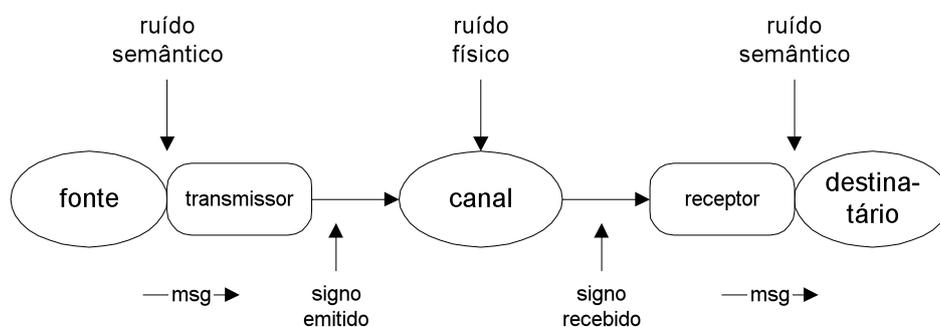


Figura 4.1: Modelo de comunicação de Shannon & Weaver

Apesar de estar fundamentado na Teoria da Informação, esse modelo já foi usado para descrever a comunicação humana. De maneira simplista, pode-se dizer que o emissor tem um pensamento, o codifica em palavras e o transmite pelo ar através de ondas sonoras que atingem o receptor, que decodifica as palavras obtendo o pensamento do emissor. Essa modelagem acrescenta pouco ao entendimento sobre

a comunicação, pois a apresenta como consistindo, basicamente, de emissores trocando pensamentos (Brown, 1996).

Westley e MacLehan (1957) propuseram um modelo derivado da comunicação de massa (Coelho Neto, 1996) que reflete as várias interações entre as entidades envolvidas na comunicação, representado graficamente pela Figura 4.2:

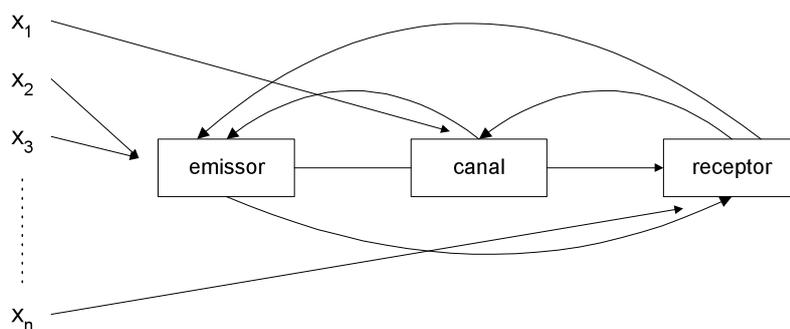


Figura 4.2: Modelo de comunicação de Westley e MacLehan.

- x_1, x_2, \dots, x_n são as mensagens disponíveis;
- o emissor, pessoa ou sistema social, seleciona e transmite intencionalmente a mensagem;
- o receptor, pessoa ou sistema social, pede e usa informações para atender a suas necessidades e solucionar seus problemas;
- o canal é um agente a serviço do receptor, destinado a selecionar e transmitir informações fora do alcance imediato de receptor;
- ocorrem realimentações do receptor para o canal, do receptor para o emissor e do canal para o emissor;
- algumas mensagens podem ser transmitidas diretamente pelo canal;
- as mensagens obtidas pelo receptor se enquadram em uma das quatro categorias a seguir:
 - mensagens captadas diretamente pelo receptor;
 - mensagens transmitidas ao receptor através de uma troca direta com o emissor;
 - mensagens transmitidas ao receptor pelo canal;

- mensagens transmitidas ao receptor pelo emissor através do canal.

Em relação ao anterior (Shannon & Weaver), esse modelo atribui a iniciativa da comunicação ao receptor, que estimula a fonte ao necessitar de uma informação, e acrescenta pontos de realimentação, evidenciando que pode haver uma reação do receptor ao receber uma mensagem, o que melhor se aproxima do processo de comunicação. Alguns pontos de realimentação referem-se a trocas diretas entre emissor e receptor.

Entretanto, todo ato de comunicação é mediado por um canal e, no escopo deste trabalho, é importante entender quais são esses canais (envolvidos na realimentação), pois são eles que viabilizam a conversação entre emissores e receptores.

O modelo de Jakobson (1991), ao contrário dos anteriores, trata do processo lingüístico, de cada ato de comunicação verbal. De acordo com esse modelo, “o **remetente** envia uma **mensagem** ao **destinatário**. Para ser eficaz, a mensagem requer um **contexto** a que se refere (...) apreensível pelo destinatário, e que seja verbal ou suscetível de verbalização; um **código**, total ou parcialmente comum ao remetente e ao destinatário (...) e, finalmente, um **contato**, um canal físico e uma conexão psicológica entre o **remetente** e o **destinatário**, que os capacite a ambos a entrarem e permanecerem em comunicação”.(Jakobson, 1991, pp. 123). O modelo está representado graficamente pela Figura 4.3.



Figura 4.3: Modelo de comunicação proposto por Jakobson.

O autor aponta que o modelo tradicional da linguagem confinava-se, basicamente, ao remetente ou primeira pessoa; ao destinatário, ou segunda pessoa; e a terceira pessoa propriamente dita, alguém ou algo de quem se fala. Ele analisa as condições necessárias para que a comunicação seja eficaz e acrescenta uma série de fatores importantes para a comunicação, como contexto, código e contato. Cada um dos seis fatores descritos determina uma diferente função da linguagem. Essas funções evidenciam um relacionamento entre remetente e destinatário, no sentido

que o predomínio de uma função sobre as outras está relacionado à intenção do remetente de manifestar seu pensamento e provocar uma determinada reação no destinatário. Apenas para exemplificar, a função emotiva, centrada no remetente, “visa a uma expressão direta da atitude de quem fala em relação àquilo que está falando. Tende a suscitar a impressão de uma certa emoção” (Jakobson, 1991, p. 124).

Esse modelo é bastante relevante, pois aponta uma série de aspectos que devem ser verificados quando da análise do espaço de comunicação para torná-la mais eficaz. Principalmente, ele aponta elementos que precisam ser compartilhados por remetente e destinatário. Entretanto, não chega a explicitar a importância da reação do destinatário como evidência de sucesso ou insucesso na comunicação.

Gillian Brown (1995) explora a conduta dos destinatários, ou ouvintes, na terminologia por ela utilizada, ao participarem em um diálogo em que se troca informações. Ela aponta que, “o que os destinatários entenderam sobre o que o remetente acabou de dizer é freqüentemente revelado pelo que os próprios destinatários dizem quando assumem seu turno como remetentes” (p. 1). Essa percepção representa o processo de negociação entre remetente e destinatário em busca de um significado “comum”.

Essa mesma perspectiva aparece na Teoria do Ato da Fala⁶ (Searle, 1969) apresentada em (Liu, 2000). Essa teoria classifica os atos do discurso segundo quatro categorias: atos locucionários, ilocucionários, proposicionais e perlocucionários. Um ato locucionário consiste em emitir verbalmente uma expressão. Um ato ilocucionário é a unidade básica, com significação, da comunicação humana e consiste em um conteúdo proposicional, carregando intenções para serem percebidas pelo ouvinte. Um ato proposicional expressa um conteúdo proposicional que é percebido ao se pronunciar uma expressão. O ato perlocucionário é relativo aos efeitos produzidos no ouvinte por ato ilocucionário.

Para o contexto deste trabalho, o aspecto fundamental deste modelo é que, durante o curso da conversação, o ato ilocucionário é uma unidade completa de comunicação produzida pelo locutor com uma certa significação e intenções incorporadas à expressão. O ouvinte recebe essa unidade de comunicação e interpreta o significado segundo sua própria perspectiva, que pode ser diferente do significado intencionado pelo locutor. Esse ato de comunicação pode (ou não) provocar efeitos no ouvinte. É interessante notar que o próprio locutor é um ouvinte enquanto emite uma expressão, o que causa nele mesmo um efeito perlocucionário.

⁶ Speech Act Theory, desenvolvida por John Searle em 1969.

A Figura 4.4 ilustra como os dois agentes envolvidos no processo de comunicação se alternam nos papéis de locutor e ouvinte.

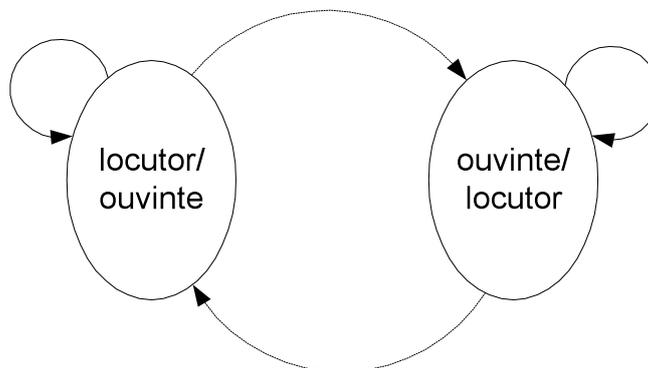


Figura 4.4: Locutor e ouvinte se alternam em turnos durante a comunicação. Linhas pontilhadas, ato perlocucionário; linhas sólidas, ato ilocucionário. Extraído de (Liu, 2000), p. 84.

O modelo fractal de comunicação (Salles *et al.*, 2000a, 2000b), apresentado na seção seguinte, captura a natureza da comunicação no processo de design, em que o principal objetivo é a criação de interface, aqui nomeada mensagem unidade, e enfatiza o fato de que esse processo envolve a troca de várias outras mensagens fracionadas que precisam ser cuidadosamente criadas e veiculadas através de canais adequados.

4.3 O modelo fractal de comunicação

O modelo fractal de comunicação (Salles *et al.*, 2000a, 2000b) está representado graficamente no diagrama da Figura 4.5.

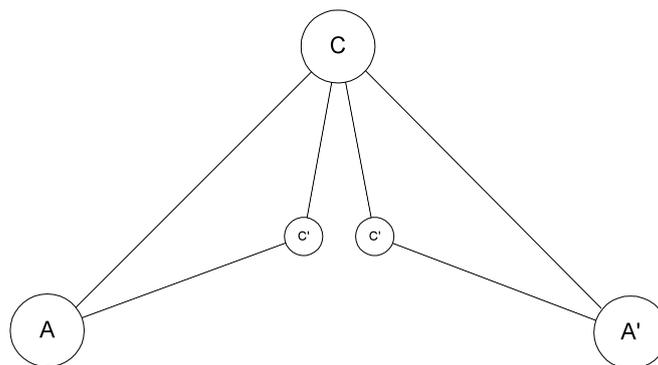


Figura 4.5: O modelo fractal de comunicação.

Nesse modelo, A e A' são agentes que enviam e recebem mensagens e C e C' são canais. A dinâmica da comunicação segundo o modelo envolve os seguintes aspectos:

- no nível mais macroscópico do fractal (nível 1 – representado pelas linhas contínuas), A e A' são os **agentes comunicantes** que se alternam nos papéis de **comunicantes** (aqueles que produzem e enviam a **mensagem**) e **audiência** (aqueles que recebem e potencialmente “consomem” a **mensagem**), utilizando C como **canal**;
- o nível mais interno (linhas tracejadas) representa a potencial conversação entre C e A utilizando C' como **canal**. C, que atua como canal no nível mais macroscópico, assume, neste nível, o papel de agente comunicante se alternando com A nos papéis de **comunicantes** e **audiência**. O mesmo ocorre entre C e A';
- cada turno da comunicação pode implicar a utilização de um **canal** específico. O conceito de **canal** adotado neste trabalho não se refere apenas ao meio físico que veicula uma mensagem, em função da recursividade do modelo. Por exemplo, um livro veicula uma mensagem impressa em papel. Em outra perspectiva, pode-se dizer que livros são uma categoria através da qual se veicula certas mensagens e se cria uma certa **conexão psicológica** com determinada **audiência**. No modelo, essa categoria é percebida como canal e que cada livro específico (uma instância dessa categoria) é uma mensagem. Pode-se analisar aspectos mais detalhados, pois ao saber se um

determinado título refere-se a um romance ou uma comédia, o leitor pode antecipar aspectos da mensagem (a categoria já carrega parte da mensagem). No escopo deste trabalho, **canal** é aquilo capaz de veicular um conteúdo. Deve-se ressaltar, ainda, que um ato de comunicação pode envolver a utilização simultânea de vários canais. Uma conversa entre duas pessoas, por exemplo, pode envolver, além da fala, gestual, expressões faciais, etc. Um professor utiliza simultaneamente vários canais em uma aula como lousa, fala, outros materiais audiovisuais. A própria Internet envolve uma multiplicidade de canais como texto, voz, imagem, vídeo [Press, 1995]. O uso conjunto desses canais tem um grande impacto na maneira como a **audiência** interpreta a mensagem uma vez que eles potencializam uma nova forma de comunicar. No modelo fractal, dependendo no nível da observação, eles serão percebidos como um único canal (ex. Internet) ou como um conjunto de todos os elementos envolvidos.

Na perspectiva deste trabalho, para que a comunicação seja bem sucedida, é preciso garantir que o **canal** seja adequado, ou seja, que propague a **mensagem** do **comunicante** para a **audiência** e que seja apropriado ao contexto da comunicação, o que estimula o estabelecimento de uma **conexão psicológica** entre os agentes. Ao escrever o livro “O meio é a mensagem”, McLuhan (1964) estava interessado não no conteúdo veiculado por de um canal, mas nos efeitos físicos e sociais intrínsecos à maneira como ele estende os nossos sentidos. A escolha que fazemos por determinados canais reflete nosso contexto, mas igualmente o altera. Por exemplo, em uma empresa grande, o uso de ferramentas eletrônicas para comunicação e trabalho em equipe torna o processo mais eficiente sob vários aspectos. Ao propor uma reunião as pessoas podem ver a disponibilidade na agenda de outras. O correio eletrônico poderia tornar a comunicação mais ágil, diminuindo o tempo de resposta. Contudo, o volume de informação veiculado pode se tornar muito grande, maior do que o indivíduo é capaz de administrar com as ferramentas que possui, exigindo um tempo e esforço considerável;

Um outro aspecto que deve ser observado é o **contexto** a que se refere a **mensagem**, que deve ser apreensível pela **audiência**. Cabe observar que, em numerosos casos, o **contexto** (alguém ou algo de que se fala) é o aspecto dominante na mensagem. O **agente comunicante** deve estar atento e sensível a isso quando da elaboração da mensagem. Por exemplo, apesar de uma organização como um todo possuir um conjunto claro de objetivos e interesses, os indivíduos e grupos que

integram a organização podem não compartilhar tais objetivos e interesses da mesma forma. Em uma organização de criação de software, as equipes de design e desenvolvimento podem diferir substancialmente quanto à formação e práticas de trabalho. Entretanto, o produto final é resultado da interação e comunicação entre essas e outras equipes. A equipe de design focaliza problemas de usabilidade e seu impacto na interação entre usuários e o artefato. Desenvolvedores focalizam a implementação de funções. Ao comunicarem um problema de usabilidade aos desenvolvedores a fim de esses que alterem o código para corrigi-lo, designers/engenheiros de usabilidade precisam expressá-lo de uma forma eficiente para que o contexto seja apreensível para a audiência (ex., usuários esperam que a função se comporte da maneira X e a lógica implementada é Y).

Um outro aspecto importante para identificar se a comunicação foi bem sucedida é a **alternância dos agentes como comunicantes e audiência**. Se os comunicantes não têm um turno como audiência, eles não podem identificar se a mensagem foi recebida com sucesso.

O modelo proposto reflete uma conversação em que os **agentes comunicantes** negociam um significado, se alternando no envio e recebimento de mensagens. Nessa perspectiva, C atua como canal, mas em um nível mais interno, ele assume a posição de **agente comunicante**, recebendo e enviando mensagens ele próprio. Por exemplo, em uma situação em que um apresentador entrevista uma pessoa e lhe faz uma pergunta enviada por um espectador, o primeiro atua como **canal** entre o entrevistado e o espectador (nível 1 do fractal). Entretanto, o apresentador pode, após ouvir a resposta do entrevistado, comentá-la ou dirigir-lhe nova pergunta, estabelecendo uma conversação em que atua como **agente comunicante** (nível 2 do fractal). Desta forma, o modelo fractal viabiliza uma representação flexível de um processo comunicativo, de acordo com os objetivos da análise.

Enquanto o modelo Shannon & Weaver focaliza a transmissão de mensagens, o modelo fractal focaliza a conversação entre agentes. Por isso, o primeiro apresenta a comunicação de maneira unidirecional, já que seu interesse primário é a eficácia da transmissão e não propriamente o estímulo provocado na audiência.

O modelo Westley & McLehan focaliza a comunicação em massa e, assim, enfatiza o canal utilizado para transmitir mensagens para uma grande audiência. A reação da audiência é um aspecto importante e os autores introduzem pontos de realimentação. Contudo, eles não consideram os canais utilizados nesses atos comunicativos (ex., TV pode ser o canal principal e pesquisa de opinião o canal

usado na realimentação). Na perspectiva do modelo fractal, o uso e seleção desses canais são aspectos tão importantes quanto a própria utilização do canal principal.

Gillian Brown e Jakobson trazem inúmeras contribuições para o entendimento da comunicação verbal que podem ser estendidas para outros processos comunicativos. O modelo fractal, contudo, não se restringe à comunicação verbal e enfatiza a influência dos canais nos processos comunicativos.

Considerando o modelo proposto, o sucesso da comunicação é determinado pelo sucesso de cada ato comunicativo, da elaboração de cada mensagem, inclusive nos níveis mais internos do fractal.

4.4 Projetando a conversação do processo de design de interfaces no modelo fractal

Nesta seção, o processo de design de interfaces é projetado sobre o modelo fractal a fim de exemplificar como ele suporta a representação da estrutura da conversação inerente a tal processo.

Dependendo do contexto de desenvolvimento e das características de cada projeto, o processo de criação do produto pode variar, mas, tipicamente, apresenta sucessivas iterações do ciclo design-implementação-avaliação com a incorporação dos resultados das avaliações no ciclo seguinte. A Figura 4.6 ilustra um possível processo de design centrado no conceito de usabilidade, que será usado como exemplo.

Partindo da visão das abordagens semióticas, de que a interface é uma mensagem enviada pelos designers aos usuários utilizando o computador como canal, o processo de design passa a ser entendido como um processo de conversação para elaboração desta mensagem. O primeiro aspecto a ser observado é que a equipe de design e os usuários se alternam em turnos como **comunicantes e audiência**,

No processo apresentado na Figura 4.6, diferentes métodos de IHC são aplicados com a intenção de coletar informações sobre os usuários, suas tarefas e a adequação das estratégias de interação à comunidade de usuários. Tais informações serão usadas para criar e melhorar a mensagem primária (interface).

Inicialmente, uma comunidade de usuários tem uma demanda por um sistema computacional e é ela que dispara todo o processo de desenvolvimento. Isso corresponde ao primeiro turno da comunicação, em que os usuários são os agentes comunicantes, e, ao receber a mensagem, a equipe de criação forma a primeira idéia

sobre o produto. No modelo fractal, esse ato de comunicação corresponde a uma mensagem enviada por U (usuários) para ED (equipe de design).

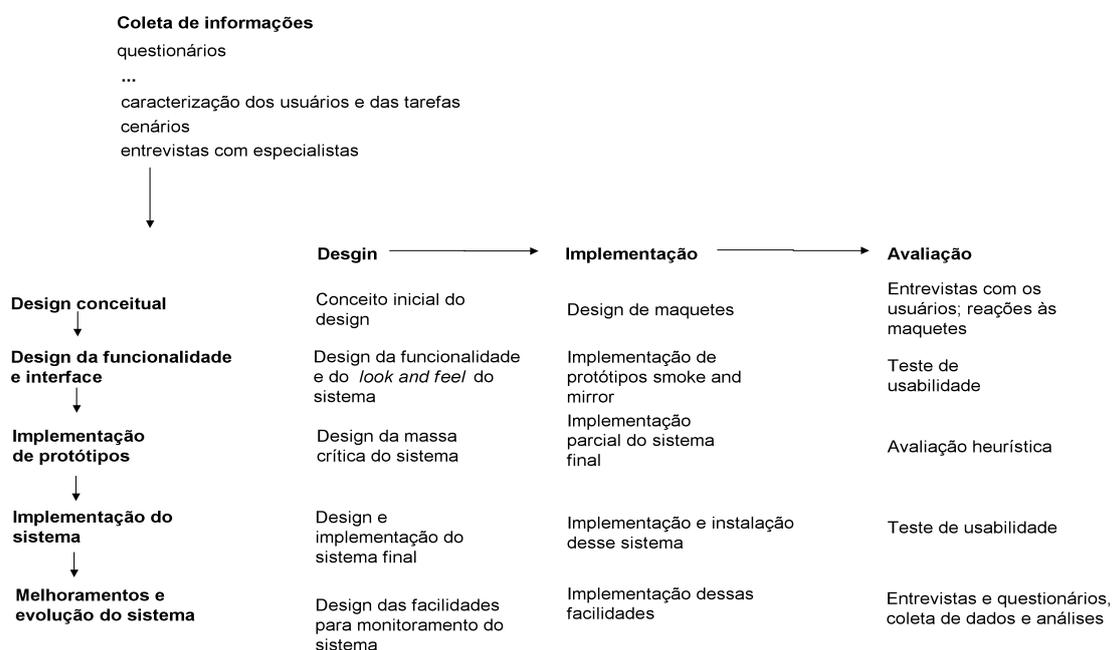


Figura 4.6: Um exemplo de processo de design iterativo com sucessivas fases do ciclo design-implementação-avaliação (extraído da Introdução do Capítulo 2 de [Baecker et al., 1995], p. 74).

A seguir, inicia-se uma fase de coleta de informações. Em um diálogo, em função do entendimento que o comunicante tem sobre a audiência, ele pode alterar a forma de uma mensagem ou adicionar conteúdo para facilitar o entendimento. Existem situações extremas. Em uma aula, por exemplo, em que o comunicante (professor) tem conhecimento profundo do domínio sobre o qual se formula a mensagem, ele seleciona o conteúdo adequado de sua própria base de conhecimento e determina uma forma conveniente para que a audiência consiga interpretar a mensagem. No caso de design, a designers e engenheiros de usabilidade não têm, necessariamente, conhecimento sobre o domínio da aplicação. Eles precisam conhecer características dos usuários e das tarefas que eles executam para determinar uma forma e conteúdo apropriados. Por outro lado, a equipe conhece o domínio computacional, que impõe limites nas soluções possíveis e até mesmo pode transformar os processos de trabalho dos usuários. No processo de design da Figura 4.6 essa fase corresponde à coleta de informações e os métodos utilizados são

aplicação de questionários, caracterização dos usuários e das tarefas, uso de cenários, etc.

Em seguida, conforme o processo da Figura 4.6, a equipe de design propõe um design conceitual inicial e o apresenta aos usuários em forma de maquetes. As reações destes ao design proposto são observadas e eles são entrevistados.

Segue-se a fase de design da funcionalidade e interface do sistema. As reações dos usuários aos protótipos produzidos são verificadas através de testes de usabilidade.

Todas as atividades do processo enumeradas anteriormente correspondem a troca de mensagens entre usuários e a equipe de design, que se alternam em turnos como **comunicantes** e **audiência**. Diferentes canais foram utilizados, de acordo com o tipo de informação que se deseja coletar e da fase do processo (ex., questionários na fase de coleta de informações, maquetes na fase de design conceitual e testes de usabilidade durante a fase de design da funcionalidade). A Figura 4.7 mostra essa comunicação no nível 1 do fractal.

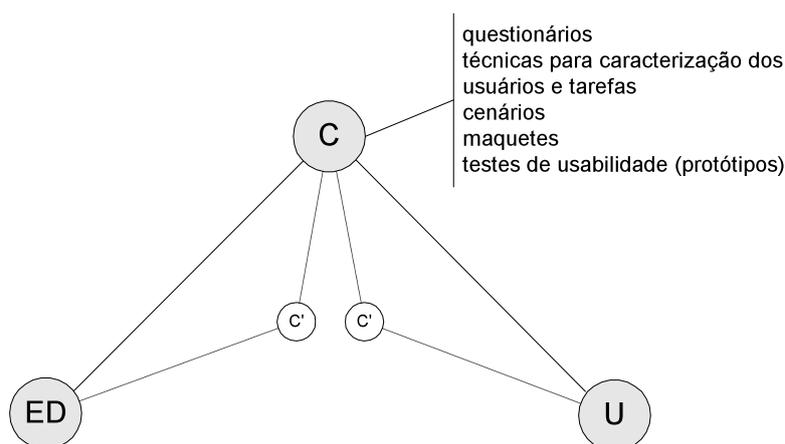


Figura 4.7: Conversação entre a equipe de design (ED) e usuários (U) através de diferentes canais.

Os resultados dos testes realimentam a fase seguinte, de implementação de protótipos. Nesta fase, a avaliação é feita através de uma avaliação heurística, um método de inspeção em que os próprios designers verificam a interface. Winograd (1996) argumenta que “o design não é tanto um processo de planejamento e execução cuidadosos, mas é uma conversa, na qual o parceiro da conversa – o próprio objeto projetado – pode gerar interrupções e contribuições inesperadas. O designer escuta o design emergente, ao mesmo tempo em que define sua forma” (p. xxi). Esta conversação entre o designer e a interface emergente implica caminhar um

nível mais interno no modelo fractal e esses agentes assumem os papéis de **comunicantes** e **audiência** e o método de inspeção é o canal utilizado (Figura 4.8)

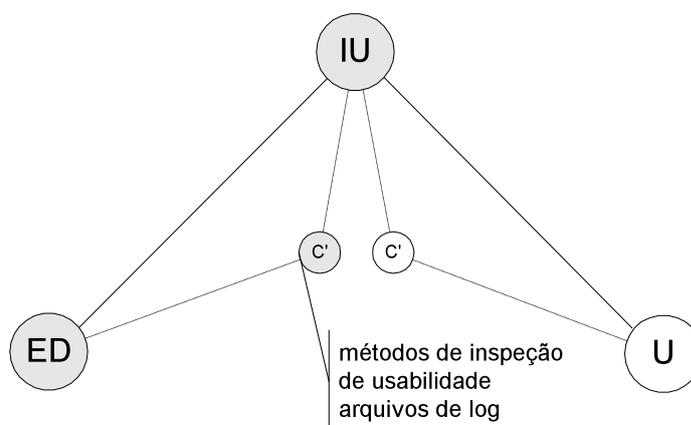


Figura 4.8: Comunicação dos designers com a interface emergente.

Na fase de implementação do sistema final, as reações dos usuários ao design são novamente verificadas através de testes de usabilidade.

Na fase seguinte, de melhoramentos e evolução do sistema, questionários e entrevistas são utilizados para avaliar as impressões dos usuários. Novamente, os métodos citados são canais utilizados na comunicação entre a equipe de design e os usuários (Figura 4.7). Essa fase inclui ainda a monitoração do uso através de arquivos de *log* que atuam como um canal entre a equipe de design e o próprio sistema (Figura 4.8).

A interação entre os usuários e o sistema representa também caminhamento para um nível mais interno no fractal (Figura 4.9).

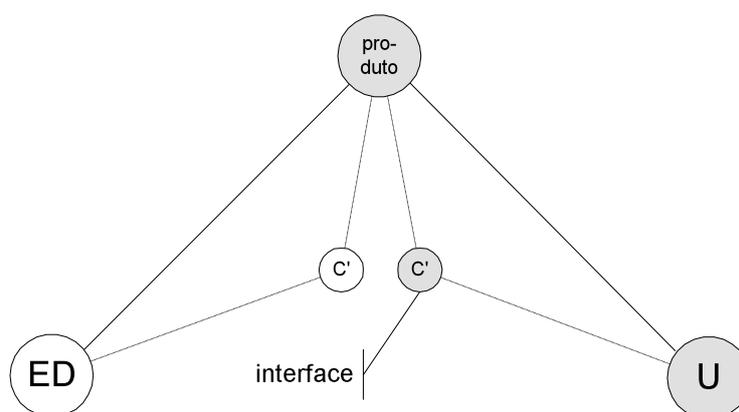


Figura 4.9: Interação usuários-sistema

Assim, o modelo apresentado acomoda as fases e iteratividade do processo de design, ou seja, o processo de conversação em que a equipe de design e os usuários negociam o que será disponibilizado.

O nível fractal foi explorado na conversação entre designers e o artefato emergente através de arquivos de *log*, inspeções e da própria interação entre usuários e o artefato.

Cabe ressaltar que, ao elaborar um questionário, por exemplo, os designers estão elaborando uma pergunta para os usuários (i.e., os designers atuam como audiência neste momento). Isso envolve a determinação de um conteúdo e forma adequados para apresentar tal pergunta, a realização de testes piloto, a modificação do questionário em função dos resultados do piloto, etc. Nielsen (1995) enfatiza a necessidade de utilizar metamétodos (Capítulo 2) para complementar a aplicação de métodos. Ele ressalta a importância de se elaborar um plano trabalho para a avaliação, submetê-lo a uma revisão por especialistas e fazer um teste piloto para minimizar dificuldades de interpretação dos usuários que participam do teste. Assim, não apenas a interface é objeto de design e implementação, mas em um nível mais elementar, cada elemento usado na avaliação (ex., questionários, testes de usabilidade) é também objeto de design e implementação.

Isso revela a natureza fractal [Mandelbrot, 1983] do processo de design. Assim, o modelo proposto acomoda não apenas a iteração do processo de design e as interações entre as partes envolvidas, mas também os vários processos de comunicação que ocorrem durante o design.

4.5 Considerações finais

Um aspecto que deve ser ressaltado é que, no processo de design da interface, os designers devem também trabalhar no design de mensagens intermediárias e escolher os canais apropriados para veicula-las. Daí a natureza fractal do design. Através da avaliação (i.e., da troca de mensagens intermediárias) o design da interface é modificado e aprimorado gradativamente.

O design das mensagens intermediárias depende do contexto de desenvolvimento (Grudin, 1991) e das características de cada projeto, já que esses fatores limitam as interações com os usuários, a escolha de canais, etc.

O modelo representado pela Figura 4.5 acomoda a iteração própria do design, os métodos e técnicas de usabilidade correntes, enfatiza a cooperação entre designers e usuários (mudando a perspectiva do diálogo entre essas partes, já que ambos são

tanto comunicantes quanto audiência) e enfatiza o design de mensagens intermediárias no processo de design da interface. Nesta abordagem, o design de interface pode ser visto como um processo de semiose coletiva, em que usuários e designers cooperam para a criação de um novo sistema de signos (a interface).

Conforme discussão apresentada no Capítulo 2, os processos de design e de desenvolvimento necessitam de maior integração. Analisando-os segundo a perspectiva da comunicação proposta pelo modelo fractal, pode-se observar que a equipe de design atua como canal de comunicação entre os usuários e o restante da equipe (Figura 4.10 (a) - nível 1 do fractal). Nessa perspectiva, a equipe de criação do produto⁷ e os usuários se alternam nos papéis de comunicantes e audiência.

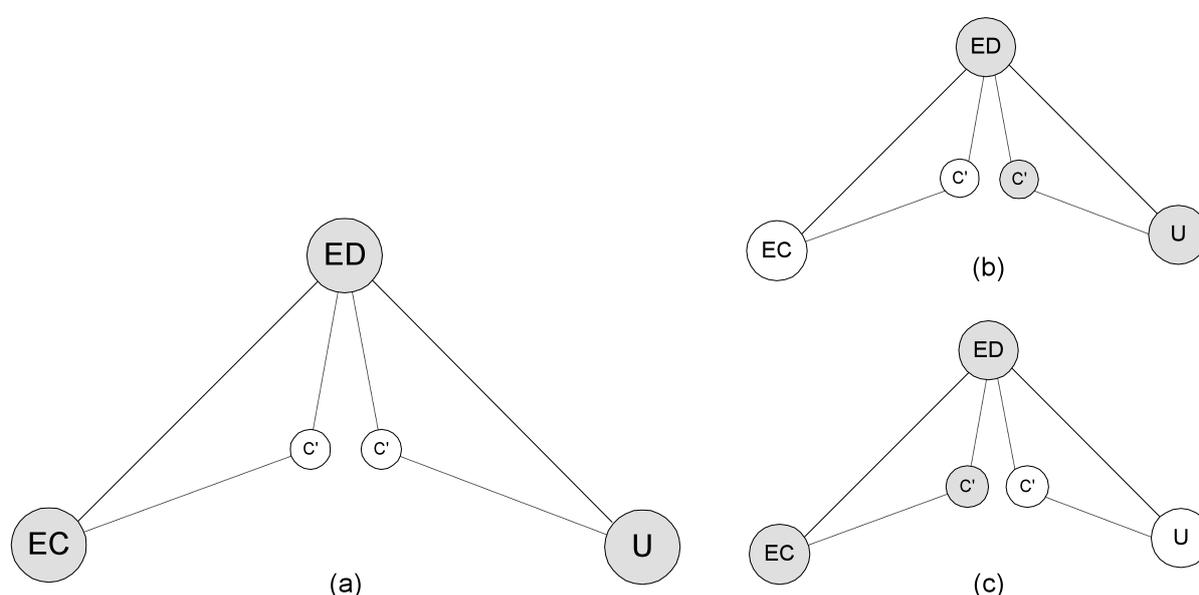


Figura 4.10: A comunicação no processo de design/desenvolvimento. (a) Nível 1 - comunicação entre a equipe de criação do produto e usuários; (b) nível 2 – equipe de design e usuários; (c) nível 2 – equipe de design e toda a equipe de criação do produto.

Para que a comunicação ocorra nesse nível, contudo, designers e engenheiros de usabilidade precisam estabelecer conversações com os usuários e com toda a equipe de desenvolvimento (Figura 4.10 (b) e (c), nível 2 do fractal). É importante notar que cada ato de comunicação no nível 2 está inserido em um contexto mais amplo, ou seja, refletem os turnos de comunicantes e audiência entre a equipe e os usuários, uma vez que:

⁷ O termo “equipe de criação do produto” refere-se a todos os agentes do processo envolvidos na criação e desenvolvimento do produto

- a comunicação entre designers/engenheiros de usabilidade e usuários reflete, de certa forma, aspectos importantes para a equipe de criação, já que o objetivo dessa comunicação é entender a relação dos usuários com o produto. Por exemplo, no caso de novas versões, a própria existência de um produto direciona a conversação com os usuários. Se a equipe precisa entender como a funcionalidade oferecida atende as necessidades desses, engenheiros podem fazer visitas *in loco* para identificar os padrões de uso (o que se “pergunta” aos usuários é direcionado pela demanda da equipe). Assim, as mensagens trocadas entre designers/engenheiros e usuários veiculam informações relevantes para a equipe como um todo. Isso corresponde ao turno da equipe como comunicante e dos usuários como audiência;
- designers/engenheiros de usabilidade precisam propagar os resultados de sua conversação com os usuários para equipe de criação do produto. Seguindo o exemplo anterior, isso envolveria, por exemplo, comunicar à equipe sobre os padrões de uso identificados em visitas *in loco* não contemplados pela funcionalidade corrente, com que frequência eles ocorrem, etc. A equipe então assume seu turno como audiência e os usuários como comunicantes. Para que essa propagação resulte em melhorias no produto, designers e engenheiros precisam utilizar canais e mensagens adequados ao contexto do projeto e atingir agentes chave no processo.

Analisar a comunicação através dessa perspectiva (nível 1 do fractal) é importante, pois a qualidade do produto emergente é resultado da qualidade da conversação entre a equipe e os usuários. Se o processo de design não estiver integrado ao processo de desenvolvimento torna-se complexo implementar melhorias no produto, do ponto de vista da usabilidade.

É necessário focalizar os seguintes elementos:

- a comunicação entre a equipe de design e os usuários precisa refletir o contexto do projeto, o que implica a identificação de canais e mensagens que sejam relevantes e oportunos para um dado projeto;
- a informação coletada sobre os usuários precisa ser propagada para a equipe de maneira adequada, potencializando melhorias no produto. Isso implica identificar canais e mensagens adequados para que designers/engenheiros de usabilidade possam comunicar-se com outros agentes do processo relevantes para a criação do produto.

É importante ressaltar que o conceito de canal adotado pelo modelo não se refere apenas ao meio físico que veicula a mensagem, em função da recursividade do modelo. Por exemplo, uma especificação funcional é percebida, do ponto de vista mais macroscópico, como uma mensagem que comunica aos desenvolvedores o que deve ser implementado, as prioridades, etc., e o canal pode ser papel. Em um nível mais interno, a especificação é percebida como um modelo de documento apropriado para veicular determinado conteúdo. Uma instância particular da especificação é uma mensagem, elaborada para o contexto do projeto.

O mesmo pode ser dito à respeito da interface. Ela é uma mensagem enviada pelos designers aos usuários veiculada através do computador. Por outro lado, em um teste de usabilidade, a interface pode ser vista como um dos canais utilizados e a mensagem trocada entre engenheiros de usabilidade e usuários refere-se aos problemas de usabilidade identificados.

Uma vez que estaremos avaliando diferentes níveis de comunicação no modelo, o mesmo agente pode assumir o papel de canal ou de mensagem, de acordo com o nível analisado.

Capítulo 5

Percurso comunicativo: um método de inspeção da comunicação no processo de design

Conforme discussão apresentada no Capítulo 2, os processos de design e de desenvolvimento necessitam de maior integração. Por um lado, os processos de design são centrados nos usuários e baseiam-se em avaliar empiricamente a qualidade da interação e proceder ao redesign considerando os resultados da avaliação. Para que essa iteratividade do processo seja factível, as modificações propostas devem ser implementadas até o próximo ciclo iterativo. Por outro lado, o processo de desenvolvimento é centrado na produção de software.

Assim, a comunicação entre designers/engenheiros de usabilidade e o resto da equipe é um aspecto crítico para que os primeiros possam participar mais efetivamente do processo e causar maior impacto na usabilidade do produto. Eles precisam, ao mesmo tempo, entender as limitações do projeto e comunicar sua perspectiva com eficiência.

Este capítulo apresenta um método de inspeção que auxilia a identificar deficiências na conversação entre designers e engenheiros de usabilidade com a equipe do produto e com os usuários.

5.1 Introdução

O **percurso comunicativo** é um método de inspeção do processo de design (e de sua integração com o processo de desenvolvimento) que focaliza os atos de

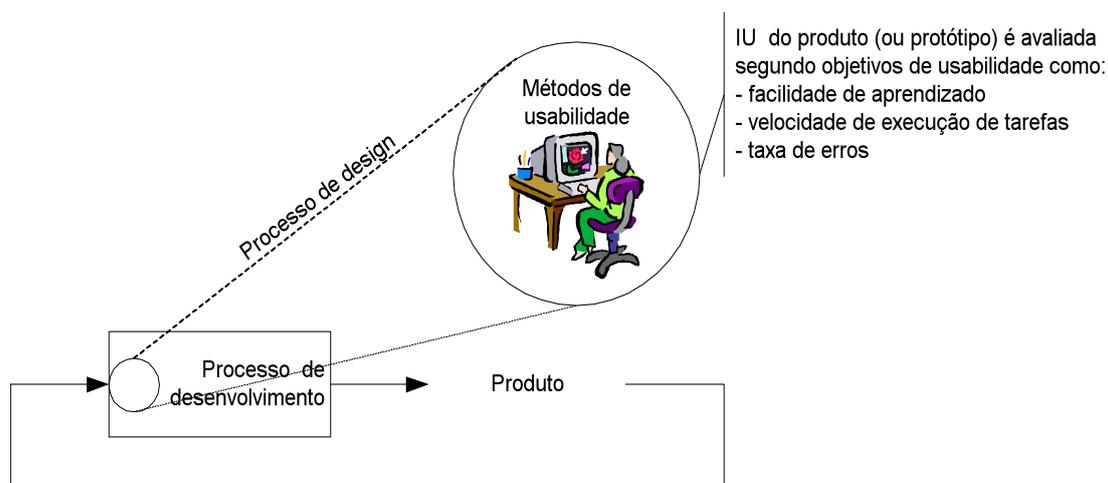
comunicação envolvendo designers e engenheiros de usabilidade com o restante da equipe envolvida na criação de um artefato de software⁸.

A literatura de IHC tem apresentado uma série de métodos e técnicas de avaliação de usabilidade de sistemas, distribuídos nas seguintes categorias:

- métodos de inspeção, em que especialistas avaliam a usabilidade do produto;
- testes de usabilidade, envolvendo a participação de usuários;
- outros métodos para medir ou obter *feedback* sobre usabilidade, como estudos de campo, questionários, etc.

Em sua grande maioria, eles focalizam a interface do produto. Engenheiros de usabilidade observam usuários interagindo com o produto ou instância do produto (i.e., protótipos) e avaliam a qualidade desta interação medindo, em geral, atributos relacionados ao desempenho dos usuários, conforme já apresentado na seção 2.2. Os dados obtidos através da observação são usados para modificar a interface produto em uma nova iteração do processo de design. A Figura 5.1 ilustra esse processo.

Figura 5.1: Processo de design baseado na aplicação de métodos de avaliação usabilidade



que focalizam o produto; dados coletados realimentam a iteração seguinte do processo.

O enfoque adotado no método proposto partiu da observação de que, em muitos projetos, o processo de design baseia-se na aplicação desses métodos e técnicas para determinar aspectos de usabilidade sem que sejam suficientemente observados os seguintes aspectos:

⁸ No restante deste trabalho, essa equipe será referenciada como “equipe de criação do produto”.

- um estudo de usabilidade é, assim como a própria interface, objeto de design e emerge de um processo análogo ao processo de design da interface (i.e., iterativo e empírico). Os métodos e técnicas de avaliação de usabilidades são modelos que orientam a equipe de design. Sua aplicação a um contexto específico implica a criação de uma instância particular para o projeto, que pode, inclusive, envolver adaptações dos mesmos. No contexto deste trabalho, tais métodos e técnicas são canais que veiculam uma meta-mensagem. Sua aplicação é equivalente à elaboração de uma mensagem em que os designers perguntam aos usuários algo sobre eles e sua relação com o artefato e/ou contexto de utilização, conforme discutido na seção 4.4. Assim como uma mensagem está relacionada a um contexto que influi na sua interpretação, a aplicação de tais métodos deve também estar fortemente relacionada ao contexto do projeto (i.e., aos objetivos do produto, características da equipe, limitações, etc.). Os métodos, tais como estão propostos na literatura, não evidenciam suficientemente esse aspecto, mas designers e engenheiros de usabilidade devem estar atentos à necessidade de adaptá-los a um contexto específico;
- métodos e técnicas de avaliação de usabilidade enfatizam a comunicação entre designers e usuários (de fato, tais métodos são canais entre esses agentes). Entretanto, essa comunicação só terá impacto no produto final se as informações coletadas se propagarem para a equipe de criação do produto de forma eficaz. Para tanto, assim como precisam identificar os métodos mais eficientes para se comunicarem com os usuários, a equipe de design precisa encontrar canais adequados e elaborar mensagens relevantes para a equipe de criação, considerando, também nesta etapa da comunicação, as características particulares do projeto.

Os aspectos discutidos anteriormente estão relacionados ao processo de design/desenvolvimento, isto é, à forma como tais métodos se estruturam nesses processos. Isto é tão determinante para o produto final quanto a própria qualidade e abrangência de um método de avaliação de usabilidade da interface.

O método de inspeção proposto neste trabalho tem como objeto de estudo o processo de design, ou seja, a comunicação entre usuários e designers e a forma como essa comunicação se insere no processo de desenvolvimento (i.e., a comunicação entre designers e equipe do projeto). Os atos de comunicação envolvendo designers e engenheiros de usabilidade com o restante da equipe são

analisados quanto aos canais usados, mensagens trocadas, propagação das mensagens para outros agentes do processo, etc. A Figura 5.2 ilustra conceitualmente a proposta.

Após esta análise, o engenheiro de usabilidade responsável pela inspeção deve procurar mecanismos para tornar a comunicação no processo mais eficaz, o que pode envolver a criação de novos canais e/ou mensagens.

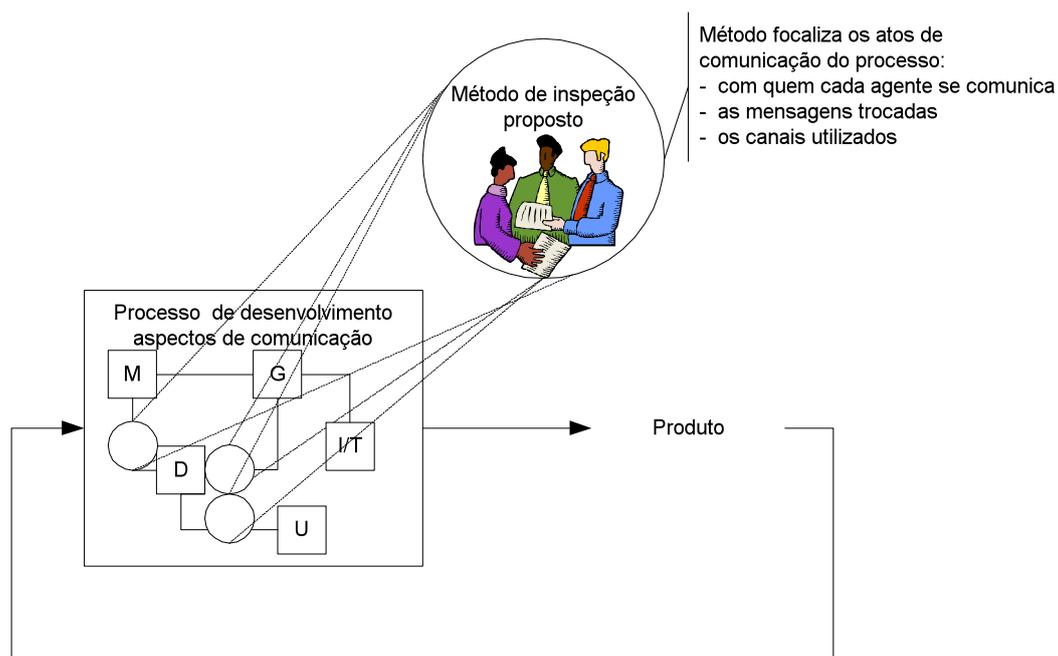


Figura 5.2: O método proposto avalia os atos de comunicação que ocorrem durante o processo de design e desenvolvimento.

5.2 O percurso comunicativo

5.2.1 Visão geral

Tipicamente, no processo de desenvolvimento de software, um percurso é um processo de revisão em que o autor de um aspecto do projeto o apresenta a um grupo de membros da equipe [Yourdon, 1989].

No caso do **percurso comunicativo**, o procedimento de inspeção é análogo ao adotado por outros tipos de percursos, mas o objeto de revisão é o processo de design e a forma como ele se integra ao processo de desenvolvimento quanto aos aspectos de comunicação. Espera-se que, identificando-se deficiências de comunicação envolvendo a equipe de design, seja possível desenvolver mecanismos para torná-la mais eficiente e, conseqüentemente, melhorar a qualidade do produto final quanto à usabilidade.

As entradas para a sessão do percurso incluem um meta-modelo das comunicações incluindo a identificação dos agentes, especialmente os que têm maior impacto no processo de design e desenvolvimento, tipos de mensagens trocadas e canais utilizados; uma descrição dos objetivos, público alvo e comunidade de usuários corrente e, dentro do possível, estatísticas com o número e severidade dos problemas de usabilidade detectados e corrigidos⁹. O meta-modelo é um mapa resultante da aplicação do modelo fractal de comunicação, apresentado no capítulo 4 deste trabalho, ao processo em questão. Um engenheiro de usabilidade prepara as entradas. O grupo então se reúne, percorre o meta-modelo e considera os aspectos de comunicação de todas as conexões. Para cada conexão, os revisores discutem se a comunicação foi ou não efetiva. Em caso negativo, eles apresentam uma justificativa para a deficiência da comunicação. A Tabela 5.1 apresenta um resumo do método.

⁹ No caso de versões de um produto, estatísticas das versões anteriores são úteis para acompanhar o progresso do processo.

<ol style="list-style-type: none"> 1. Preparação das entradas Construção do meta-modelo com os atos de comunicação do processo que envolvem designers e engenheiros de usabilidade de acordo com os passos seguintes: Passo 1: identificação dos agentes do processo Passo 2: identificação dos canais e mensagens trocadas; conexão dos agentes através dos canais, observando-se que alguns agentes atuam como canais Descrição dos objetivos, público alvo e comunidade de usuários corrente Estatísticas com o número de problemas de usabilidade detectados e corrigidos na versão (ou iteração) anterior. 2. Identificação e convocação dos revisores 3. Caminhada pelo meta-modelo analisando os atos de comunicação quanto à: <ol style="list-style-type: none"> a) Adequação e suficiência dos canais e mensagens b) Propagação das mensagens 4. Registro da revisão 5. Alterações no processo para minimizar as deficiências de comunicação encontradas
--

Tabela 5.1: Visão geral do percurso comunicativo

5.2.2 A formação do grupo de revisão

O percurso comunicativo pode ser realizado individualmente ou em grupo. No caso da inspeção em grupo, a participação de alguns agentes é recomendável, já que a etapa posterior à análise implica alterações no processo que podem ter impacto nas atividades dos mesmos. Nesta seção, discuti-se a formação do grupo e as responsabilidades dos diferentes elementos, antes, durante e após a sessão de inspeção. A formação discutida é apenas uma sugestão. Espera-se que, para cada organização e, mais especificamente, para cada projeto, características particulares influenciem na escolha dos revisores, em função das diferentes formações de cada equipe. O engenheiro de usabilidade, que organiza a revisão, deve estar atento a essas oportunidades.

Sugere-se a participação dos seguintes agentes no grupo de revisão:

- **O organizador** da revisão deve ser um dos engenheiros de usabilidade do projeto. É de sua responsabilidade preparar o material de entrada para a inspeção, apresentá-lo ao grupo e assegurar que os comentários dos revisores foram entendidos. Talvez a etapa mais importante envolva atividades relacionadas a alterações no processo após a revisão. O organizador deve considerar todas as sugestões, críticas e comentários séria e objetivamente e decidir quais são apropriados ao contexto em questão. A seguir, ele deve

procurar mecanismos para melhorar os aspectos deficientes detectados no processo de revisão (ex., criação de novos canais, identificação da necessidade de outras mensagens, etc). O organizador pode designar alguém do grupo para fazer o registro da discussão ou providenciar algum mecanismo para esse fim (ex., *videotape*);

- a participação de **desenvolvedores** pode ser benéfica, já que são eles que, efetivamente, implementam as alterações no produto na iteração seguinte à avaliação. Deficiências de comunicação entre desenvolvedores e designers/engenheiros de usabilidade tem um forte impacto no processo. Por um lado, eles conhecem com profundidade o que é tecnicamente possível e o custo de cada alteração, considerando especialmente o cronograma do projeto. Por outro lado, IHC é uma área ainda recente nas organizações que criam produtos de software e sua divulgação entre os desenvolvedores pode promover um maior comprometimento da parte deles com aspectos de usabilidade. Isso facilita que as alterações propostas sejam rapidamente implementadas, como requer o processo de design;
- a participação do **gerente do desenvolvimento** é muito importante, já que ele unifica as discussões do projeto, administra cronograma e orçamento, etc;
- no caso de **desenvolvimento interno**, sugere-se a participação do **gerente da seção cliente**, isto é, da seção a que se destina o produto, e de **representantes dos usuários**. Neste caso, objetivos estratégicos da empresa, o contexto de utilização do produto (ex., como a comunidade de usuários do produto e as tarefas que executam se inserem na organização) precisam ser considerados;
- no caso de **desenvolvimento de produtos de prateleira**, recomenda-se a participação de agentes do processo envolvidos com os usuários potenciais (i.e., mercado alvo, clientes):
 - **representante da equipe de marketing**. Várias decisões do projeto são conseqüências dos objetivos de *marketing*. A equipe de marketing coleta dados sobre o público alvo do produto, informações sobre os produtos concorrentes e sobre seus compradores. Os objetivos de *marketing* direcionam vários aspectos do produto e essa equipe desempenha um papel expressivo nas fases iniciais do projeto. Nesse sentido, a comunicação entre o pessoal de *marketing* e

designers/engenheiros de usabilidade é importante, já que as informações coletadas por ambos se complementam;

- **representante do suporte aos clientes/usuários.** A equipe de suporte registra e organiza dados sobre problemas reportados por usuários. Um aspecto fundamental é que estes problemas surgem em um contexto de uso específico, ao contrário dos problemas de usabilidade detectados em laboratório.

A revisão pode também ser feita por um único agente. Várias situações no projeto podem dificultar as revisões em grupo, especialmente se usabilidade ainda não for uma prioridade para a equipe. Nesses casos, o engenheiro de usabilidade deve proceder à análise individualmente. Entretanto, ele deve considerar aspectos como comprometimento e interesse da equipe quando propuser o uso de novos canais e mensagens, já que as diferentes soluções demandam um envolvimento por parte da equipe.

5.2.3 Sobre o registro

Assim como no caso de outros atos de comunicação do processo, é necessário encontrar um canal apropriado para registrar a discussão do grupo.

Recomenda-se fortemente utilizar materiais que todo o grupo possa visualizar. Em particular, é importante que o meta-modelo esteja visível para todos, já que a discussão se baseia em elementos como as conexões entre os agentes, canais e mensagens utilizados e todo o grupo precisa focalizar o mesmo ponto do diagrama. É provável que revisores questionem diversos aspectos do meta-modelo e é produtivo que outras versões do mesmo, ilustrando os pontos discutidos, sejam produzidas durante a revisão (ex., utilizando transparências sobrepostas para incluir elementos).

Além dessas versões do mapa de comunicação, é importante registrar com precisão as justificativas de deficiência da comunicação. É com base nelas que o engenheiro de usabilidade irá propor melhorias no processo. Pode-se designar um dos revisores para fazer o registro eletronicamente, gravar a discussão em *videotape*, o que for mais adequado para o contexto do projeto.

5.2.4 Quando executar um percurso

O objetivo do **percurso comunicativo** é ser uma ferramenta que auxilie na melhoria contínua do processo. Como o contexto envolvendo o projeto muda constantemente (ex., a maturidade da equipe, características do mercado, posição dos produtos competidores, organização da empresa, etc.), as oportunidades e barreiras para melhorias do processo também mudam.

Sugere-se que, para cada versão do produto, aplique-se o percurso pelo menos uma vez, registrando dados para monitorar os benefícios da aplicação do método. Obviamente, o organizador deve evitar as etapas de maior pressão para equipe (ex., proximidade com prazos para entrega de liberações).

5.2.5 Descrição detalhada do percurso comunicativo

O **percurso comunicativo** tem duas fases, uma preparatória e uma de análise. A fase preparatória refere-se à construção de um espaço de análise da comunicação envolvida no processo de design. O organizador constrói um meta-modelo com base no modelo fractal de comunicação apresentado no Capítulo 4, prepara uma descrição dos objetivos, público alvo e comunidade de usuários corrente e, tanto quanto possível, coleta estatísticas com o número e severidade dos problemas de usabilidade detectados e corrigidos na última versão do produto ou iteração do processo. A maior parte do trabalho de análise ocorre na segunda fase, quando os revisores se reúnem e discutem cada conexão do mapa.

A seguir, cada etapa do método é detalhada.

Fase 1

A primeira atividade desta fase é a construção do meta-modelo de comunicação. Ele é um mapa da comunicação envolvida no processo de criação do software e deve ser produzido em dois passos. O primeiro passo é a identificação de todos os agentes do processo de desenvolvimento (i.e., pessoas que desempenham cada uma das funções do processo como pessoal de *marketing*, gerentes do projeto, desenvolvedores, designers/engenheiros de usabilidade, usuários). Eles irão constituir parte dos nós do mapa (vide Figura 5.3).

Num segundo passo, os canais e mensagens trocadas devem ser identificados (inclusive a intenção presumida das mensagens). Os canais, juntamente com os agentes, constituem os nós do mapa. Os agentes que trocam mensagens devem ser

conectados através de um canal. A Figura 5.4 ilustra parte do processo de construção do mapa.

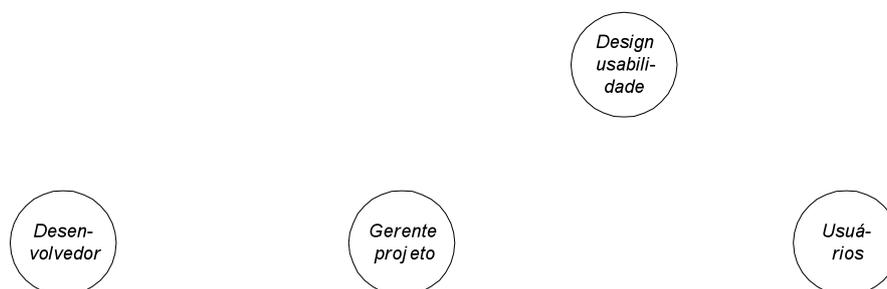


Figura 5.3: Primeiro passo - identificação de todos os agentes do processo.

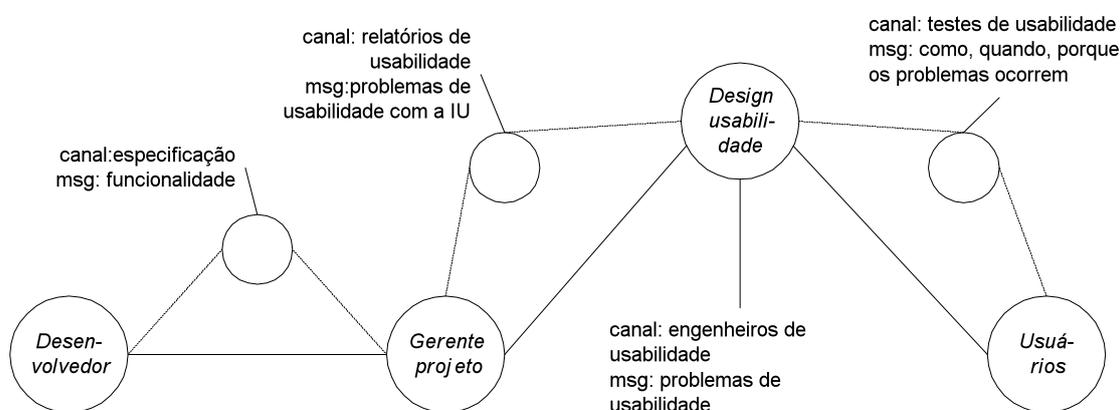


Figura 5.4: Segundo passo – as conversações entre os agentes são explicitadas (canais e mensagens identificados).

Nesta figura, o gerente do projeto se comunica com os desenvolvedores através das especificações, os engenheiros de usabilidade se comunicam com usuários através de testes de usabilidade e o gerente do projeto se comunica com os usuários utilizando os engenheiros de usabilidade como canal. Conforme apresentado no Capítulo 4, os agentes do processo também podem atuar como canais.

Sugere-se que o organizador produza ainda uma descrição simples e sucinta dos objetivos, público alvo e comunidade de usuários corrente do produto. Isso pode ajudá-lo a identificar fontes de informações relevantes para o processo de design e agentes com os quais seja importante estabelecer ou intensificar uma conversação. Já que diferentes agentes participam da fase de análise, a revisão é uma oportunidade para verificar se as perspectivas de cada um deles se complementam e corroboram e para consolidá-las em uma visão para o grupo. Além disso, esses

aspectos, somados a outras características do projeto, compõem o contexto em que as mensagens do processo são trocadas.

Recomenda-se ainda que, dentro do possível, o organizador colete estatísticas com o número e severidade dos problemas de usabilidade detectados e corrigidos durante a última versão do produto ou iteração do processo. Esses dados ajudam a avaliar se as mensagens se propagam ou não de forma efetiva e servem de registro para avaliação das melhorias no processo.

Fase 2

Na fase 2, os revisores (no caso de análise em grupo) reúnem-se para analisar o meta-modelo gerado na fase 1. Existem dois enfoques principais nesta análise: a conversação entre designers/engenheiros de usabilidade e usuários e a conversação entre os primeiros e a equipe do projeto.

Considerando a primeira conversação, os designers/engenheiros de usabilidade precisam obter informações sobre os aspectos seguintes para melhorar a usabilidade do produto:

1. quem são os usuários do sistema¹⁰. Do ponto de vista da usabilidade, isso inclui não apenas uma descrição dos usuários, mas uma descrição do que eles farão com o sistema;
2. quais são os problemas de usabilidade que eles experimentam. A identificação de um problema está geralmente associada a uma tarefa. O problema de usabilidade é caracterizado pela observação do que os usuários não conseguem ou têm dificuldade para fazer (ex., não conseguem encontrar um determinado tópico no *help*);
3. como os problemas se manifestam. Este aspecto está relacionado às reações dos usuários, isto é, às ações que executam, à maneira como eles lidam com o problema (estratégias que utilizam), o que evidencia a interpretação dos mesmos sobre o aspecto da interface a que se refere o problema. Por exemplo, em um teste de *help*, ao fazerem consultas utilizando o dispositivo de busca, alguns participantes utilizaram determinados termos e não obtiveram tópicos relevantes como resultado. Eles adicionaram termos à consulta que continuou retornando um conjunto insatisfatório. Os usuários continuaram alterando a

¹⁰ Gould, (1995, p. 97) observa que os designers freqüentemente relutam em (1) definir “quem” são os usuários e (2) em considerar essa definição com seriedade durante o processo. Contudo, ela tem implicações fundamentais nas decisões de design.

consulta, adicionando mais termos e/ou conjunções, preposições, etc., para tornar a expressão mais próxima da linguagem natural. Essas diversas iterações para refinar a consulta evidenciam um problema de usabilidade e as expectativas dos usuários;

4. porque o problema ocorre. Esta questão está fortemente relacionada às anteriores. Entendendo como os problemas ocorrem, como os usuários reagem a eles, é possível estabelecer hipóteses sobre as expectativas desses e compará-las ao comportamento do sistema. Obviamente, a identificação das causas, do “porque”, está relacionada à interpretação dos engenheiros de usabilidade e é neste momento que a qualidade de sua conversação com a equipe durante todo o projeto se manifesta. Os engenheiros de usabilidade precisam conhecer as decisões de design, as limitações e objetivos da equipe do projeto para entender o comportamento do sistema e identificar as causas do problema. No caso do exemplo usado anteriormente, a causa para o problema é que os usuários associam o conteúdo do tópico que estão buscando a determinados termos que não são as palavras-chave utilizadas na implementação do *help*.

Os diferentes métodos de usabilidade atuam como canais na conversação entre designers/engenheiros de usabilidade e usuários e respondem a subconjuntos das quatro questões discutidas (quem, quais, como e porque), com maior ou menor precisão (ex., pesquisas de *marketing* e questionários são duas técnicas distintas que podem ser aplicados para responder quem são os usuários; os dados resultantes da aplicação de diferentes métodos se complementam e se corroboram).

A qualidade dessa conversação depende da aplicação de cada um desses métodos (i.e., do planejamento de cada estudo de usabilidade) e de sua combinação (da aplicação de um conjunto de métodos que responda, da melhor forma possível, as quatro questões enumeradas acima).

Alguns projetos apresentam mais oportunidades para a aplicação de alguns métodos e maiores dificuldades para a aplicação de outros. Assim, um dos objetivos da revisão é identificar as áreas deficientes entre as questões apresentadas que sejam prioritárias e que tenham impacto no produto.

Por outro lado, as respostas às questões enumeradas anteriormente (quem, quais, como e porque) precisam ser propagadas para equipe de projeto. É através do uso dessas informações que designers/engenheiros de usabilidade podem aumentar sua participação e seu impacto nas negociações do processo de desenvolvimento.

Assim, o outro enfoque da revisão é verificar se o conteúdo das mensagens se propaga de forma eficaz para a equipe de projeto. Eficaz, neste contexto, implica que o conteúdo da mensagem é relevante, ela atingiu os agentes certos e ela foi veiculada através de canais adequados. Esses aspectos são fundamentais já que, por vezes, um determinado conteúdo atinge os agentes certos, mas, em função do canal e do contexto, não se estabelece um contato psicológico¹¹ entre os agentes da comunicação que capacite a ambos a entrarem e permanecerem em comunicação, o que é necessário para haver uma ação efetiva (neste caso, uma melhoria do artefato através da correção de problemas de usabilidade).

Assim, nesta fase do método, os revisores devem percorrer o meta-modelo e verificar:

- 1) para a conversação entre designers/engenheiros de usabilidade e usuários/interface emergente se os canais e mensagens são suficientes para responder:
 - quem são os usuários;
 - quais os problemas de usabilidade que eles experimentam;
 - como os problemas se manifestam,
 - porque eles ocorrem;
- 2) se as mensagens se propagam de maneira eficaz para a equipe do projeto, ou seja:
 - se o conteúdo de cada mensagem é relevante;
 - se ela atingiu os agentes certos;
 - se ela foi veiculada através de canais adequados.

A cada vez que algum desses critérios não for satisfeito, os revisores devem elaborar justificativa para a deficiência da comunicação.

5.2.6 Escopo e limitações do método

O percurso comunicativo focaliza o processo de design e desenvolvimento quanto aos aspectos de comunicação entre os elementos nele envolvidos. Isto promove uma reflexão sobre: (a) a eficiência das técnicas de avaliação de

¹¹ Conforme definição de Jakobson, Roman. *Lingüística e comunicação*. Ed. Cultrix.

usabilidade utilizadas, ou seja, como elas se complementam, corroboram e se são adequadas ao contexto; (b) a aceitação das práticas de usabilidade pela equipe do projeto (o que implica entender e implementar as modificações necessárias rapidamente).

Entretanto, modificar o processo como um todo é um trabalho que requer cooperação de toda a equipe do projeto e apoio da gerência. Algumas alterações menores podem ser feitas com mais autonomia pelos engenheiros de usabilidade, mas outras exigem um envolvimento de toda a equipe (ex., participação de designers e engenheiros de usabilidade nas fases de planejamento, em que se decide a funcionalidade do sistema).

Por outro lado, a aplicação do método demanda um trabalho de preparação por uma pessoa com uma visão geral sobre o processo de design/desenvolvimento, capaz de identificar os agentes com quem os designers e engenheiros de usabilidade se comunicam e com quem **poderiam** se comunicar para aumentar o impacto da usabilidade no produto (ex., *marketing*, suporte, etc). Isto exige atitude diferente do pessoal de design e usabilidade. Essa equipe tem uma visão do produto como um todo e não de um conjunto de funções. Por outro lado, muitas vezes não entendem o processo e é complexo estruturar o produto sem entender as restrições que se aplicam a cada parte.

Um aspecto interessante é que o método pode ser estendido (ou adaptado) para abranger cenários importantes, mas que não são ainda objeto de estudo das metodologias de IHC correntes. Por exemplo, numa situação de desenvolvimento interno em uma dada organização, a usabilidade de cada sistema individualmente pode ser boa (i.e., a produtividade de cada usuário é satisfatória), mas os sistemas, em conjunto, podem não ser eficientes (i.e., não permitem uma produtividade satisfatória para a organização). Por um lado, o conceito corrente de usabilidade limita-se a observar a produtividade do usuário interagindo individualmente com o produto e é neste contexto que temos aplicado o percurso comunicativo. Por outro lado, situações como a exposta anteriormente são freqüentes e precisam ser contempladas, o que implica, neste caso específico, análises mais abrangentes. O **percurso comunicativo** pode ser aplicado para esse cenário mais amplo. As perguntas do passo (a) da fase 2 (i.e., quem, quais, como e porque) serão referentes ao conjunto de sistemas e grupos de usuários, como eles se complementam, como auxiliam ou dificultam a comunicação entre esses grupos, etc. As perguntas do passo (b) da fase 2 serão referentes às comunicações entre as equipes de criação de

cada um dos sistemas, promovendo uma visão geral da organização mais consistente entre os vários grupos. Novos canais precisam ser criados com este fim.

Deve-se observar ainda que, no escopo deste trabalho, o método focaliza o processo de design de interfaces. Entretanto, ele pode ser aplicado a outros subprocessos do processo de desenvolvimento (ex., projeto do software, implementação e testes) caso sejam feitas pequenas alterações na Fase 1.

5.3 Considerações finais

Os métodos vigentes para avaliação de usabilidade focalizam o produto (a interface do software). Ao final de cada iteração, o produto é avaliado e os dados coletados realimentam a iteração seguinte para modificar o produto.

O método proposto, o **percurso comunicativo**, focaliza o processo de design e desenvolvimento. Os dados coletados têm por objetivo mudar o processo de design, tornando a comunicação entre designers e usuários mais efetiva e buscando sua integração com o processo de desenvolvimento.

Esse enfoque partiu da observação de que os métodos correntes enfatizam a relação entre designers e usuários. Entretanto, o produto que é efetivamente disponibilizado para os usuários é resultado de negociações entre toda a equipe de desenvolvimento. O processo de desenvolvimento é ainda centrado no software e aspectos de usabilidade são ainda secundários em muitas organizações ou projetos. Assim, a equipe de design precisa identificar as limitações do projeto, balanceá-las com as necessidades dos usuários e comunicá-las de maneira eficiente à equipe (i.e, encontrar canais e elaborar mensagens adequadas).

Tanto no caso dos métodos de usabilidade apresentados na literatura quanto no caso do método proposto, o objetivo final é melhorar a qualidade do produto para os usuários finais. Entretanto, o **percurso comunicativo** propõe uma melhoria contínua no processo de criação de software.

No contexto deste trabalho, melhorar o processo significa diminuir a lacuna entre os processos de design e de desenvolvimento. O **percurso comunicativo** propicia a disseminação da cultura de usabilidade para a equipe de projeto. Analogamente, designers e engenheiros de usabilidade adquirem uma visão mais geral sobre aspectos do projeto que, ainda que sejam contraditórios a objetivos de usabilidade, envolvem a organização como um todo.

Capítulo 6

Aplicação do percurso comunicativo: estudo de casos

Neste capítulo, é apresentado um estudo de casos com exemplos detalhados de aplicação do **percurso comunicativo**. Os processos analisados são de uma grande organização de desenvolvimento de produtos de prateleira.

Nessa organização, aqui nomeada ORG, há um modelo para o ciclo de vida do software que reflete a estrutura organizacional (i.e., estrutura hierárquica dos agentes, como eles se relacionam, etc.), embora cada produto possa apresentar variantes em seu processo específico.

Para ilustrar a aplicação do percurso, foram selecionados para o estudo os seguintes casos:

- o processo de criação de software da empresa como um todo, ou seja, o modelo do ciclo de vida do produto;
- o processo de design e desenvolvimento de aplicativos de entretenimento e de ferramentas para desenvolvimento de software, para ilustrar o uso e necessidade de canais e mensagens específicos para cada produto;
- o processo de design de um teste de usabilidade, para ilustrar como o **percurso comunicativo** se aplica ao design de mensagens.

6.1 O percurso comunicativo aplicado ao modelo do ciclo de vida do produto na ORG

Esta seção apresenta uma análise do modelo do ciclo de vida de software da empresa ORG. A estrutura da equipe de criação do produto reflete as áreas de especialidade (responsabilidades técnicas) de seus agentes.

A equipe de *marketing* lida com as questões do mercado e atua em atividades relacionadas ao planejamento do produto. O gerente do produto, líder da equipe como um todo, integra esse grupo.

Os principais especialistas técnicos em cada projeto são os gerentes de programação, os desenvolvedores de software e os testadores. Em cada produto, esses agentes trabalham lado a lado, formando pequenos grupos responsáveis pelo desenvolvimento de um conjunto de funções. Em geral, cada grupo consiste em um gerente de programação e em um pequeno número de desenvolvedores, que trabalham em parceria com testadores (para cada desenvolvedor, há um testador responsável por testar as funções implementadas pelo primeiro).

Os funcionários do suporte aos clientes pertencem a uma divisão separada, mas os especialistas em cada produto trabalham próximos à equipe de criação do mesmo.

Designers e engenheiros de usabilidade integram a equipe de criação e são responsáveis pelo processo de design da interface do produto. Para algumas famílias de produtos, os engenheiros de usabilidade formam um grupo paralelo, em que cada indivíduo do grupo trabalha com um produto.

Em função do escopo deste trabalho, alguns detalhes sobre o ciclo de vida serão omitidos por não terem um impacto importante nas atividades de design. Por exemplo, o grupo que produz a documentação para os usuários não foi incluído nesta análise. Eles têm um processo independente e interagem com a equipe de criação para conhecer o que vai ser disponibilizado para os usuários.

6.1.1 Descrição do modelo do ciclo do produto

Foram identificados os seguintes agentes no processo da ORG: *marketing*, gerentes de programação, desenvolvedores e testadores, designers e engenheiros de usabilidade, além da equipe de suporte ao produto.

Tipicamente, o ciclo de vida dos projetos da ORG possui três fases: a fase de planejamento, em que se produz a especificação funcional e o cronograma do projeto; a fase de desenvolvimento, em que liberações internas com um subconjunto

da funcionalidade são produzidas; fase de estabilização, em que se testa extensivamente o produto, interna e externamente (beta teste). Cada fase será detalhada a seguir.

Fase de planejamento

Nesta fase, o documento de “visão do produto” (modelo conceitual do produto) e a especificação funcional são produzidos, definindo o que será feito durante o projeto.

O documento de visão é produzido pelo gerente do produto (*marketing*) juntamente com o gerente de programação e descreve os objetivos de *marketing* para o produto. Em geral, esse documento é sucinto e define um conjunto de objetivos que dirige o processo de desenvolvimento do produto. Ele focaliza os temas, em ordem de prioridade, em que o grupo de *marketing* deseja que o produto se concentre. O documento inclui a análise de produtos competidores e projeções para versões futuras. Pode ainda discutir quais problemas serão corrigidos e quais funções serão adicionadas com base em pesquisas de mercado ou com clientes, incorporando as sugestões dos desenvolvedores e da equipe de suporte ao cliente, no caso de novas versões do produto.

A especificação é, inicialmente, um esboço preliminar dos recursos a serem disponibilizados pelo produto. Não se produz uma especificação completa e detalhada nesta fase. Ela evolui à medida que os gerentes de programação e desenvolvedores discutem detalhadamente as funções e implementam protótipos. Durante o andamento de um projeto, os produtos competidores, as necessidades dos usuários (e o que se conhece delas) e as oportunidades de mercado mudam e, por isso, é necessário haver flexibilidade. Por outro lado, o documento precisa ser suficientemente específico para que seja possível definir um cronograma inicial e permitir que os desenvolvedores progridam sem a necessidade de refazerem as funções constantemente.

O modelo de ciclo de vida na ORG é orientado pelo documento de visão e pela especificação em alto nível. Ao mesmo tempo em que assegura uma estruturação para o projeto, usar documentos em alto nível possibilita acomodar as mudanças durante o processo de desenvolvimento.

Fase de desenvolvimento

Nesta fase, a especificação funcional evolui com a participação dos desenvolvedores. Esses agentes têm uma atuação importante na definição de quais

funções serão mantidas, uma vez que eles avaliam a viabilidade técnica para implementar tais funções.

Partindo da definição da funcionalidade pretendida para todo o produto, definem-se subconjuntos da mesma e estabelecem-se datas para a sua conclusão (*milestones*). É como se o produto fosse dividido em pedaços, chamados liberações internas, de tal forma que aqueles contendo as funções mais críticas sejam liberados primeiro.

As liberações executáveis são produzidas em uma abordagem iterativa que difere do modelo cascata e se aproxima do modelo espiral (vide seção 2.2.2).

Cada liberação envolve atividades de codificação, teste e *debugging*, e o objetivo é torná-la estável, eliminar todos os *bugs* para as funções implementadas. Os desenvolvedores implementam o código seguindo a especificação funcional e testadores projetam testes para verificar se as funções e áreas do produto funcionam adequadamente. Os desenvolvedores corrigem continuamente os erros encontrados pela equipe de teste e por testes automatizados. Desenvolvedores e testadores trabalham em grande integração. A interação entre esses agentes e os gerentes de programação é também intensiva.

Essa abordagem torna o cronograma mais previsível, já que não se chega ao final do projeto com tantos *bugs* de tal forma que seja impossível prever a data para concluí-lo. Isso ainda dá à equipe uma certa flexibilidade para lidar com contingências (ex., problemas que atrasem liberações iniciais podem implicar corte de funções nas liberações finais, ou mesmo alterações no cronograma dessas últimas, etc.).

Nesta fase, a interface também é testada iterativamente. Depois da última grande liberação seus principais componentes (ex., menus, caixas de diálogo) não são mais passíveis de modificação.

Fase de estabilização

Nesta fase, o software é extensivamente testado, interna e externamente.

Internamente, a equipe de testes e os desenvolvedores encontram e corrigem *bugs* continuamente.

Externamente, o produto é submetido ao beta teste em que participam uma combinação de usuários (indivíduos, grandes corporações, distribuidores, empresas de venda de software, etc.).

Quando não são encontrados *bugs* de alta severidade, o produto é “liberado para a manufatura”, o que significa que os disquetes serão reproduzidos.

6.1.2 Aplicação do percurso comunicativo ao modelo

Percurso – fase preparatória

A tabela 6.1 apresenta uma lista dos agentes do processo que desempenham as diversas funções no ciclo do produto da organização, identifica com quem cada agente se comunica e os canais e mensagens utilizados nesta comunicação.

Deve-se observar que esta tabela focaliza as conversações mais relevantes à aplicação do método, envolvendo agentes e mensagens com impacto mais direto no processo de design/desenvolvimento. Conversações informais, como conversas de corredor, ou mesmo reuniões envolvendo toda a equipe, não foram incluídas.

A Figura 6.1 apresenta o meta-modelo de comunicação aplicado ao processo de desenvolvimento do produto nesta organização.

Agentes	Se comunicam com	Canais e mensagens
<i>Marketing</i>	Mercado/clientes	Pesquisa de mercado/clientes msg: quem são as pessoas que formam o mercado
	Gerente de programação (co-autores)	Documento de visão msg: objetivos, prioridades, o que não incluir no produto
Gerente de programação	D&EU Equipe de desenv. Suporte	Especificação msg: prioridades para as novas funções e melhorias
	<i>Marketing</i> , D&EU, Equip. desenv., Suporte	Cronograma msg: tarefas e datas para conclusão; prioridades do projeto
Designers e engenheiros de usabilidade (D&EU)	Gerente de programação, Suporte	Relatórios de usabilidade msg: características dos usuários, problemas de usabilidade detectados
	Usuários	Testes de usabilidade msg: problemas de usabilidade (quais, como, porque ocorrem – parcialmente respondidos)
	Interface emergente	Inspeções msg: problemas de usabilidade (quais são e porque ocorrem parcialmente respondidos, dependendo do método de inspeção usado)
Desenvolvedores e equipe de testes (Equipe de desenv.)	Gerente de programação	Plano de desenvolvimento msg: projeto de software
	Grupos de teste (indivíduos, corporações)	Beta teste msg: quais são os <i>bugs</i> do produto
Suporte ao cliente	Gerente do Produto	Relatórios, reuniões msg: comentários, críticas, sugestões dos usuários
	Clientes	Email, telefone msg: registram os problemas encontrados pelos clientes e provêm orientação para solucioná-los
		Questionário msg: satisfação dos clientes

Tabela 6.1: Agentes (correspondem às funções organizacionais) do processo e as conversações que estabelecem.

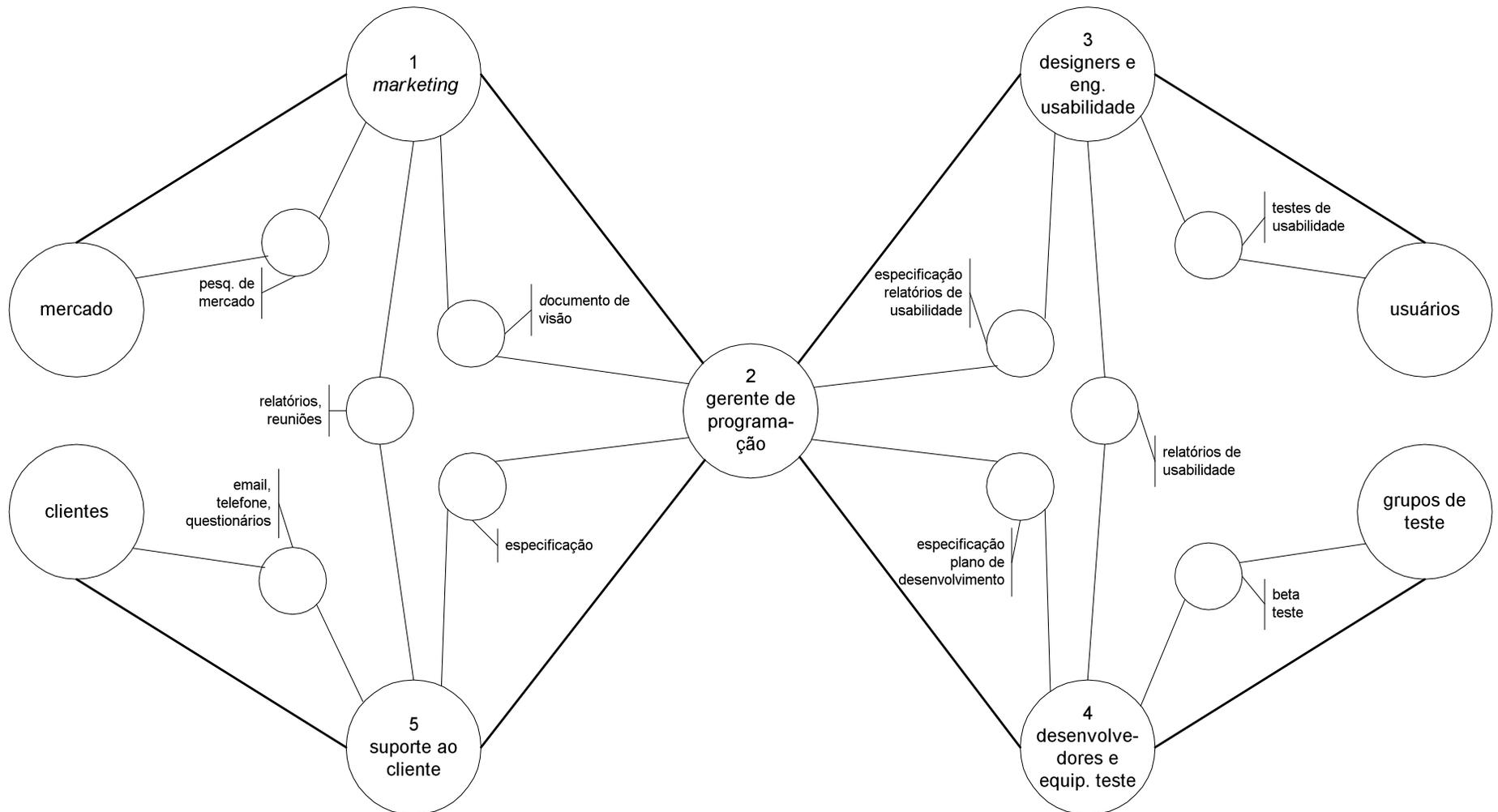


Figura 6.1 (a): Meta-modelo de comunicação aplicado ao processo de desenvolvimento da organização - comunicação entre os agentes do processo.

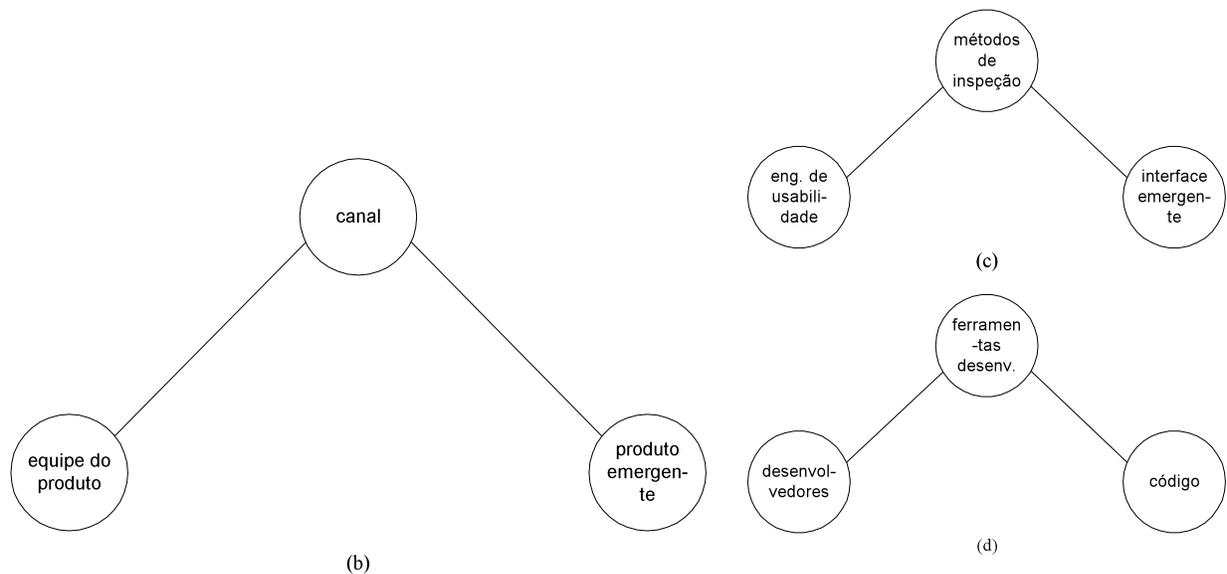


Figura 6.1: (b) Os diferentes agentes em conversação com o produto emergente; (c) conversação entre engenheiros de usabilidade e a interface; (d) conversação entre desenvolvedores e o código.

Os diferentes agentes do processo também mantêm uma conversação com o produto emergente (suas diversas instâncias) de acordo com a sua função na organização (Figura 6.1(b), 6.1(c) e 6.1(d)). Por exemplo, existe uma conversação entre os engenheiros de usabilidade e a interface emergente utilizando métodos de inspeção como canais. Já os desenvolvedores utilizam ferramentas de desenvolvimento (ex., compiladores, editores e *debugging tool*) numa conversação com o código. O conceito de conversação, neste contexto, é o mesmo utilizado por Schön [1996], quando esse se refere à conversação que o designer estabelece com o artefato em criação.

Percurso comunicativo– fase de análise

Nesta fase, percorre-se o meta-modelo analisando-se aspectos de comunicação entre os agentes. Objetiva-se verificar: (a) se há canais e mensagens suficientes para responder às questões de usabilidade (i.e., quem são os usuários, quais são os problemas de usabilidade, como e porque eles ocorrem); (b) se as mensagens relativas a cada uma dessas questões se propagam de maneira eficaz entre a equipe de projeto (i.e., se o conteúdo da mensagem é relevante para a audiência, se atinge os agentes certos e se ela é veiculada através de canais adequados).

1. Quem são os usuários?

- a) O único canal efetivamente utilizado para responder esta questão é a pesquisa de *marketing*/cliente, cuja ênfase consiste em uma descrição demográfica do público alvo (vide Figura 6.1 (a) e Tabela 6.1). A equipe de *marketing* precisa assegurar que os investimentos em um projeto sejam revertidos em lucro para a empresa (que o produto irá atender aos interesses de quem se dispõe a pagar por ele). Ao fazerem a pesquisa de mercado, são focalizados aspectos demográficos (ex., idade, gênero, faixa salarial) nos quais se baseiam para elaborar e enviar mensagens eficazes aos clientes em potencial. Por outro lado, os dados de interesse do *marketing* não são, necessariamente, os mais relevantes para o design. Por exemplo, informações sobre o tipo de equipamento dos clientes (ex., resolução de tela, tipo e velocidade de conexão à Internet) têm um impacto direto do design de um *website*, mas não são necessariamente abordados pelo *marketing*.

Deve-se considerar ainda que o entendimento sobre “quem” usa o programa possui diferentes significados para os vários agentes do processo. Na Figura 6.1 (a) podemos identificar diferentes “usuários” em comunicação com diferentes agentes: pessoas que participam de estudos de usabilidade e estão conectados aos designers/engenheiros de usabilidade; mercado, ou seja, todos os clientes em potencial, que estão conectados ao *marketing*; clientes das versões anteriores e corrente que estão conectados ao suporte; participantes do beta teste que estão conectados à equipe de teste. Conhecer os usuários através de uma perspectiva adequada é importante para que os agentes do processo possam melhorar a qualidade da conversação que estabelecem com os mesmos.

Além disso, todas essas conexões explicitam a existência de canais disponíveis que poderiam ser usados para troca de mensagens que complementassem as informações disponibilizadas pelo *marketing*. Esse mesmo agente poderia incluir perguntas de interesse dos designers e engenheiros de usabilidade na pesquisa de mercado.

Dessa forma, considera-se que o canal e mensagens utilizados não são suficientes para responder esta pergunta.

- b) O processo em questão produz uma mensagem sobre quem são os usuários cujo conteúdo é, sem dúvida, relevante para a equipe de *marketing* e se reflete em todo o processo que, por sua vez, prioriza a funcionalidade do produto definida pelo *marketing* na fase de planejamento. Entretanto, outros agentes irão precisar de informações adicionais, como foi discutido no item

anterior, ou seja, o conteúdo disponibilizado não é suficiente para elaborar mensagens relevantes para toda a equipe do projeto.

Para realmente melhorar a usabilidade do produto, a perspectiva dos designers e engenheiros de usabilidade sobre quem são os usuários deve também ser comunicada de forma eficaz (o que exige a determinação de novos conteúdos que irão compor mensagens relevantes para agentes chave no processo, conforme foi discutido no item anterior). Tal perspectiva envolve o entendimento sobre as práticas de trabalho dos usuários, contextos de uso, etc.

Sobre os agentes que recebem a mensagem em questão, o *marketing* estabelece uma conversação com o mercado a fim de entender quem são os potenciais clientes e determinar as áreas prioritárias para o produto. Essa equipe se comunica diretamente com o gerente de programação e com a equipe de suporte (vide Figura 6.1 (a)). O gerente atua como co-autor do documento de visão estabelecendo uma conversação intensiva com o *marketing* a fim de entender como o produto irá preencher as necessidades do mercado.

O gerente atua também como canal entre o *marketing* e o restante da equipe (i.e., desenvolvedores, equipe de teste, designers e engenheiros de usabilidade). Ele propaga a informação para os desenvolvedores e testadores, utilizando a especificação funcional como canal que veicula uma mensagem comunicando as funções pretendidas para o produto e prioridades para implementá-las. Da perspectiva da equipe de desenvolvimento, essa mensagem é uma tradução eficiente da mensagem intencionada pelo *marketing*, pois baseia-se nas atividades realizadas pelos primeiros.

O gerente de programação atua ainda como canal entre os designers/engenheiros de usabilidade e o marketing, embora não enviem uma mensagem específica sobre “quem são os usuários” que atenda às necessidades dos primeiros. Entretanto, esses dois últimos agentes compartilham uma série de interesses e se beneficiariam mutuamente de uma conversação direta sobre essa questão. A equipe de design precisa entender as intenções e perspectiva da equipe de *marketing* (quem são os usuários e porque a funcionalidade pretendida é importante para eles), pois isso tem implicações no design da interface. Essa equipe deve também conhecer a descrição provida pelo *marketing* para recrutar participantes representativos do segmento de mercado que se deseja atingir. Por outro lado, informações

que venham a ser obtidas pelos designers com a criação de novos canais e/ou mensagens podem ser de interesse do *marketing* e precisam ser propagadas para este grupo (ex., informações sobre as práticas de trabalho dos usuários podem ser centrais para a elaboração de mensagens para o mercado).

Assim, uma comunicação entre o *marketing* e a equipe de design não deveria ser intermediada apenas por um agente do processo (o gerente de programação, neste caso). Ao contrário, dever-se-ia estabelecer uma conversação entre eles através da criação de canais e mensagens relevantes para ambos.

A análise a seguir refere-se à adequação dos canais utilizados para veicular a mensagem sobre “quem são os usuários”.

O documento de visão estabelece uma conexão psicológica entre o *marketing* e o gerente de programação. Eles atuam como co-autores, definindo os objetivos do produto e como esse irá atender aos interesses do mercado.

O gerente atua como um canal entre os desenvolvedores/testadores e o *marketing* e é bem sucedido nesta função. Analisando-se um nível mais interno no fractal, verifica-se que o gerente traduz o conteúdo do documento de visão para o documento de especificação funcional. Esta, por sua vez, define as funções pretendidas bem como suas prioridades e é utilizada como canal de comunicação com a equipe de desenvolvimento. A especificação estabelece uma forte conexão psicológica com tal equipe, pois é possível mapear cada função desse documento em unidades de código que são os elementos manipulados por desenvolvedores e testadores.

O gerente atua, ainda, como canal entre os designers/engenheiros de usabilidade e o *marketing*. O mesmo documento de especificação é utilizado, mas, neste caso, ele não estabelece uma conexão psicológica entre os agentes. Enquanto desenvolvedores pensam em funções como unidades de implementação (o que é consistente com os objetivos de *marketing*), os designers e engenheiros de usabilidade precisam entender como as funções atuam em conjunto, como toda a funcionalidade se encaixa na prática de trabalho dos usuários. Assim, eles demandam uma visão mais abrangente que não é comunicada pela especificação.

2. Quais os problemas de usabilidade que os usuários experimentam?

- a) Quanto à suficiência dos canais e mensagens para responder esta questão, identifica-se a existência de quatro canais que podem ser utilizados para esse fim (vide Figura 6.1 (a), Figura 6.1 (c) e Tabela 6.1): inspeções, testes de usabilidade, atendimento a clientes e beta testes.

Os métodos de inspeção envolvem a conversação entre engenheiros de usabilidade e a interface emergente (vide Figura 6.1(c)). Sua aplicação identifica potencialmente uma parcela dos problemas de usabilidade da interface emergente. As mensagens produzidas (os problemas identificados) dependem do objetivo do método utilizado (ex., inspeção de aderência a padrões, consistência com uma família de produtos, facilidade de aprendizado, conformidade com diretrizes de design¹², etc.). Em geral, tais métodos focalizam problemas que foram observados com frequência em outras situações para um grande número de usuários.

No caso dos testes de usabilidade, os engenheiros preparam uma lista de tarefas e observam os usuários executá-la em laboratório. A lista de tarefas delimita as áreas que os engenheiros de usabilidade querem observar e, naturalmente, o espectro de problemas que podem ser identificados. Deve-se observar ainda que, como o teste é realizado em laboratório, grande parte dos problemas de usabilidade relacionados ao contexto de uso não serão identificados. Por exemplo, um usuário experiente pode executar as tarefas com precisão e rapidez e, no entanto, preferir o design de um produto competidor, mais conveniente ou eficiente para o seu contexto de uso (o problema de usabilidade em questão refere-se à eficiência de um design para o contexto de um usuário). Assim, os testes permitem que se identifique parcialmente os problemas de usabilidade para um dado design da interface.

Um terceiro canal relacionado a esta questão é o atendimento aos clientes provido pela equipe de suporte ao produto. Vários problemas de usabilidade podem ser identificados através deste canal com a imensa vantagem de eles surgirem em um contexto real de uso. É importante observar que, para que este canal seja usado para responder "quais os problemas de usabilidade", as mensagens enviadas pelos clientes que contactam o suporte devem ser registradas em um formato adequado (ex., que permita distingui-las de

¹² A identificação de problemas depende, não apenas do método utilizado, mas da experiência e percepção do engenheiro de usabilidade, já que, eventualmente, a aplicação rígida de regras e diretrizes pode introduzir problemas.

mensagens sobre *bugs*). Isso significa que a mensagem deveria se adequar aos agentes e seus objetivos.

Finalmente, o beta teste é um canal existente que também poderia ser usado para identificar quais os problemas de usabilidade com a interface. Apesar de o beta teste ser realizado na fase de estabilização, em que mudanças no design já não são mais possíveis (a interface não pode ser alterada nesta etapa do processo), os dados coletados seriam bastante úteis para que designers e engenheiros de usabilidade pudessem influenciar a fase de planejamento da versão seguinte do produto. Entretanto, apesar do canal existir, o foco de sua utilização é a identificação de *bugs* e não de problemas de usabilidade¹³ (i.e., não se aproveita a existência do canal para este fim). Observando-se que o beta teste envolve a participação de uma variedade de usuários (ex., grandes corporações, representantes de venda, indivíduos) em seu próprio contexto de uso, a utilização deste canal poderia prover dados interessantes aos designers e engenheiros de usabilidade.

Concluindo, sobre a suficiência dos canais e mensagens para determinar quais os problemas de usabilidade, o modelo de ciclo de vida da ORG propõe o uso de métodos de inspeção e testes de usabilidade. Eles auxiliam na identificação de uma parcela dos problemas de usabilidade. Observa-se a disponibilidade de dois outros canais, suporte aos clientes e beta testes, que podem auxiliar na identificação de um grande número de problemas de usabilidade. Para tanto é necessário, contudo, proceder a criação de novas mensagens.

- b) Neste ponto, iremos analisar a propagação de mensagens quanto à relevância do conteúdo, agentes atingidos e canais utilizados.

Dois aspectos devem ser observados: conteúdos produzidos pelos engenheiros de usabilidade e propagados para a equipe de projeto; conteúdos produzidos por outros agentes da equipe que podem beneficiar as atividades de usabilidade.

Considerando a análise do item 2.(a), os canais suporte ao cliente e beta teste poderiam potencialmente produzir conteúdos relevantes para a equipe de usabilidade. Por exemplo, problemas identificados através desses canais

¹³ Conforme explicitado no Capítulo 4 - Modelo fractal de comunicação, cada elemento do processo (ex., teste do produto, estudo de usabilidade) é uma mensagem que também é objeto de design e reflete a intenção de quem as emite. No caso do beta teste, a mensagem enviada aos participantes tem por intenção apenas a identificação de *bugs*.

podem incluir uma descrição das características demográficas do usuário, do domínio e contexto de uso, etc. Seria possível estabelecer hipóteses sobre associações entre categorias de usuários e padrões de comportamentos e investigá-las através de outros canais (ex., estudos de campo).

No caso do suporte, o conteúdo gerado não se propaga até a equipe de design. Como pode ser observado na Figura 6.1 (a), esses agentes se comunicam utilizando o gerente de programação como canal, mas não há troca de mensagens para comunicar o conteúdo em questão. Por outro lado, para poder gerar tais mensagens, o suporte precisa registrar os dados relativos a problemas de usabilidade de maneira sistemática e estruturada. Observa-se aí a necessidade de uma conversação entre suporte e equipe de design, de preferência não intermediada por outros agentes do processo.

De acordo com o modelo do ciclo de vida da ORG, o beta teste não produz um conteúdo relevante para as atividades de usabilidade (como foi visto no item 2.(a), não se coletam dados sobre usabilidade). A comunicação entre designers/engenheiros de usabilidade e testadores é intermediada por dois canais: (a) relatórios de usabilidade, enviado pelos primeiros, que não promovem, necessariamente, uma conversação em que testadores terão um turno como remetentes e que serão discutidos a seguir; (b) gerente de programação (vide Figura 6.1 (a)), o que revela um distanciamento entre os agentes em questão, considerando ainda que não há troca de mensagens sobre o aproveitamento de informações de beta teste para atividades de usabilidade. Para que possa aproveitar o canal disponível, as equipes de usabilidade e teste precisam estabelecer uma conversação para planejar os testes de tal forma que ambos os grupos se beneficiem de sua aplicação. Novos canais e mensagens precisam ser criados com esse fim.

Quanto ao conteúdo gerado pelos engenheiros de usabilidade, pode-se considerar as melhorias na usabilidade do produto como medida da eficiência de sua propagação. O conteúdo em questão é relevante para toda a equipe, embora determinados aspectos causem mais impacto a determinados agentes¹⁴.

Um relatório detalhado sobre o estudo de usabilidade e um *email* destacando de maneira sucinta os problemas mais sérios (ex., observados com

¹⁴ Por exemplo, dados quantitativos (especialmente se se referem a um grande número de participantes) são eficazes para desenvolvedores, que focalizam muito o custo de implementação. A equipe de *marketing*, por outro lado, se beneficiaria de dados quantitativos associados a cenários de uso (em que eles pudessem visualizar um cliente representativo do mercado descrito em termos dos dados demográficos que eles usam).

frequência ou que impeçam ou dificultem muito a execução da tarefa) são os canais utilizados para comunicar o conteúdo aos gerentes de programação e equipe de desenvolvimento.

Entretanto, o relatório de usabilidade não estabelece uma conexão psicológica entre os agentes da comunicação. Geralmente, ele não é lido pela equipe, mas sim arquivado e eventualmente lido por outros engenheiros de usabilidade. Tendo em vista o grande volume de documentação gerada no processo, a leitura cuidadosa de cada documento consumiria muito tempo, e faz parte da cultura da empresa produzir mensagens mais curtas que focalizem os aspectos primordiais.

Por outro lado, o *email*, por ser muito sucinto, nem sempre provê detalhes suficientes para o entendimento da mensagem. Principalmente, é difícil contextualizar o problema em um *email* sucinto, mostrar suas implicações na interação como um todo, o que limita as informações para tomada de decisão sobre quais problemas corrigir, quais podem ser deixados para versões futuras, etc., considerando as limitações do projeto.

Resumindo, é possível tornar as comunicações no processo referentes à questão “quais os problemas de usabilidade” mais eficazes. Para identificá-los, pode-se fazer melhor uso de canais disponíveis (i.e., beta teste e suporte) sendo necessária a criação de novas mensagens para os usuários e novos canais e/ou mensagens entre os agentes do processo (entre designers/engenheiros de usabilidade e suporte e entre aqueles e testadores). Deve-se ainda ressaltar que engenheiros de usabilidade precisam identificar canais e mensagens mais eficientes (que estabeleçam uma conexão psicológica com os destinatários) para comunicar problemas de usabilidade, além dos relatórios e *emails*.

3. Como os problemas ocorrem?

- a) Esta questão envolve a observação das reações dos usuários. A análise seguinte trata da suficiência das mensagens e canais para determinar essas reações.

Os métodos de inspeção podem evidenciar potencialmente uma parcela dessa questão. Por exemplo, no *cognitive walkthrough* (Wharton *et al.*, 1994), os revisores examinam cada “ação” que os usuários executariam para completar uma tarefa e tentam elaborar uma estória verídica sobre a interação típica desses com a interface (que elucida a questão “como”).

Os testes de usabilidade permitem que os usuários sejam observados cuidadosamente. Toda a seqüência de passos realizada é registrada através de vários canais (ex., fitas de vídeo, registros do próprio engenheiro durante a sessão, arquivos de *log*, etc.). Cada ação (e reação) dos usuários pode ser detalhadamente observada e revista. Tudo isso torna este canal bastante apropriado para determinar como os problemas ocorrem.

Por outro lado, os canais “suporte aos clientes” e “beta teste” são bastante apropriados para se coletar quais são os problemas de usabilidade, mas não como eles se manifestam (quais as reações dos usuários, como eles lidam com o problema, etc.). Não é possível, através apenas das informações coletadas através desses canais, reconstruir a experiência dos usuários ao lidarem com um problema de usabilidade e estabelecer hipóteses sobre suas expectativas, entendimento, etc. Entretanto, se os engenheiros de usabilidade tiverem acesso a esses dados, eles podem utilizá-los durante o design de um teste para identificar “como” tais problemas ocorrem. Essa combinação de diferentes fontes de informação teria um impacto considerável na qualidade dos dados coletados.

- b) Sobre a propagação para a equipe de desenvolvimento, a discussão do item 3.(a) aponta que os principais canais para tratar a questão “como” são os testes de usabilidade e métodos de inspeção. Esses, por sua vez, são utilizados pelos engenheiros de usabilidade em conversações com os usuários e com a interface emergente, respectivamente.

É neste ponto que os engenheiros de usabilidade precisam ser bastante eficientes como canal entre a equipe do produto e os usuários. É comunicando claramente “como” os usuários reagem e quais as suas expectativas que a equipe poderá entender a lacuna entre o que foi especificado e o que os usuários precisam e esperam.

Essa informação, se bem entendida, pode causar impacto na equipe, uma vez que seus integrantes poderão perceber aspectos frustrantes da experiência dos usuários. No entanto, ela é propagada para o gerente de programação e equipe de desenvolvimento através de relatórios de usabilidade e *emails* resumindo o conteúdo dos relatórios. Os relatórios, conforme foi discutido no item 2(b), não são lidos pela maior parte da equipe e os *emails* resumidos tornam difícil a descrição suficientemente detalhada do contexto relativo a um problema de usabilidade (ex., no caso de jogos, “como” os usuários

reagem é uma informação fundamental e é necessário prover uma série de detalhes para descrevê-la).

Deve-se observar ainda que, o conteúdo sobre a questão “como” é, da mesma forma, relevante para outros agentes do processo, como a equipe de *marketing* e também nesse caso ele precisa ser comunicado adequadamente para a audiência. Por exemplo, os engenheiros de usabilidade podem produzir *videoclip* com problemas identificados durante o teste de usabilidade do produto completo (liberado para o mercado) que não tenham sido corrigidos e usá-lo na fase de planejamento. Pode-se agrupar as seqüências de vídeo de cada participante para uma dada tarefa, o que comunica de forma objetiva e intensa um determinado problema.

4. Por que os problemas de usabilidade ocorrem?

- a) Esta questão se refere à identificação das causas dos problemas de usabilidade (ex., um usuário não consegue executar uma função ou encontrar uma informação que procura porque não associa sua tarefa ao rótulo utilizado pelo sistema).

Ela está intrinsecamente relacionada à anterior. Identificando “como” os problemas se manifestam, como os usuários reagem a eles, é possível estabelecer hipóteses sobre as interpretações dos mesmos, sobre suas expectativas, etc. O “porque” envolve a indicação da lacuna entre tais hipóteses e o que foi efetivamente implementado; entre o que os usuários acham que devem fazer e o que a equipe acha que eles deveriam fazer ao interagir com um determinado elemento da interface.

Assim, a análise dessa questão envolve aspectos um pouco diferentes das anteriores. Por um lado, o êxito da determinação do “porque” envolve sucesso na identificação do “como” (vide item 3.(a) desta subseção). Por outro lado, o mapeamento do “como” para o “porque” envolve uma boa conversação entre engenheiros de usabilidade e o restante da equipe durante todo o projeto. A avaliação da suficiência de canais e mensagens aqui é mais subjetiva, já que, nessa conversação, o uso de canais informais, como conversas de corredor, têm um grande impacto. Diferentes membros da equipe compartilham sua perspectiva sobre uma mesma questão em conversas informais, o que provoca reflexão em outros membros e reflete na evolução do projeto como um todo. O uso desse canal presencial promove uma maior fluidez à conversação. Até mesmo a organização física do ambiente ajuda a

promover a integração com a equipe¹⁵, já que a proximidade física facilita que as pessoas se encontrem rapidamente sem a necessidade de agendamento, etc. Entretanto, é bastante produtivo levantar e discutir essas questões durante a inspeção em grupo.

- b) Quanto à propagação para a equipe de projeto, o modelo do ciclo de desenvolvimento propõe a utilização de relatórios de usabilidade e *emails* resumindo seu conteúdo dirigidos ao gerente de programação e à equipe de desenvolvimento. A mesma análise do item 3.(b) pode ser aplicada aqui, ressaltando-se, no entanto, que essa questão (i.e., porque os problemas de usabilidade ocorrem) traz oportunidades de discussão e integração entre a equipe, pois os engenheiros de usabilidade podem aprender aspectos do projeto que limitam as soluções de design, além de esclarecer aos desenvolvedores o impacto de alguns problemas na experiência dos usuários.

6.1.3 Considerações finais

Com a aplicação do percurso comunicativo foram identificadas algumas deficiências no processo e oportunidades para melhoria. Com a criação de novos canais (e de novas mensagens, aproveitando os canais existentes), pode-se alcançar um processo de design mais integrado ao processo de desenvolvimento, a melhor utilização dos recursos disponíveis e, conseqüentemente, produtos com mais qualidade.

Os itens seguintes resumem os resultados da análise:

- O *marketing* é o único agente envolvido na identificação de quem são os usuários, mas outros agentes do processo demandam um entendimento sobre essa questão. Em particular, observa-se a necessidade de uma conversação (através da criação de novos canais e mensagens) entre a equipe de *marketing* e a de design. No processo corrente, a comunicação entre esses agentes é primordialmente intermediada pelo gerente de programação;

¹⁵ Em uma entrevista informal na ORG., foi identificado um problema dessa natureza. Em um dos projetos, toda a equipe técnica estava fisicamente próxima, exceto o engenheiro de usabilidade que estava circunstancialmente em outro prédio. Através de conversas de corredor, encontros rápidos, etc., alguns aspectos da especificação eram modificados. Por sua proximidade física, os membros da equipe eram simultaneamente informados e isso era imediatamente refletido em suas tarefas, executadas já considerando a alteração. Entretanto, há um intervalo de tempo até que a especificação seja alterada para refletir tal mudança. O engenheiro de usabilidade baseava-se na especificação e, em função disso, sempre recebia as alterações com atraso, havendo assim, uma falta de sintonia entre ele e a equipe.

- suporte aos clientes e beta testes são canais poderosos para a identificação de problemas de usabilidade (respondem à questão “quais”). No entanto, no primeiro caso, os dados coletados não se propagam até os engenheiros de usabilidade. Observa-se, portanto, a necessidade de criação de um canal entre esses últimos e a equipe de suporte. Por outro lado, o beta teste não é usado para a coleta de dados para identificação dos problemas de usabilidade. Sua utilização envolveria a necessidade de criação de novos canais (ex., entre a equipe de design e de testes) e mensagens. Em ambos os casos, a criação de novos canais para comunicação entre os agentes do processo tem por objetivo complementar a comunicação intermediada pelos gerentes de programação, facilitando a propagação das mensagens;
- novos canais, que estabeleçam uma conexão psicológica entre os agentes, precisam ser criados para que os engenheiros de usabilidade comuniquem os problemas de usabilidade à equipe de desenvolvimento, já que os *emails* e relatórios de usabilidade têm limitações. O uso desses canais reflete a cultura da empresa, em que se pratica a criação de um registro suficientemente detalhado (neste caso, relatórios de usabilidade) e um resumo ressaltando os tópicos principais, tendo em vista que o volume de documentação produzido é muito grande para ser analisado com cuidado por cada membro da equipe que focaliza apenas os elementos mais importantes. Aspectos culturais da empresa devem ser considerados quando da criação de novos canais (engenheiros de usabilidade devem estar atentos às oportunidades);
- com a criação de novos canais e mensagens para coletar e propagar dados sobre quais os problemas de usabilidade, engenheiros de usabilidade podem tornar a comunicação com usuários mais eficaz. O contexto de desenvolvimento de produtos de prateleira, como é o caso da ORG, limita, sob alguns aspectos, a interação com usuários. A utilização de dados do suporte e beta teste significaria o aproveitamento de oportunidades já existentes. Dados coletados através dessas fontes poderiam servir como entrada para o design de testes de usabilidade (a partir de uma lista de “quais” os problemas, pode-se observar “como” eles ocorrem). Desta maneira, o conjunto de métodos utilizado iria se complementar com mais eficiência;
- pode-se observar, de acordo com a análise feita, que os objetivos de *marketing* dirigem todo o processo, o que é consistente com o contexto da

empresa (desenvolvimento de produtos de prateleira em uma economia capitalista). Essa visão se reflete no desenvolvimento voltado para a funcionalidade. Por outro lado, aspectos de usabilidade estão relacionados, muitas vezes, à maneira como tais funções se estruturam no produto como um todo, o que determina a experiência dos usuário e implica a necessidade de se considerar a visão da equipe de design, especialmente nas fases iniciais.

Os resultados da análise apontam a necessidade de criação ou adequação de canais e mensagens que conduzam à melhoria do processo como um todo, potencializando a melhoria do produto. Os tipos de canais e mensagens são explicitados, tanto para conversação com os usuários, buscando identificar as questões “quem”, “quais”, “como” e “porque”, quanto para propagação para a equipe. É importante ressaltar, ainda, que as oportunidades existentes no processo foram evidenciadas pela aplicação do método (ex., beta teste para identificação da questão “quais”), o que é economicamente importante.

6.2 Como as especificidades do produto são evidenciadas pelo meta-modelo

Em geral, a literatura em IHC apresenta métodos e técnicas para design de interfaces e avaliação de usabilidade, mas não resalta dois aspectos importantes que são evidenciados pelo entendimento da comunicação, conforme a perspectiva do modelo fractal. Quais sejam:

- os métodos e técnicas apresentados na literatura são “modelos” que orientam designers e engenheiros de usabilidade. Sua aplicação envolve um entendimento profundo do próprio método ou técnica, do contexto em que será usado, dos objetivos específicos que se deseja atingir, etc., o que requer algumas adaptações para adequá-los a um caso específico. Essa questão é importante e deve ser explicitada. O modelo fractal de comunicação (Capítulo 4) a evidencia na medida em que percebe tais métodos como canais de comunicação entre designers/engenheiros de usabilidade e usuários. A aplicação do método de usabilidade (instanciação) corresponde à criação de uma mensagem (em que ele é particularizado para o contexto de um projeto);

- assim como o método de usabilidade precisa ser particularizado, as formas para propagar as informações obtidas com a sua aplicação também dependem das especificidades do projeto como objetivos do produto, características da equipe, etc. Identificar canais que tornem essa propagação efetiva (resultem em mudanças e melhorias no produto) é tão importante quanto o sucesso na aplicação de um método.

Os estudos de caso apresentados nesta seção exemplificam esses aspectos, apontando como os contextos de cada projeto trazem diferentes oportunidades e obstáculos para o uso de mensagens e canais específicos, tanto na comunicação com os usuários (design de cada estudo de usabilidade), quanto na propagação dessa comunicação para a equipe. Serão apresentados apenas os pontos em que os processos de criação dos produtos se diferenciam do modelo do ciclo de vida (seção 6.1).

6.2.1 Jogos

Os agentes envolvidos no processo descrito a seguir são os engenheiros de usabilidade (EU), a equipe de *marketing*, designers, usuários, e produtos competidores (vide Tabela 6.2 e Figura 6.2).

Enquanto o objetivo da maioria dos softwares é auxiliar os usuários na execução de uma tarefa ou localização de uma informação, o objetivo de um jogo é proporcionar diversão aos usuários. Esse é o aspecto primordial que distingue os jogos das outras categorias de software, mas as características enumeradas a seguir são também peculiaridades da categoria em questão:

- os jogos apresentam desafios que devem ser superados de acordo com os esforços dos usuários, com as habilidades que eles desenvolvem. Assim, aspectos fundamentais da interação são tornar claras as regras dos jogos, bem como as respostas às ações dos usuários (*feedback*);
- jogos de computador constituem uma forma de interação social em que pequenas comunidades compartilham um interesse comum. Além de interagirem com jogos, os usuários interagem entre si utilizando os primeiros como motivação. Cabe observar que isso não ocorre apenas para jogos que possam ser usados simultaneamente por vários usuários. Pessoas que se interessem pelos mesmos jogos discutem estratégias para ultrapassar um estágio, comentam sobre os personagens, sons, efeitos gráficos, mesmo que

os jogos sejam individuais. Muitos usuários criam e utilizam um espaço comum virtual, no qual vários membros podem contribuir, e a motivação, o tema discutido, é sempre um conjunto de jogos que interessam àquela comunidade;

- o design de um jogo envolve, em geral, além de toda a parte gráfica, a criação de uma história, de um tema. Estórias bem sucedidas viabilizam a criação de novas versões de um jogo¹⁶.

Os objetivos de usabilidade no processo de criação de jogos precisam refletir essas peculiaridades, já que elas determinam aspectos fundamentais da comunicação com os usuários e de sua interação com o produto.

Assim como no caso de outras categorias de software, os testes de usabilidade envolvem observar como as pessoas realizam tarefas ou reagem a elas. Entretanto, no caso dos jogos, a lista de tarefas precisa ser elaborada considerando os objetivos do teste (ou seja, observar o que afeta a diversão dos usuários). Na ORG, os designers participam (como expectadores) dos testes de usabilidade. Eles comparecem ao laboratório e observam duas ou três sessões (vide Figura 6.2, testes de usabilidade como canal entre designer e engenheiro de usabilidade). Antes de prosseguir com as demais sessões, engenheiros de usabilidade e designers usam a lousa do laboratório¹⁷ para discutir os problemas de usabilidade e propor soluções, que são implementadas e, posteriormente, são realizadas as demais sessões (Figura 6.2 lousa como canal entre os agentes mencionados).

Além dos testes de usabilidade descritos acima que são realizados iterativamente durante o processo, há também a aplicação de *benchmarks*. Tipicamente, testes de usabilidade *benchmark* têm por objetivo medir o desempenho dos usuários (ex. tempo necessário para executar tarefas, taxa de erros). No caso dos jogos, *benchmarks* são aplicados com o propósito de coletar dados sobre as preferências dos usuários em relação a um jogo, que incluem o quanto eles gostam dos sons, gráficos, se eles comprariam o jogo ou o recomendariam a outras pessoas.

As informações coletadas durante a aplicação de um *benchmark* são usadas pela equipe de *marketing* durante a fase de planejamento (Figura 6.2, relatórios do *benchmark* como canal entre engenheiros de usabilidade e *marketing*).

¹⁶ Pode-se estabelecer um paralelo com filmes que envolvem estórias de sucesso permitindo a criação de continuações que são, inclusive, aguardadas pelo público.

¹⁷ O laboratório é dividido em duas partes: uma destinada aos usuários e outra destinada aos engenheiros de usabilidade e pessoas que queiram observar os testes. Elas são separadas por um espelho *one way* e, do lado dos engenheiros, há uma lousa.

Essa equipe realiza ainda uma análise competitiva para assegurar que o produto terá as mesmas características que outros jogos populares (Figura 6.2, análise competitiva como canal entre *marketing* e produtos competidores) e repassa essa informação para designers e engenheiros de usabilidade (Figura 6.2 reuniões, discussões como canais entre esses agentes).

A equipe de design, então, trabalha na proposição de um tema (uma estória) e no design gráfico. No caso de produtos inteiramente novos, algumas idéias emergem totalmente dos designers. No caso de novas versões de produtos existentes, eles propõem algumas inovações. Em ambos os casos, deve-se considerar que o tema é um elemento fundamental para a criação de novas versões, conforme foi discutido anteriormente nesta subseção.

A aplicação do percurso comunicativo

Percurso - fase preparatória

Agente	Com quem se comunica	Canais e mensagens
Engenheiro de usabilidade	Usuários	Teste de usabilidade msg: problemas de usabilidade (que afetam a diversão dos usuários)
		Teste <i>benckmark</i> msg: preferências (quantos os usuários gostam dos sons, gráficos, etc.)
		Reuniões msg: discussões sobre características do produto
	Designers	Reunião (laboratório) msg: problemas de usabilidade
		Resumos na lousa do laboratório msg: problemas de usabilidade e soluções
	<i>Marketing</i>	Relatório do <i>benckmark</i> msg: como os usuários respondem aos jogos
Relatório de usabilidade msg: problemas de usabilidade		
<i>Marketing</i>	Produtos competidores	Análise competitiva msg: características dos jogos populares
	Designers	Documento de visão, reuniões msg: garantir que o produto terá as mesmas características dos jogos populares
Usuários	Designers	Teste de usabilidade msg: problemas de usabilidade
		Teste <i>benckmark</i> msg: como as pessoas respondem aos jogos.

Tabela 6.2: Agentes do processo e as conversações que estabelecem.

A Tabela 6.2 apresenta uma lista dos agentes envolvidos, identifica com quem cada agente se comunica e os canais e mensagens utilizados nesta comunicação.

A Figura 6.2 apresenta o meta-modelo de comunicação aplicado ao processo de criação de jogos.

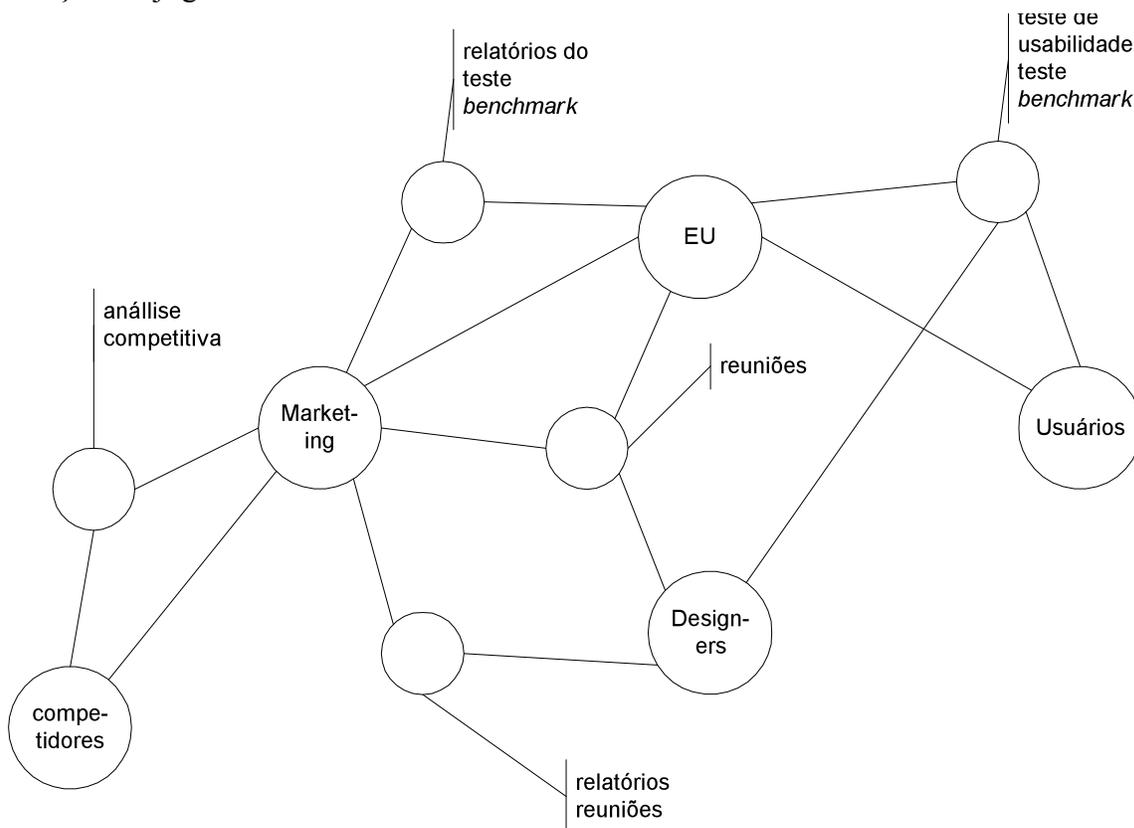


Figura 6.2: Meta-modelo de comunicação aplicado ao processo de criação de jogos.

Percurso - fase de análise

Nesta fase, o meta-modelo produzido é percorrido verificando-se a comunicação entre os agentes. Conforme foi ressaltado anteriormente, apenas os pontos distintos do modelo do ciclo de vida do produto serão discutidos.

1. Quem são os usuários.

- a) Os testes *benchmark* são aplicados com o propósito de coletar informações sobre as preferências dos usuários ao interagirem com um jogo (ex., se gostam dos sons, gráficos, se comprariam e recomendariam o produto). Essa informação descreve, na perspectiva dos engenheiros de usabilidade, “quem” são os usuários, uma vez que evidencia seu envolvimento com os jogos, o que os perturba ou incomoda, o que os entusiasma (do que eles gostam e do que não gostam na experiência de interação com o jogo).

- b) No tocante à propagação das mensagens, entender “quem” são os usuários através da perspectiva descrita acima é importante para os designers e para o *marketing*.

Esses últimos utilizam efetivamente os dados dos testes *benchmark* na fase de planejamento (Figura 6.2, relatório de *benchmark* como canal entre *marketing* e engenheiros de usabilidade). As informações coletadas refletem o envolvimento dos usuários com o jogo, resumem a experiência dos mesmos a interação e identificam especificamente se eles o recomendariam a outras pessoas ou o comprariam. Tais informações são particularmente relevantes para produto em questão, já que um dos aspectos que distingue os jogos das demais categorias de software é o fato deles estimularem as interações sociais. As impressões (quanto aos gráficos, sons, etc.) de um usuário são também compartilhadas por outros tantos. Além disso, suas experiências, positivas ou negativas, se propagam com facilidade entre pessoas com interesse pelos mesmos tipos de jogos. O design da interação precisa valorizar os aspectos positivos e buscar soluções para os negativos.

Assim sendo, o conteúdo é relevante para designers e para o *marketing* e é propagado para ambos pelos engenheiros de usabilidade. No caso do *marketing*, os canais usados são o relatório do *benchmark* e conversações informais. Pode-se considerar que os canais são adequados, já que o conteúdo da mensagem é realmente usado pelo *marketing* durante o planejamento do produto. Os designers observam as sessões do teste e recebem dos próprios usuários mensagens sobre suas reações (Tabela 6.2 e Figura 6.2, teste de usabilidade como canal entre usuários e designers), além de discutirem os resultados com os engenheiros de usabilidade.

Utilizar essas informações na fase de planejamento, ao mesmo tempo em que se coleta dados sobre produtos competidores, reflete que a perspectiva dos usuários será considerada em uma etapa importante do ciclo de vida. Desta forma, pode-se considerar que a propagação da mensagem em questão foi satisfatória.

2. As questões “quais”, “como” e “porque” serão discutidas conjuntamente neste item.
- a) Os testes de usabilidade são o canal utilizado para a identificação de quais são os problemas de usabilidade, como e porque eles ocorrem.

É importante observar que, para alguns jogos, a mudança de estágio envolve as mesmas tarefas, ações, etc., apenas com um nível de dificuldade maior. Outros jogos, contudo, envolvem um aprendizado, um entendimento sobre os personagens, enredo e uma série de outros aspectos. A mudança de estágio implica horas ou até mesmo dias de interação e os usuários podem salvar uma partida em um ponto e retomá-la posteriormente.

Assim, analisando a suficiência dos canais e mensagens para responder à questão “quais”, os testes de usabilidade expõem potencialmente uma parcela dos problemas de usabilidade da interface (vide seção 6.1). No caso de jogos em que a mudança de estágio demanda a experiência de fases anteriores, o teste de usabilidade em laboratório, em função de limitações na duração da sessão, não é tão apropriado para apontar “quais” os problemas de usabilidade. A criação de canais que focalizem esse aspecto é particularmente importante para esses tipos de jogos.

Um outro aspecto particular dos jogos é que há outros atributos de usabilidade envolvidos na interação (vide seção 2.1, sobre atributos de usabilidade). Por exemplo, no caso citado anteriormente, em que ultrapassar um estágio demanda horas de interação, estão presentes aspectos como o envolvimento dos usuários com o jogo, o quanto eles acham o jogo prazeroso. O jogo precisa se manter interessante, envolvente e não pode se tornar monótono em um determinado ponto. Esses aspectos não podem ser observados em sessões curtas.

Por outro lado, testes de usabilidade são um canal apropriado para a observação de “como” os problemas se manifestam, uma vez que permite a observação detalhada das reações dos usuários (conforme discutido na seção 6.1).

A determinação do “porque” está relacionada ao entendimento de “como” os problemas se manifestam (expectativas dos usuários, etc.) e ao conhecimento dos engenheiros de usabilidade sobre as decisões de implementação, design, limitações técnicas, etc. Isso pressupõe uma interação entre engenheiros, desenvolvedores e designers. Esses últimos observam as sessões de teste e discutem os problemas identificados com os engenheiros. Contudo, não existe na Figura 6.2 uma conexão entre engenheiros de usabilidade e desenvolvedores. Essa interação é necessária. Por exemplo, alguns jogos envolvem questões de desempenho do sistema que são fundamentais para a qualidade da interação. Os desenvolvedores precisam

estar integrados à equipe de design para avaliar, iterativamente, a viabilidade de implementação eficiente das soluções de design propostas. Observa-se aí a necessidade de criação de novos canais e mensagens.

- b) Quanto à propagação, os designers comparecem ao laboratório e observam duas a três sessões. O teste de usabilidade atua como um canal, em que os usuários enviam mensagens sobre os problemas de usabilidade aos designers, cujo o conteúdo é fundamental para esses agentes. (Figura 6.2, o teste como canal entre usuários e designers).

No caso dos jogos, existe uma diversidade de cenários, dependendo do caminho percorrido por cada usuário, em que podem ocorrer problemas de usabilidade. Por outro lado, tendo em vista que o objetivo do teste é encontrar problemas que possam afetar a diversão dos usuários, suas reações precisam ser percebidas com clareza. Não é simples descrever tudo isso com detalhes suficientes em um relatório de usabilidade para que os designers possam propor soluções de redesign. No entanto, utilizar o teste de usabilidade como canal através do qual os usuários podem enviar mensagens implica tornar a comunicação entre esses agentes mais eficaz.

Após as sessões iniciais, designers e engenheiros de usabilidade, discutem os problemas, suas causas e propõem soluções, utilizando a lousa no laboratório como canal adicional na troca de mensagens (além da própria fala). Durante as sessões, resumos e anotações sobre as mesmas são colocados na lousa. A perspectiva de um agente é compartilhada com o outro de maneira dinâmica.

É importante observar que as demais sessões somente são realizadas após a correção dos problemas. Isso significa que as mensagens sobre os problemas de usabilidade e sobre as soluções para os mesmos atinge agentes do processo e resulta em modificações efetivas.

Assim, pode-se dizer que os canais utilizados nas comunicações para identificar e propagar as questões sobre os problemas de usabilidade para a criação de jogos são eficazes.

Considerações finais

Como foi visto na seção 6.1, tipicamente se utiliza engenheiros de usabilidade como canais entre designers e usuários na ORG (e os engenheiros, por sua vez, utilizam relatórios de usabilidade - Figura 6.3(a)). No caso dos jogos, essa comunicação foi aproximada (Figura 6.3(b)), o que é relevante, especialmente no contexto em questão. Não é simples descrever as reações dos usuários e o cenário em que ocorre um problema de maneira suficientemente detalhada em um relatório de usabilidade. A maior proximidade entre os designers e usuários implica uma maior eficácia na comunicação entre esses agentes.

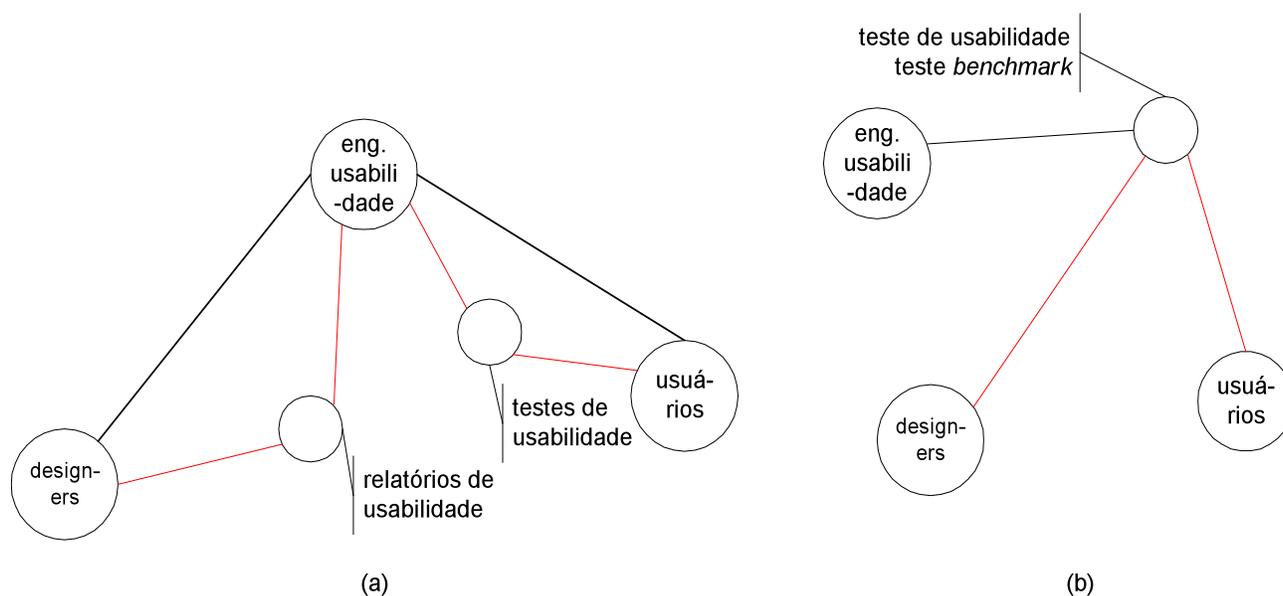


Figura 6.3: Teste de usabilidade como canal entre designers e engenheiros de usabilidade em: (a) modelo do ciclo de vida; (b) processo de criação de jogos.

Por outro lado, engenheiros de usabilidade e designers utilizam canais que permitem a cooperação de ambos na identificação das causas dos problemas e na proposição de soluções. Após as sessões iniciais, eles e os engenheiros de usabilidade permanecem no laboratório (que dispõe inclusive de equipamentos para rever as fitas de vídeo) e discutem os problemas, o que assegura que não há atraso na comunicação, que as mensagens são recebidas e compreendidas (relatórios demoram a ser produzidos, não se pode assegurar que são lidos e que seu conteúdo é entendido com clareza). A comunicação entre esses agentes foi também aproximada (inclusive temporalmente). Essa maior integração entre designers e engenheiros de usabilidade não ocorre no modelo do ciclo de vida (seção 6.1).

Outro aspecto importante para jogos está relacionado ao design da lista de tarefas do teste de usabilidade. Em testes de outras categorias de software as tarefas

podem ser quebradas em pequenas ações atômicas, com começo, meio e fim (ex., escolher uma fonte para um trecho de texto, um tamanho, colocá-lo em negrito, etc.), o que permite que se crie uma lista explorando diversas áreas do produto. O jogo, contudo, pressupõe um contínuo de ações, sendo difícil fragmentar a interação em pequenas tarefas, principalmente porque se deseja saber o quanto os usuários estão se divertindo. Esse aspecto precisa estar refletido no design da lista de tarefas, o que exemplifica que a criação de cada instância de um teste de usabilidade é análoga à preparação de uma mensagem que precisa ser criada para comunicar aspectos específicos do contexto.

Um aspecto interessante no caso dos jogos é que, após duas ou três sessões iniciais, os problemas são corrigidos e as sessões seguintes são realizadas já com uma nova versão, o que implica que as mensagens atingiram a equipe de desenvolvimento e resultou em mudanças no produto.

É ainda importante observar que os usuários de jogos são mais exigentes do que usuários de outras categorias de software. Se o produto não atinge seus objetivos, ou seja, se lhes proporciona diversão, eles não precisam usá-lo. Por outro lado, se o produto é bom, eles até mesmo aguardam o lançamento de versões futuras. Em ambos os casos, experiências boas ou ruins de um usuário são propagadas para outros com os quais o primeiro compartilha interesse pelos mesmos tipos de jogos. Na perspectiva do *marketing*, os usuários podem ser percebidos como um canal para divulgar jogos. Contudo, o *marketing* não tem nenhum controle sobre as mensagens veiculadas e a única forma de garantir que elas sejam positivas é tornar a experiência dos usuários satisfatória.

Por isso, os testes de *benchmark* são um canal adicional que disponibiliza uma nova perspectiva sobre quem são os usuários, comparando-se ao modelo do ciclo de vida (seção 6.1) que focalizava apenas dados demográficos. Nesse último caso, a ênfase do projeto é definir e desenvolver funções. Não se focaliza a experiência de interação dos usuários como um todo, mas o uso ou interesse em funções específicas. Assim, essa especificidade da criação de jogos se refletiu na criação de um canal adicional para identificar “quem são os usuários” através de uma outra perspectiva.

Entretanto, apesar dessas informações serem importantes para o *marketing*, elas são propagadas para esses agentes através de relatórios. Por outro lado, elas envolvem uma série de aspectos subjetivos, como ocorre também nos testes de usabilidade. No caso dos testes, contudo, a comunicação entre designers e usuários foi aproximada, eliminando-se o engenheiro de usabilidade (através do relatório de

usabilidade) como principal canal de comunicação (Figura 6.3(b)). Seria interessante que, assim como os designers observam os testes de usabilidade, a equipe de *marketing* também observasse o teste de *benchmark*, tornando sua comunicação com os usuários mais eficaz.

6.2.2 Ferramentas para desenvolvedores

Este estudo de caso trata do processo de desenvolvimento de ferramentas para profissionais de desenvolvimento de software.

Os principais agentes envolvidos são os engenheiros de usabilidade, gerentes de programação e os usuários. Uma característica importante que diferencia a equipe deste produto, é que, neste caso específico, os engenheiros de usabilidade têm formação em computação, ou seja, a mesma capacitação técnica que os desenvolvedores.

Nas fases iniciais do processo de criação dessas ferramentas, os engenheiros de usabilidade realizam um teste em um protótipo de papel com o objetivo de analisar modelo conceitual do produto (Figura 6.4).

Ainda nesta etapa, eles realizam um teste de usabilidade da versão anterior (no caso de desenvolvimento de novas versões de um produto existente) e discutem os resultados pessoalmente com os gerentes de programação para assegurar que foram bem entendidos. Esse teste é realizado antes que o gerente de programação comece a redigir a especificação funcional.

Durante a fase de desenvolvimento (vide seção 6.1.1 Descrição do processo, para o modelo do ciclo de vida), testes de usabilidade são realizados de maneira iterativa. Neste projeto, são usados testes *end-to-end*, em que os usuários são solicitados a executarem um projeto do início ao fim. Esse tipo de estudo traz mais contexto para o teste e revela problemas que não foram observados em testes regulares ou cuja importância não foi percebida.

Os resultados dos testes de usabilidade são reportados pelos engenheiros de usabilidade através de encontros pessoais com os gerentes de programação, chamados “encontros 1-1”, relatórios de usabilidade e um banco de dados internos do projeto. Tanto problemas de usabilidade quanto *bugs* são reportados através desse banco de dados e são igualmente tratados, o que implica que a responsabilidade pela correção de cada problema de usabilidade é atribuída a um gerente de programação, diferentemente do que foi reportado nos outros casos. Em função da utilização deste canal, o grupo consegue perceber estatisticamente a relação entre o número de problemas identificados e corrigidos.

São ainda realizadas visitas *in loco*, aproximadamente no meio do ciclo. O objetivo é encontrar padrões de uso que não sejam contemplados pela funcionalidade disponível. Os resultados são documentados e enviados aos gerentes de programação e, principalmente, discutidos com eles.

Os engenheiros de usabilidade deste projeto procuram se comunicar com a equipe de tal forma que as mensagens por eles enviadas causem o máximo de impacto no produto. Assim, eles enfatizam os aspectos positivos inclusive para obter a atenção da equipe quando um aspecto negativo é comunicado.

A aplicação do percurso comunicativo

Percurso - fase preparatória

A tabela 6.3 apresenta uma lista dos agentes envolvidos, identifica com quem cada agente se comunica e os canais e mensagens utilizados nesta comunicação.

Agente	Se comunica com	Canais e mensagens
Engenheiros de usabilidade (EU)	Usuários	em protótipos de papel msg: correspondência entre os conceitos de modelagem e o domínio
		da versão anterior msg: problemas de usabilidade
		<i>end-to-end</i> msg: problemas de usabilidade
	Gerentes de programação	Visitas <i>in loco</i> msg: padrões de uso que não sejam contemplados pela funcionalidade disponível
		Encontros 1-1 msg: resultados dos testes de usabilidade e dados coletados durante as visitas <i>in loco</i>
		Relatórios msg: problemas de usabilidade e sucessos (aspectos positivos) das soluções de design correntes; dados coletados durante as visitas <i>in loco</i>
	Banco de dados interno msg: problemas de usabilidade	

Tabela 6.3: Agentes do processo e as conversações que estabelecem.

A Figura 6.4 apresenta o meta-modelo de comunicação aplicado ao processo de criação de ferramentas para desenvolvedores.

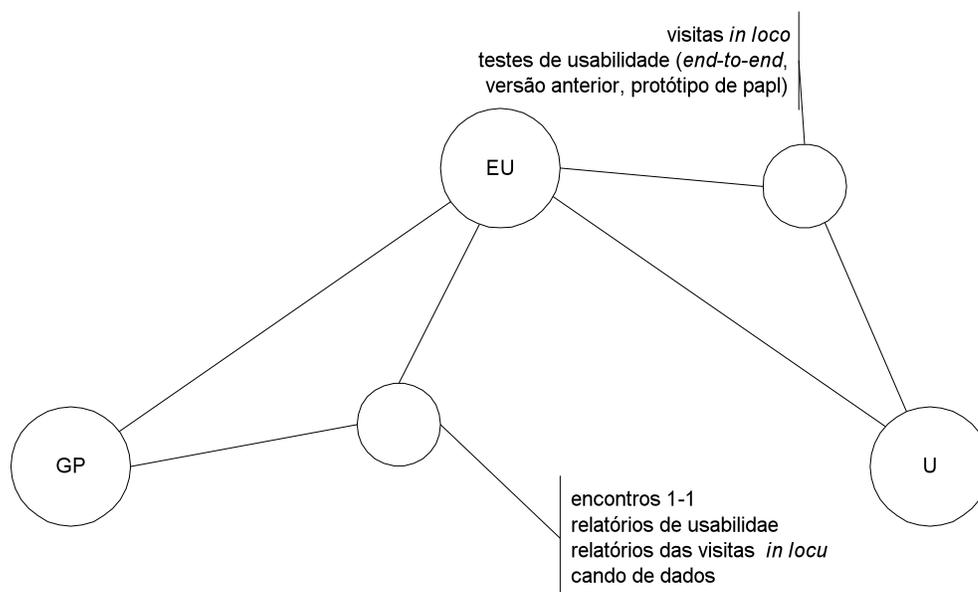


Figura 6.4: Meta-modelo de comunicação aplicado ao processo de criação de jogos.

Percurso – fase de análise

O meta-modelo será percorrido e as comunicações avaliadas.

1. Quem são os usuários.

- a) As visitas *in loco* são um canal para identificar essa questão, que traz a perspectiva da equipe de design para o projeto ao focalizar o contexto de trabalho dos usuários (identifica padrões de uso não contemplados pela funcionalidade do sistema).

As visitas são feitas aproximadamente no meio do ciclo, quando a maior parte da funcionalidade já está definida, mas ainda há espaço para contribuições. Assim, eles podem comparar o que pretendem disponibilizar com o que os usuários realmente precisam.

- b) Quanto à propagação da questão “quem são os usuários”, os dados coletados são documentados pelos engenheiros de usabilidade e enviados aos gerentes de programação. Esses agentes também discutem o conteúdo em questão em encontros pessoais.

Os gerentes de programação coordenam o projeto e efetivamente determinam o que vai ou não ser implementado. Assim, esse conteúdo é

importante para eles, pois explicita padrões de uso não antecipados nas outras fases, o que implica que a funcionalidade precisa ser acrescida de determinadas funções para viabilizar a realização de tarefas relevantes para os usuários.

Os encontros 1-1 entre esses agentes são importantes, pois garantem que a mensagem realmente atingiu o ouvinte (i.e., gerente de programação). Já o relatório permite consultas posteriores, o que é importante para que o gerente de programação possa atualizar a especificação funcional, que por sua vez assegura que a mensagem se propague para o restante da equipe do produto, em especial, para os desenvolvedores e testadores.

Assim, as visitas *in loco* são um canal adicional para identificar “quem são os usuários” através da perspectiva da equipe de design. Essa mensagem é propagada para a equipe com sucesso, através do gerente de programação.

2. Quais os problemas de usabilidade.

- a) Os testes de usabilidade são canais para a identificação dessa questão. O uso de variações (ex., protótipo de papel, *end-to-end*) está relacionado à instanciação de mensagens específicas para o contexto do projeto (incluindo a fase do desenvolvimento, os objetivos do teste, etc.).

Nas fases iniciais do ciclo, aplicam-se os testes com protótipos de papel, que têm por objetivo identificar problemas com os conceitos de modelagem usados e permitir sua correção bem cedo no ciclo. São também realizados testes de usabilidade da versão anterior do produto.

Os testes de usabilidade *end-to-end* explicitam outros problemas que surgem usualmente com o contexto de uso (em que a tarefa é realmente executada), além de evidenciarem problemas geralmente observáveis em testes de usabilidade convencionais. No caso desses últimos, os participantes realizam tarefas envolvendo o uso de funções importantes do produto, mas o conjunto dessas tarefas não necessariamente reflete operações envolvidas na realização de uma atividade dos usuários. Já no teste *end-to-end*, os participantes iniciam e terminam um projeto inteiro e o uso dessas funções está embebido em um contexto em que, por exemplo, o fluxo do diálogo é importante. Assim, esse canal amplia potencialmente o espectro de problemas que podem ser identificados em um teste convencional (sendo mais adequado para identificar a questão “quais”).

- b) Quanto à propagação, o conteúdo produzido com a aplicação de testes de protótipo de papel é bastante relevante para os gerentes de programação. Problemas com o modelo conceitual do produto são identificados bem cedo e comunicados à esses agentes pelos engenheiros de usabilidade através de relatórios e encontros 1-1. O mesmo ocorre com os testes da versão anterior.

Os encontros pessoais entre os gerentes e engenheiros de usabilidade asseguram que a mensagem chega ao destinatário e que seu conteúdo é recebido. As mensagens produzidas pelos canais citados são enviadas suficientemente cedo para que os gerentes incorporem seu conteúdo à primeira versão da especificação funcional, o que implica que ele se propaga para o restante da equipe e evidencia a participação da equipe de design nas fases de planejamento do produto.

Assim como nos casos anteriores, o conteúdo produzido com a aplicação dos testes de usabilidade *end-to-end* é relevante para os gerentes de programação. Também neste caso, os engenheiros de usabilidade enviam relatórios e mantêm encontros 1-1 com os gerentes.

Um importante canal adicional é usado para comunicar os problemas de usabilidade: o banco de dados interno. O uso desse canal tem implicações importantes, pois estende o tratamento dado aos *bugs* aos problemas de usabilidade: a responsabilidade de tratar cada problema reportado pelos engenheiros de usabilidade é atribuída a um gerente de programação que, por sua vez, designa um desenvolvedor para isso.

Não se observa, contudo, integração entre engenheiros de usabilidade e designers. As mensagens sobre “quais os problemas de usabilidade” são igualmente relevantes para os designers, mas não há canais para tornar a comunicação entre esses agentes mais efetiva (comparando-se ao modelo do ciclo de vida, seção 6.1).

Assim, no caso do processo de criação de ferramentas para desenvolvedores, foram adicionados canais para a identificação da questão “quais são os problemas de usabilidade”. A propagação dessa mensagem resulta, em última instância, na modificação do produto para corrigir o problema.

3. As questões “como os problemas de usabilidade se manifestam” e “porque eles ocorrem” serão analisadas conjuntamente.

- a) A identificação das causas dos problemas de usabilidade (“porque”) está relacionada à identificação das expectativas dos usuários (“como”) e do entendimento dos engenheiros de usabilidade sobre aspectos do desenvolvimento do software (ex., comportamento das funções, dificuldades técnicas para implementar algumas soluções de design).

Todos os testes de usabilidade mencionados são canais adequados para identificar “como” os problemas se manifestam, já que permitem a observação detalhada das reações dos usuários.

Por outro lado, os encontros 1-1 entre os engenheiros de usabilidade e gerentes de programação são um canal envolvido na identificação do “porque”, já que, ao mesmo tempo em que comunicam os problemas de usabilidade, eles aprendem quais as prioridades para o projeto, as limitações, entendem as decisões e aspectos já implementados. Esse canal é especialmente eficaz porque os EUs tem a mesma formação técnica que os gerentes de programação.

- b) Quanto à propagação dessa questão, a mesma análise do item 2.(b) pode ser aplicada, ressaltando-se que os encontros 1-1 entre os engenheiros de usabilidade e gerentes de programação são de particular importância para veicular “como os problemas se manifestam” e “porque eles ocorrem”.

Os gerentes de programação determinam as prioridades para correções e alocam desenvolvedores para implementá-las. Assim, à medida que entendem as limitações dos projetos (inclusive técnicas), engenheiros de usabilidade terão mais peso para negociar as correções ao relacionar o custo de implementação para corrigir um problema e o impacto que o mesmo tem na atividade dos usuários (propagação das questões “como” e “porque”). A percepção dos EUs é valorizada, já que eles entendem os aspectos técnicos do desenvolvimento.

As mensagens sobre as questões “como” e “porque” são também relevantes para os designers, mas a propagação para esses agentes não foi alterada em relação ao modelo do ciclo de vida.

Considerações finais

O estudo de caso apresentado exemplifica uma maior integração entre os processos de design e desenvolvimento, o que é evidenciado por um maior participação dos engenheiros de usabilidade nas fases iniciais, uma preocupação em identificar um maior espectro de problemas de usabilidade e uma ênfase na

propagação dos resultados da aplicação dos métodos para a equipe, que efetivamente resulta em melhorias para o produto. A maior cooperação entre esses agentes está relacionada ao fato de que os engenheiros de usabilidade que integram a equipe de criação de ferramentas para desenvolvedores têm uma formação técnica em computação, assim como os gerentes de programação. Os itens a seguir resumem os canais e mensagens utilizados com esse fim:

- a aplicação de testes em protótipos de papel e na versão anterior do produto demonstram uma maior participação dos EUs nas fases iniciais do ciclo. Eles realizam ainda visitas *in loco* para identificar padrões de uso não antecipados pela equipe. Os resultados dessas atividades são incorporados na especificação funcional e propagados para toda a equipe. É importante ressaltar que, nos outros casos analisados, não se observou a participação dos EUs na determinação da funcionalidade do sistema;
- a aplicação de testes *end-to-end* amplia o espectro de problemas observáveis em um teste de usabilidade convencional, o que reflete uma preocupação com a questão “quais”;
- os engenheiros de usabilidade comunicam os problemas aos gerentes de programação de uma maneira eficiente. Canais adequados são usados para negociar as correções (ou seja, para expressar a importância, severidade, impacto de um problema nas atividades dos usuários e comparar esses aspectos ao custo de correção). Quando os EUs reportam um problema de usabilidade utilizando o banco de dados interno, a responsabilidade pela correção do problema é automaticamente atribuída ao gerente de programação responsável pela funcionalidade relacionada ao problema. O gerente, por sua vez, designa um desenvolvedor para corrigi-lo, o que implica a propagação do problema para a equipe e evidencia um maior impacto na usabilidade no produto.

6.2.3 Considerações finais

A análise dos casos apresentados nesta seção, sob a perspectiva da comunicação segundo o modelo fractal proposto neste trabalho, evidencia que tanto a comunicação com os usuários (aplicação de métodos e técnicas de design de interfaces e avaliação de usabilidade) quanto a propagação para a equipe precisa considerar as especificidades do projeto. Os casos apresentados nesta seção

demonstram que alguns grupos percebem particularidades no processo de criação de alguns produtos e propõem soluções específicas para o contexto.

Um exemplo é o caso dos jogos, em que se criou um canal adicional para entender as preferências dos usuários ao interagirem com um jogo (Figura 6.2: teste *benchmark* como canal adicional para identificar a questão “quem”). Considerando que os usuários de jogos trocam mensagens entre si referentes à sua experiência de interação com determinados jogos, a equipe de *marketing* precisa assegurar que essas mensagens serão positivas. Esses agentes utilizam os resultados dos testes *benchmark* na fase de planejamento, a fim de identificar o que precisa ser valorizado ou melhorado no produto.

No processo de criação de ferramentas para desenvolvedores foi também criado um canal para a identificação da questão “quem”. Engenheiros de usabilidade fazem visitas *in loco* para identificar padrões de uso não antecipados na especificação funcional.

Uma grande integração entre designers e engenheiros de usabilidade pode ser observada para o processo de criação de jogos. O objetivo das atividades de usabilidade é encontrar problemas que afetem a diversão dos usuários e é fundamental entender suas reações, o contexto (cenário) em que ocorre um problema de usabilidade. Descrever tudo isso através de relatórios é difícil. Os canais encontrados tornaram a comunicação mais eficiente. Os designers observam as sessões e recebem mensagens dos usuários sobre os problemas de usabilidade, o que não apenas elimina as dificuldades que surgem com o uso apenas dos relatórios como também aumenta a possibilidade de identificação de mais problemas, já que os designers podem perceber aspectos não observados pelos engenheiros. Os agentes em questão discutem os problemas observados e as soluções no próprio laboratório.

No caso do processo de criação de ferramentas para desenvolvedores, existe uma maior integração entre os engenheiros de usabilidade e gerentes de programação. Esses agentes compartilham uma linguagem e entendimento sobre questões técnicas (como foi dito anteriormente, no caso deste grupo, os engenheiros de usabilidade têm formação em computação). Por outro lado, os gerentes de programação são a ponte entre o *marketing* e o resto da equipe. Eles têm uma visão mais abrangente do projeto (em relação aos desenvolvedores, cujo foco é mais dirigido às funções que precisam implementar) e precisam pensar na experiência do consumidor. Um aspecto adicional é que o domínio de aplicação dos produtos é o desenvolvimento de software, ou seja, um domínio conhecido por ambos.

Talvez pelas razões enumeradas acima, encontros 1-1 que implicam em uma comunicação intensiva e dinâmica entre os agentes, sejam tão adequados para esse projeto. Os agentes em questão compartilham uma linguagem. A comunicação entre eles é propagada para desenvolvedores e testadores através da especificação funcional.

O uso de um banco de dados interno evidencia a conscientização da equipe sobre questões de usabilidade, já que os problemas de usabilidade reportados recebem o mesmo tratamento que os *bugs*. Isso implica que as mensagens sobre usabilidade se propagam por toda equipe resultando em mudanças no software. O uso dos canais mencionados reflete a percepção de oportunidades criadas no contexto do projeto.

6.3 Meta-modelo no nível 2 – o design de mensagens

Os estudos de caso anteriores tratam da aplicação do percurso comunicativo ao processo de design/desenvolvimento do produto, focalizando os atos comunicativos envolvidos no design da interface. Nesta seção, o método foi aplicado ao processo de design de um teste de usabilidade.

Uma das características da aplicação do modelo fractal de comunicação (Capítulo 4) a IHC é a possibilidade de aplicá-lo em diversos níveis. Do ponto de vista de comunicação, ele pode ser aplicado no nível de planejamento e execução da aplicação de um método de usabilidade.

Na perspectiva abordada nos estudos de caso anteriores, o teste pode ser visto como canal através do qual engenheiros de usabilidade e usuários se comunicam para determinar os problemas de usabilidade com um dado design. Entretanto, a eficácia da comunicação em todo o processo está relacionada ao sucesso de cada ato comunicativo. Um teste de usabilidade, por exemplo, envolve a preparação de materiais como a lista de tarefas, protótipos, a realização de um piloto, e o redesign do próprio teste com base nos resultados do piloto. Analisando detalhadamente, cada um desses materiais é um canal que veicula uma mensagem (ex., a lista de tarefas comunica aos usuários em que aspectos os engenheiros precisam que os primeiros focalizem), o que corresponde a um nível mais interno no fractal.

6.3.1 Cenário de aplicação do método

O objetivo do teste em questão era avaliar a interface do *help* de cinco aplicações Web da ORG. As aplicações serão referenciadas neste texto como A_i , $1 \leq i \leq 5$. A interface do *help* é comum à todas as aplicações.

Os agentes envolvidos em conversações para a preparação do teste foram: o engenheiro de usabilidade responsável pelo teste (EU_R), os engenheiros de usabilidade responsáveis pelas aplicações testadas (EU_{A_i}), o designer do *help* que também acumula a função de gerente do projeto e os participantes do teste.

O designer do *help* foi contactado pelo EU_R via *email* (Figura 6.5, comunicação entre designer e EU_R) para definir os principais objetivos do teste, que envolviam avaliar os seguintes aspectos da interface do *help*:

- como os usuários iniciam e finalizam o *help*. No caso de aplicações Web, muitos usuários confundem o *help* da aplicação com o *help* do *browser*. Há também algumas inconsistências entre as aplicações quanto ao local da interface em que se inicia o *help*;
- a navegação;
- o uso do dispositivo de pesquisa (se os usuários usam uma linguagem mais próxima da natural ou palavras-chave);
- como eles mudam o foco do *help* para a aplicação e retornam ao estado inicial;
- se eles usam (ou pelo menos entendem) todas as facilidades da interface;
- se eles são capazes de retornar a um ponto já previamente visitado no *help* (ex., achar um dado conteúdo para fazer e desfazer uma ação);
- quais são as suas expectativas.

Todos os relatórios de usabilidade produzidos na ORG são armazenados e disponibilizados através de um *site* interno. Como versões anteriores da interface em questão já haviam sido testadas, a primeira tarefa realizada no design do teste foi recuperar os relatórios relacionados a esses testes.

Tais relatórios funcionaram como um canal de comunicação entre o engenheiro de usabilidade responsável pela preparação do teste em questão e os engenheiros

envolvidos nos testes anteriores (Figura 6.5, comunicação EU_R e EU_{ant}). Embora não envolvessem as mesmas aplicações, tais testes também focalizavam principalmente a interface do *help* e não seu conteúdo.

É importante notar que, para cada uma das aplicações envolvidas no teste, há um engenheiro de usabilidade responsável. A interface do *help* é, contudo, criada e implementada independentemente da aplicação e as equipes de cada produto desenvolvem apenas seu conteúdo¹⁸.

O EU_R contactou, via *email*, os engenheiros das aplicações (Figura 6.5, EU_i são canais de comunicação entre as aplicações A_i e EU_R) e solicitou informações sobre:

- perfil dos usuários. Essa descrição envolve especificidades que caracterizam os usuários de cada aplicação. No entanto, seria difícil conseguir um número suficiente de participantes que satisfizessem, simultaneamente, características de todos os grupos. Assim, apenas as restrições comuns a todas as aplicações foram mantidas e a principal característica dos participantes recrutados é que eles se enquadravam no perfil de iniciantes e intermediários no uso da Web;
- tarefas ou cenários que eles considerassem suficientemente difíceis para motivar os participantes a consultar os arquivos de *help*, já que, tipicamente, eles evitam tal auxílio.

Outra conversação importante foi a que o EU_R estabeleceu com as próprias aplicações envolvidas (Figura 6.5, comunicação entre EU_R e A_i utilizando a interface como canal). Eventualmente, novas perguntas sobre as mesmas foram dirigidas aos outros engenheiros.

Com base em todas essas conversações, foram elaborados (1) uma lista de tarefas envolvendo as cinco aplicações, (2) um questionário sobre a experiência prévia dos usuários, (3) um questionário sobre a experiência de utilização do *help* durante a sessão, (4) arquivos usados como entrada para o aplicativo que os engenheiros de usabilidade da ORG usam para registrar a interação (ex. registro dos elementos da interface usados por cada usuário, comentários, etc.). Todos esses materiais foram produzidos iterativamente.

Dessa forma, pode-se perceber que, em uma determinada perspectiva, o teste de usabilidade é um canal que veicula mensagens sobre problemas de usabilidade da

¹⁸ Apesar da interface do *help* ser comum às aplicações, alguns projetos incluem elementos no hipertexto que, à medida que se mostram bem sucedidos, são incorporados pelos outros e se tornam padrões.

interface testada. No entanto, vários canais são empregados na sua realização, o que caracteriza um nível mais interno no fractal.

Deve-se observar que a redação do texto da lista de tarefas foi bastante cuidadosa para comunicar com clareza a tarefa evitando, no entanto, usar termos do *help* e, conseqüentemente, influenciar os usuários. A comunicação entre EU_R e os materiais emergentes estão explicitadas na Figura 6.5 (canal entre EU_R e o teste de usabilidade).

O local, horários e datas das sessões foram comunicados ao designer e EU_{Ai} . Muitos deles compareceram às sessões e observaram os usuários (Figura 6.5, *email* como canal entre EU_R e designers/ EU_{Ai}).

O laboratório é dividido em duas partes: uma destinada aos usuários, em que se encontram o computador por eles usado, câmeras de vídeo que capturam vários ângulos da interação (que podem ser modificados do outro lado), microfones e alto-falantes; o outro lado, destinado ao engenheiro de usabilidade responsável pelo teste, possui um monitor com equipamento de gravação de vídeo que exhibe a tela do participante, um aparelho de TV em que a mesma tela pode ser observada, uma lousa e um computador no qual é executado um aplicativo para auxiliar o EU a registrar informações sobre a sessão. O EU_R prepara arquivos de entrada para o aplicativo que refletem os objetivos do teste e as aplicações envolvidas. Esses elementos não serão detalhados, pois isso implicaria analisar um nível ainda mais interno no fractal. As duas partes do laboratório são separadas por um espelho *one way*.

Membros da equipe e pessoas interessadas podem observar o teste do lado dos engenheiros.

No teste em questão, a lousa foi usada para resumir os resultados. Foi criada uma tabela com entradas para os participantes e aspectos que se desejava observar.

Durante o teste, os usuários são orientados a “repetir seus pensamentos em voz alta” (*Thinking aloud*, (Nielsen, 1993)). O EU_R pode falar com os usuários utilizando um microfone.

6.3.2 Aplicação do percurso comunicativo

A aplicação do método para o caso de design de uma mensagem (nível mais interno no fractal) é um pouco diferente dos casos anteriores.

A análise do processo envolve avaliar a suficiência de canais e adequação das mensagens para responder às questões sobre “quem são os usuários”, “quais são os problemas de usabilidade”, “como eles se manifestam” e “porque eles ocorrem”. A análise aplicada ao design de uma mensagem (1) exemplifica o uso de um canal mais

apropriado para responder a um subconjunto dessas questões (ex. testes de usabilidade para “como”) e (2) evidencia como as outras questões complementam e influenciam o design da mensagem (ex. identificar quem são usuários é importante para recrutar participantes para o teste).

Percurso - fase preparatória

A tabela 6.4 apresenta uma lista dos agentes envolvidos, identifica com quem cada agente se comunica e os canais e mensagens utilizados nesta comunicação.

Agentes	Se comunicam com	Canais e mensagens
Engenheiro de usabilidade responsável - EU_R	Designer	<i>Email</i> msg: objetivos do teste
		Resumo na lousa do laboratório msg: quais os elementos da interface utilizados por cada participante
	Engenheiros de usabilidade das aplicações - EU_{Ai}	<i>Email</i> msg: quem são os usuários, quais os cenários e tarefas difíceis para cada aplicação
		Resumo na lousa do laboratório msg: quais os elementos da interface utilizados por cada participante
Aplicação _i	Interface msg: quais as funções do produto e como usá-las	
EU_{ant} (testes das versões anteriores do <i>help</i>)	Relatórios de usabilidade msg: descrição dos testes anteriores, relação dos problemas de usabilidade identificados (questão “quais”)	

Tabela 6.4: Agentes envolvidos no design do teste do *help* e as conversações que estabelecem.

Engenheiro de usabilidade responsável - EU _R	Usuários	Teste de usabilidade msg: problemas de usabilidade	Lista de tarefas msg: em que o EU quer que os usuários focalizem
			Monitor replicando a tela dos usuários msg: enfatizar cada ação que eles executam
			Questionário msg: experiência prévia dos usuários com Internet, aplicativos, etc.
			Questionário msg: impressões de uso do <i>help</i> durante a sessão
			Aplicação _i msg: funções disponíveis para realizar as tarefas e como elas podem ser usadas
			Aplicação para registro do teste msg: ações executadas pelos usuários
			<i>Thinking aloud</i> msg: expectativas dos usuários (eles podem ser solicitados a esclarecer alguns aspectos)
Usuários	Designer	Teste de usabilidade msg: problemas de usabilidade	
	EU _{Ai}	Teste de usabilidade msg: problemas de usabilidade	
	Aplicação _i	Interface msg: quais as funções do produto e como usá-las	

Tabela 6.4: Continuação.

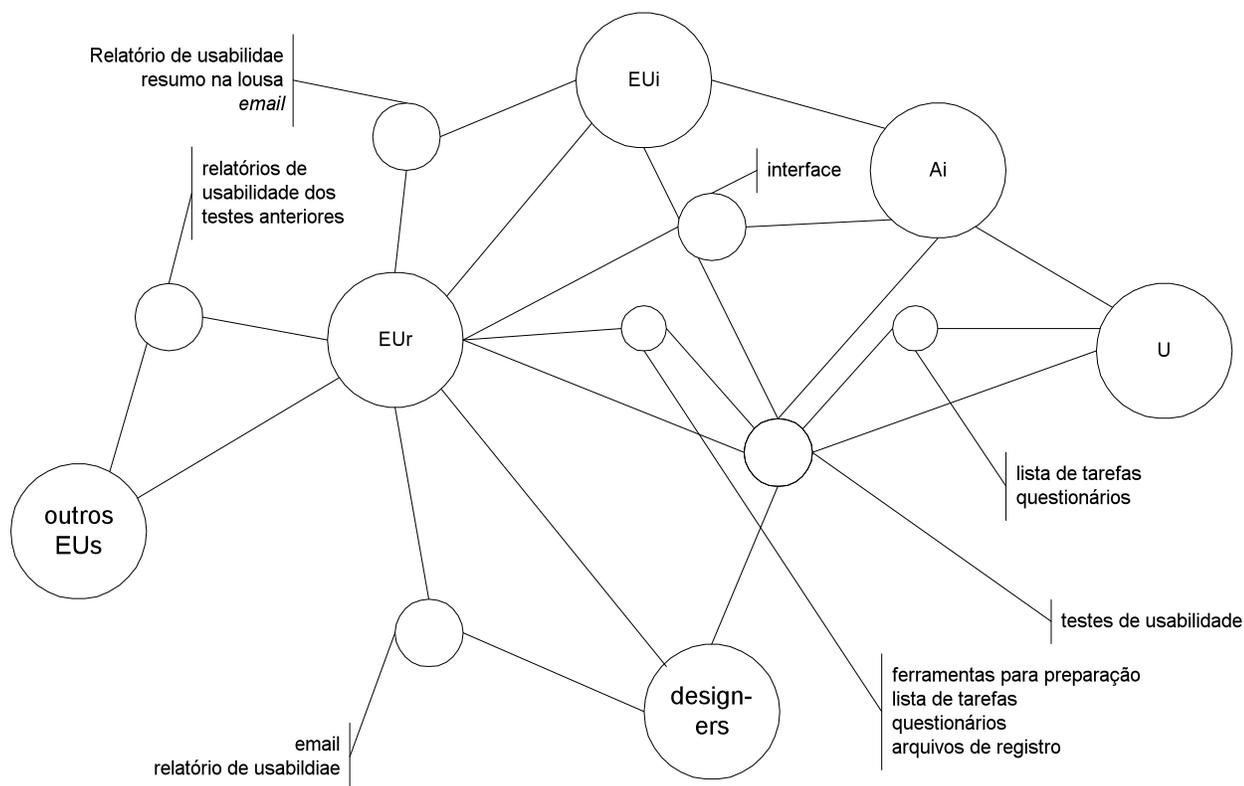


Figura 6.5: O meta-modelo de comunicação aplicado ao processo de design do teste de *help* das aplicações citadas.

Percurso - fase de análise

Nesta fase, percorre-se o meta-modelo verificando-se a comunicação entre os agentes. Conforme foi ressaltado anteriormente, a análise para algumas questões não se refere tanto à suficiência dos canais e mensagens, mas sobre como são importantes (complementam) para o design do teste.

1. Quem são os usuários.

- a) Essa informação foi obtida dos engenheiros de usabilidade das aplicações envolvidas. Trata-se da mesma descrição que eles usam para recrutar participantes para os seus próprios testes.
- b) Quanto à relevância do conteúdo, essa questão tem implicações importantes dependendo do teste de usabilidade. Para aplicações financeiras, por exemplo, o público alvo (ou sua grande maioria) se enquadra num perfil de pessoas que lidam com investimentos. Recrutar participantes com essa

característica tem implicações não apenas na sua motivação, mas no próprio sucesso com que realizam tarefas (ex. questões de terminologia podem ser um problema para usuários leigos naquele domínio, mas totalmente adequadas para usuários proficientes).

Já no caso do teste do *help*, que envolve um conjunto de aplicações, é mais difícil encontrar usuários que se enquadrem, simultaneamente, em vários perfis. Por outro lado, o teste focaliza em aspectos bem mais gerais (ex. como iniciar e finalizar o *help*, como usar o dispositivo de busca) e os problemas discutidos acima não têm tanto impacto. Por essa razão, as restrições mais particulares foram eliminadas e foram recrutados participantes iniciantes ou intermediários no uso de Internet.

Quanto aos agentes atingidos, a informação foi rapidamente obtida pelo EU_R, assim que o mesmo a requisitou aos outros engenheiros. O canal utilizado na conversação entre esses agentes foi o *email*, que se mostrou bastante adequado. Faz parte da cultura da empresa ORG responder os *emails* com a maior agilidade possível. Todas as solicitações foram respondidas prontamente.

2. Quais são os problemas de usabilidade.

- a) Durante o design do teste, foram recuperados relatórios de usabilidade das versões anteriores do *help*. Eles foram analisados detalhadamente e os problemas reportados foram foco de observação cuidadosa no design do teste em análise.

Vários canais são usados durante os testes de usabilidade na ORG: lista de tarefas, questionários, monitor replicando a tela dos usuários, a técnica de *thinking aloud* e a própria aplicação (vide Tabela 6.2).

Poucos problemas de usabilidade foram identificados na aplicação do teste em questão. A maioria e também os mais críticos estavam associados ao conteúdo dos arquivos de *help*, apesar da avaliação do conteúdo não ser objetivo inicial do teste.

Durante a análise do processo (seção 6.1), foi discutido que os testes de usabilidade evidenciam parte dos problemas existentes e que outros métodos devem complementar essa informação. A análise aqui, contudo, refere-se à suficiência dos canais considerando o escopo do teste de usabilidade, delimitado pela lista de tarefas que determina o espaço de atuação dos usuários e pelos objetivos do teste. Considerando esses aspectos, o uso dos

canais mencionados foi suficiente para identificar “quais os problemas de usabilidade”.

- b) Quanto à propagação das mensagens, deve-se verificar se o conteúdo em questão é relevante para todos os agentes envolvidos com usabilidade das aplicações e com a criação do *help*.

Os problemas identificados foram reportados através de *emails* os resumindo e relatórios de usabilidade, que foram enviados para os EU_{Ai} e para o designer do *help*.

O relatório de usabilidade reportou de maneira detalhada todos os problemas identificados, mesmo aqueles relacionados ao conteúdo das aplicações. Deve-se observar que tais problemas foram mais críticos para algumas aplicações.

O resumo no quadro do laboratório foi bastante útil para promover discussões e o acompanhamento das sessões por parte dos agentes que foram observar os testes (Figura 6.5, comunicação entre EU_R e EU_{Ai}).

3. Como os problemas de usabilidade se manifestam.

- a) O teste de usabilidade mostrou-se um canal bastante apropriado para se responder à essa questão.

Durante o teste, pode-se observar as reações dos usuários através de um espelho *one way*, com câmeras focalizando diferentes ângulos da interação. Os participantes são também orientados para “repetir em voz alta seus pensamentos”. O EU_R pode acompanhar todas as suas ações em um monitor e as pessoas que estiverem observando, podem acompanhá-las através de uma TV.

Muitas vezes, as mensagens veiculadas pelos canais são redundantes (ex., usuário diz em voz alta as palavras que está procurando ao mesmo tempo em que as digita). Outras vezes, são complementares (ex., usuário diz palavras que descrevam o tópico que está procurando ao mesmo tempo em que navega em busca do mesmo).

Assim, esses diversos canais viabilizaram a observação detalhada do procedimento dos usuários.

- b) Sobre a propagação, ou seja, relevância do conteúdo, agentes atingidos e adequação dos canais.

Entender as expectativas e procedimentos dos usuários é bastante relevante para os EU_{Ai} e para o designer do *help*. Parte desses agentes compareceu às sessões e pôde observar os participantes (o teste atuou como canal entre eles), o que é importante, já que diversos canais são empregados facilitando a observação dessa questão.

Esses agentes receberam também um relatório de usabilidade com a descrição detalhada de “como” os problemas se manifestam, o que é importante para eles e particularmente útil para os engenheiros de usabilidade que não compareceram às sessões e principalmente para os aqueles cuja aplicação apresentou problemas com o conteúdo.

O EU_R enviou *email* para os EU_{Ai} e para o designer do *help* colocando-se à disposição para esclarecer dúvidas sobre o relatório, mas a conversação prosseguiu apenas com o designer.

Um terceiro canal utilizado foi um *email* resumido, enviado para os EU_{Ai} e para o designer.

O uso de *email* e relatórios de usabilidade como os únicos canais de comunicação entre EU_R e os outros agentes (EU_{Ai} e designer) não estimula que os agentes da comunicação permaneçam em comunicação, nem garantem que a mensagem foi “consumida” (ou seja, lida, entendida). Mais difícil ainda é determinar se a mensagem foi propagada a outros agentes (ex., gerentes de programação), implicando mudanças no produto.

4. Por que os problemas ocorrem?

a) Para o teste em questão, apesar do EU_R não estar diretamente ligado às aplicações envolvidas, não houve dificuldades na determinação do “porque”. As causas dos problemas mais críticos foram:

- as palavras usadas pelos participantes para procurar tópicos eram diferentes das palavras-chave usadas na implementação do *help*;
- o conteúdo do *help* estava inconsistente com a aplicação, não refletia a seqüência de passos necessária para realizar a tarefa.

O EU_R determinou as causas dos problemas baseando-se na observação do procedimento dos usuários e na sua própria interação com os aplicativos e *help* associado.

Assim, os canais disponíveis foram suficientes para determinar as causas dos problemas identificados.

- b) Sobre a propagação, o conteúdo da questão “porque” é relevante para os EU_{Ai} e para o designer.

Um relatório de usabilidade e *email* resumido as causas foram enviados aos EU_{Ai} e ao designer.

Como no caso do teste em questão a equipe que produz o conteúdo não esteve diretamente envolvida, os EU_{Ai} são o canal entre eles e o engenheiro de usabilidade responsável pelo teste. Não é possível determinar, no entanto, se alguma mensagem sobre a questão corrente atingiu a equipe relacionada ao conteúdo.

Em alguns casos, sabe-se apenas que a mensagem (veiculada através do relatório de usabilidade) chegou a um EU_{Ai} , mas, exceto no caso em que tal mensagem motiva discussões posteriores, como ocorreu com o designer, não é possível determinar se seu conteúdo foi ou não lido e compreendido.

Alguns dos engenheiros de usabilidade responsáveis pelas aplicações e o designer compareceram à maior parte das sessões, trazendo oportunidades para discutir os problemas identificados, inclusive suas causas. O uso desse canal presencial estabelece maior contacto entre os agentes mencionados e EU_R e entre eles e os usuários.

6.3.3 Considerações finais

O estudo de caso apresentado em que se exemplifica a aplicação do percurso comunicativo ao design de um teste de usabilidade evidencia aspectos importantes.

Assim como o design de cada produto envolve a criação de uma mensagem única (i.e., a interface de um produto é especialmente criada para ele), na perspectiva adotada por este trabalho, a aplicação dos diferentes métodos de usabilidade apresentados na literatura envolve a criação de mensagens com objetivos específicos.

Os métodos correspondem a canais mais ou menos adequados para veicular determinados conteúdos e as mensagens correspondem à instâncias dos mesmos, criadas especificamente para um contexto de comunicação (contexto de um projeto). Os benefícios que se pretende alcançar para a visão mais macroscópica do processo de design podem ser atingidos para cada ato de comunicação desse processo.

Ainda que um método seja mais adequado a um determinado conteúdo (ex., determinar “como” os problemas de usabilidade se manifestam), a criação das mensagens envolve outros conteúdos (ex., saber quem são os usuários é importante para recrutar os participantes corretos). Assim, pode-se perceber, com a aplicação de

um método, a importância de se identificar agentes que possam comunicar um determinado conteúdo.

6.4 Conclusões e discussões finais

A aplicação do percurso comunicativo ao processo de criação de software foi capaz de evidenciar deficiências na comunicação que têm implicações na usabilidade do produto (conforme demonstrado na seção 6.1). De acordo com a perspectiva do método, duas conversações são fundamentais:

- a comunicação entre os designers/engenheiros de usabilidade e os usuários;
- a propagação dessa comunicação para a equipe do produto.

Assim, o impacto das atividades de usabilidade não depende apenas da abrangência ou eficiência de um método de avaliação de usabilidade, mas também da maneira como os resultados de sua aplicação se propagam pela equipe, resultando em modificações no produto.

As soluções para corrigir as deficiências identificadas durante o percurso no meta-modelo envolvem a criação de mensagens e/ou canais. Entretanto, deve-se observar que cada projeto apresenta oportunidades e obstáculos para a criação desses elementos (ex., os objetivos do produto, características da equipe e limitações do desenvolvimento). O percurso comunicativo auxilia a identificar tais aspectos.

O uso eficiente dos canais implica que mensagens precisam ser criadas especificamente para cada ato de comunicação. No método de inspeção proposto, os métodos e técnicas de design de interfaces e avaliação de usabilidade são percebidos como canais. Aplicá-los implica criar instâncias específicas para o contexto em que serão usados, o que demanda um entendimento profundo do método e do projeto em questão.

A análise do processo através da perspectiva da comunicação abrange a questão sobre como os canais se complementam, ou seja, no contexto de design, como os diferentes métodos de design e avaliação de usabilidade podem ser combinados para responder às diferentes questões como “quem são os usuários”, “quais são os problemas de usabilidade”, por exemplo.

Todos os aspectos discutidos foram demonstrados através da aplicação do percurso comunicativo aos casos apresentados.

Pode-se perceber, através da análise realizada que, na ORG, o desenvolvimento é ainda centrado no produto, não nos usuários, uma vez que focaliza a implementação de funções, mas não a forma como elas estão estruturadas no conjunto da aplicação. Isso é evidenciado ainda pela integração entre desenvolvedores e testadores, e desses com os gerentes de programação. Os desenvolvedores implementam o código seguindo a especificação e os testadores verificam se o código produzido funciona adequadamente, retornando os problemas identificados para os primeiros, que os corrigem continuamente (esses agentes trabalham em parceria).

A integração entre esses agentes não se observa entre designers e engenheiros de usabilidade. Apesar de serem realizados testes iterativamente, eles não atuam de maneira tão integrada. É relativamente usual que designers observem as sessões dos testes de usabilidade, isso não ocorre de maneira sistemática. O principal canal de comunicação entre os agentes é o relatório de usabilidade, que não assegura que a mensagem irá causar impacto no processo. Ao contrário. Quando ocorre tal impacto, ele é casual, pois depende exclusivamente do envolvimento e interesse das pessoas envolvidas em uma situação particular (i.e., do esforço pessoal de um engenheiro de usabilidade ou designer em promover a usabilidade do produto e identificar mecanismos para tanto).

Certamente, alguns processos percebem a falta de integração entre designers e engenheiros de usabilidade e as conseqüências disso no processo de design. No caso dos jogos, por exemplo, designers e engenheiros de usabilidade trabalham com mais integração. O design do teste de usabilidade implica a participação (não casual, mas oficial) dos designers, que observam as sessões e discutem com os engenheiros de usabilidade os problemas identificados e as soluções os mesmos. A comunicação entre designers e engenheiros de usabilidade foi aproximada, bem como a comunicação entre os primeiros e os usuários (vide Figura 6.3).

Tornar o modelo do ciclo de vida mais centrado nos usuários implica tornar a comunicação entre designers e engenheiros de usabilidade mais intensiva e eficaz. Entretanto, as soluções encontradas precisam ser aplicadas de maneira sistemática, como ocorre com desenvolvedores e testadores, que trabalham lado a lado.

O enfoque centrado no software do modelo do ciclo de vida da ORG é evidenciado também pela integração entre a equipe de desenvolvimento e os gerentes de programação. Entre esses três agentes a comunicação ocorre de forma efetiva e eficaz.

Os gerentes de programação funcionam como um canal entre o *marketing* e a equipe de desenvolvimento. Ao mesmo tempo em que têm um entendimento sobre os detalhes técnicos relacionados à cada função, os gerentes de programação têm uma visão mais abrangente sobre produto, mais ampla, ao contrário dos desenvolvedores que focalizam as funções. O fato de a ORG ter um agente do processo que atue como canal entre esses grupos reflete a importância que a empresa confere a eles. O *marketing* define em que investir e os gerentes de programação traduzem isso para a linguagem dos desenvolvedores. Os gerentes coordenam as comunicações e o desenvolvimento.

No processo de criação de ferramentas para desenvolvimento, a atuação dos gerentes de programação como agentes críticos no processo foi percebida pelos engenheiros de usabilidade e a interação entre esses agentes é mais intensiva. No caso do produto em questão, os relatórios de usabilidade são apenas um dos canais que os EUs utilizam para se comunicarem com os gerentes. Esses agentes também mantêm encontros 1-1, em que discutem e negociam os diversos problemas de usabilidade identificados.

No caso desse produto, a comunicação entre os gerentes de programação e os engenheiros de usabilidade é bastante eficaz. Deve-se observar, contudo, que no caso do desenvolvimento de ferramentas para desenvolvedores, isso pode ser um reflexo da formação técnica em computação dos engenheiros de usabilidade, o que não é um fator comum em outros grupos. Isso aproxima os agentes em questão, já que os gerentes, ao mesmo tempo em que entendem dos detalhes técnicos, têm uma visão mais ampla do produto, bem como os engenheiros de usabilidade que, nesse caso particular, também entendem dos aspectos técnicos. Cabe perguntar porque razão, na ORG, apenas os EUs envolvidos na criação de ferramentas para desenvolvedores têm, necessariamente, formação técnica.

Um outro aspecto explicitado com a aplicação do percurso comunicativo ao modelo do ciclo de vida da ORG e que evidencia o enfoque no desenvolvimento de software é a pouca participação da equipe de design nas fases iniciais do processo. Contudo, a influência e participação dos desenvolvedores nessas etapas é notada. Eles são valorizados, pois sabem o que é tecnicamente possível de ser implementado.

Esse ponto merece destaque, pois evidencia que a perspectiva dos usuários ao interagirem com todo o software (e não com funções) não é incorporada de maneira sistemática ao processo, através da participação de designers e engenheiros de usabilidade.

Alguns grupos perceberam esse aspecto e desenvolveram soluções para tornar a participação desses agentes mais efetiva.

No caso dos jogos, foi criado um canal adicional para responder à questão “quem são os usuários”. Os testes de *benchmark* focalizam essa questão por um ângulo diferente daquele tipicamente utilizado pelo *marketing* (relacionado a dados demográficos). Os dados coletados referem-se à experiência de interação dos usuários com o produto, às suas preferências. A preocupação do *marketing* em utilizar esses dados reflete, contudo, uma atitude mais exigente dos usuários de jogos. Para o jogo, o “enjoyment” é fundamental. Para as aplicativos *desktop*, por exemplo, existe a cultura do “sério”, onde não há lugar para o *enjoyment*, quando deveria haver do ponto de vista da interação e interface. É o que aponta a Brenda Laurel [1990] quando se refere à “engajamento prazeroso” ao falar sobre a interface de maneira geral.

O processo de criação de ferramentas para desenvolvedores utiliza também um canal adicional (se comparado ao modelo do ciclo de vida do produto, seção 6.1) para responder à questão “quem são os usuários”. Engenheiros de usabilidade fazem visitas *in loco* para determinar padrões de uso não antecipados pela equipe de criação do produto. Nesse projeto em particular, procurou-se entender os usuários através de outra perspectiva (ou seja, qual o contexto em que utilizam o software, para que, de que maneira). Deve-se observar que o domínio da aplicação é o mesmo domínio com o qual trabalham engenheiros de usabilidade, desenvolvedores, testadores, gerentes de programação.

A análise do modelo do ciclo de vida explicitou ainda que existem uma série de canais disponíveis, como os beta testes, que poderiam fornecer dados sobre “quem são os usuários”, permitindo que se incorporasse outras perspectivas, além daquela praticada pelo *marketing*, ao desenvolvimento dos produtos.

Um outro aspecto importante e que foi explicitado ao longo de todos os casos apresentados é a importância da propagação eficiente dos resultados da aplicação dos métodos de usabilidade, resultando em mudanças no produto.

No modelo do ciclo de vida os principais canais utilizados para esse fim são os relatórios de usabilidade¹⁹ e *email* resumindo os problemas identificados. Esses canais não garantem que a comunicação se efetivou (que os agentes estabeleceram uma conexão psicológica e um entendimento mútuo sobre aspectos relevantes para ambos), como foi visto no estudo de caso sobre o design de um teste de usabilidade (seção 6.3).

¹⁹ O relatório de usabilidade é circulado via *email* ou disponibilizado eletronicamente.

É fundamental que se identificar canais adicionais para realizar essa comunicação, já que na ORG o volume de documentação produzida é tão grande. Além disso, todas as comunicações ocorrem via *email*. Esse canal é bastante adequado para uma série de mensagens, como solicitação de informações sucintas, agendamento de compromissos, etc. Entretanto, quando a comunicação exige uma reação dos destinatários, é importante assegurar que a mensagem foi recebida. Ainda mais importante é assegurar que os agentes permaneçam em comunicação, já que, no contexto em questão, alterações para corrigir problemas de usabilidade implicam negociações e, portanto, agentes se alternando em turnos como destinatários e remetentes.

A análise apontou ainda que há canais adicionais disponíveis para a identificação de “quais” são os problemas de usabilidade e que, contudo, não são efetivamente utilizados.

Cabe observar que a aplicação do percurso comunicativo ao design de um teste de usabilidade demonstra que um teste, assim como a interface, é objeto de design. A eficácia da comunicação no processo depende da elaboração de cada mensagem.

Capítulo 7

Discussão e conclusões

No capítulo 2, alguns dos processos de design de software foram revistos. Tipicamente, eles consistem de três etapas. A primeira refere-se ao estudo sobre os usuários e análise de tarefas. A seguir, objetivos mensuráveis de usabilidade são especificados. Essa especificação é usada como mecanismo de controle em um processo iterativo, em que um protótipo é produzido, avaliado quanto à usabilidade e refinado (*redesigned*), até que os objetivos estabelecidos sejam alcançados.

Esses processos estruturam a aplicação de métodos e técnicas de design e avaliação de usabilidade do produto. Contudo, o contexto de cada projeto apresenta oportunidades ou obstáculos para a aplicação de diferentes métodos, o que dificulta a proposição de um processo de design genérico: ou eles são pouco detalhados, ou se ressalta a necessidade de adaptá-los às especificidades de cada projeto.

É importante observar também que a aplicação de cada método exige uma preparação rigorosa. Disso depende a qualidade do processo de design como um todo.

Além disso, para que o processo de design seja realmente iterativo e empírico, é necessário haver uma estratégia de implementação que permita que os dados obtidos com as avaliações resultem em modificações rápidas e efetivas.

Por outro lado, os processos de desenvolvimento da Engenharia de Software, também revistos no capítulo 2, são centrados no projeto e desenvolvimento do software. Grande parte dos recursos de desenvolvimento e manutenção de um sistema é dedicada a correções por falta do entendimento sobre os usuários, suas necessidades e contexto em que utilizam o aplicativo. Assim, esses processos perdem com a falta de integração.

Os vários agentes envolvidos no processo de criação de um produto participam de uma negociação que reflete a resposta da organização como um todo às demandas e interesses dos usuários. A equipe de design precisa conhecer as limitações do

projeto (ex., técnicas, de cronograma e orçamento) para propor soluções factíveis. Por outro lado, ela precisa propagar de maneira eficaz as necessidades dos usuários, os problemas de usabilidade identificados e as prioridades para corrigi-los, etc. A própria capacidade da organização em gerenciar as habilidades e atuações de cada grupo determina a qualidade do produto final.

Analisar esses processos pela perspectiva da comunicação auxilia a identificar as deficiências, oportunidades, prioridades que envolvem a equipe de criação do produto como um todo. Tal análise, contudo, exige uma representação que acomode os diversos níveis da comunicação no processo de design e que permita focalizar elementos importantes que influenciam o sucesso ou insucesso da comunicação.

Neste trabalho, o conceito de comunicação foi trazido de bases semióticas para se entender a mediação de canais e mensagens específicos na integração dos processos de design e desenvolvimento de artefatos de software.

7.1 Contribuições

A partir da identificação da falta de integração entre os processos de design e desenvolvimento que, em última instância, resulta em prejuízo para a usabilidade do produto, e da percepção da interface como uma mensagem enviada pelos designers aos usuários, passamos a abordar o problema pela perspectiva da comunicação.

Foi objetivo deste trabalho entender os diversos níveis de comunicação envolvidos no processo de design de software. A qualidade da comunicação entre a equipe de design e os usuários é fundamental. Contudo, sem focalizar o processo de desenvolvimento de forma mais ampla, não se pode garantir que essa conversação tenha reflexos na interface do produto, melhorando sua usabilidade. É necessário propaga-la de forma adequada para a toda a equipe de criação, principalmente para agentes chave do processo, para que os esforços da equipe de design resultem em modificações efetivas no produto.

O primeiro passo nessa direção foi a proposição do modelo fractal de comunicação (capítulo 4). Esse modelo estende o conceito de comunicação de forma a captar diferentes dimensões do processo comunicativo, representadas em sua estrutura fractal, contribuindo, assim, com um espaço de análise para esse processo. Assim como em alguns modelos que focalizam a comunicação verbal, no modelo fractal, os agentes da comunicação se alternam em turnos como comunicantes e audiência, explicitando que a troca de papéis entre os agentes é determinante para o

sucesso da comunicação. Além disso, o foco em dimensões diferentes da estrutura fractal permite que se analisem os próprios agentes como canais de comunicação.

A representação proposta pelo modelo fractal de comunicação mostrou-se fundamental para se entender a comunicação nos processos de design e desenvolvimento de software. Se a comunicação é bem sucedida, os diversos agentes se alternam como comunicantes e audiência, o que evidencia que todos focalizam em um objetivo comum e que suas participações são igualmente importantes. Em particular, ao focalizar a conversação entre a equipe de design e os usuários, essa percepção é muito mais consistente com a visão das abordagens participativas que enfatizam a cooperação entre esses agentes até mesmo na escolha da terminologia. O termo “aprendizado mútuo”, por exemplo, é adotado em substituição ao convencional “análise” [Kyng, 1991].

O modelo enfatiza também a importância do canal que, em determinados níveis, atua como agente da comunicação. Alguns canais dificultam determinar se a mensagem é “consumida” pela audiência, o que deveria resultar em uma ação (pelo menos, motivar um turno desse agente como comunicante). Além disso, os canais causam um impacto no contexto em que atuam e isso precisa ser observado (ex., a ênfase na utilização de correio eletrônico como um dos canais prioritários na ORG provocou um volume excessivo de informação que trouxe dificuldades de gerenciamento para os indivíduos).

Outra contribuição do modelo ao ser projetado no processo de design é o fato de acomodar os métodos e técnicas correntes de design e usabilidade como canais. A aplicação de tais métodos é entendida como o design de uma mensagem, e segue os mesmos princípios do processo de design da interface como um todo (iterativo, empírico e centrado nos usuários). Isso significa que a aplicação de cada método demanda que ele seja instanciado para um contexto particular. A equipe de design precisa entender sua contribuição em relação ao processo de comunicação como um todo e, freqüentemente, precisa adapta-lo para maximizar seu aproveitamento no contexto. Explicitar esse entendimento também é uma contribuição relevante para a prática de IHC. Isso exige da equipe de design um entendimento profundo sobre a técnica em si e sobre o contexto em que será aplicada. Por outro lado, enfatiza a importância de cada ato de comunicação (cada estudo de usabilidade, no caso) no processo como um todo.

Desta forma, o modelo fractal de comunicação traz a flexibilidade necessária para se representar processos comunicativos complexos mantendo uma simplicidade de representação. No método de inspeção proposto, o percurso comunicativo, o

modelo fractal é usado para meta-modelar a comunicação do processo de design/desenvolvimento de um projeto específico.

O método proposto focaliza duas conversações inerentes ao processo de design: (1) a conversação entre a equipe de design com os usuários, que envolve a aplicação de métodos e técnicas de design e avaliação de usabilidade para coletar informações sobre eles e as tarefas que realizam; (2) a conversação entre a equipe de design e toda a equipe de criação do produto, que diz respeito à utilização dos dados coletados sobre os usuários para suportar as decisões de design.

Em relação à comunicação entre usuários e a equipe de design, o método avalia a suficiência dos canais e mensagens para responder a quatro questões de usabilidade: quem são os usuários, quais são os problemas de usabilidade, como eles se manifestam e porque eles ocorrem. A determinação de respostas a essas questões é uma contribuição no sentido de organizar os problemas de usabilidade no espaço de análise da comunicação. Ao analisar a comunicação quanto a essas questões, é possível:

- identificar como os diferentes métodos e técnicas de design e avaliação de usabilidade são mais ou menos adequados para responder à cada uma das questões;
- determinar se os métodos adotados no processo se complementam de forma adequada;
- identificar outros agentes do processo cujas atividades possam contribuir para responder a algumas dessas questões. Isso se reflete no produto emergente, que incorpora uma visão mais unificada de toda a equipe.

As questões propostas acomodam tanto os métodos e técnicas que focalizam em aspectos mais quantitativos do uso, quanto aqueles voltados para aspectos qualitativos.

Enquanto grande parte dos métodos e técnicas correntes enfatiza a comunicação com os usuários para melhorar a usabilidade dos sistemas, na perspectiva deste trabalho, a propagação dessa conversação é igualmente importante. Designers e engenheiros de usabilidade precisam focalizar no design de mensagens apropriadas para a equipe de criação, ou seja, precisam determinar os agentes chave no processo para os quais o conteúdo de usabilidade é relevante, elaborar mensagens adequadas e propaga-las através de canais adequados.

Por outro lado, assim como os diferentes contextos de desenvolvimento apresentam oportunidades e obstáculos para a utilização de cada método ou técnica de design e avaliação de usabilidade (capítulo 2), na perspectiva deste trabalho, cada projeto específico irá apresentar tais oportunidades e obstáculos. O estudo de casos apresentado confirma essa visão, já que, em uma mesma empresa que propõe um modelo para o ciclo de vida do produto, diferentes projetos encontram soluções particulares para a comunicação entre a equipe de design, usuários e toda a equipe do produto.

O método proposto lida com a complexidade inerente dessa visão do contexto de criação do software. O procedimento de inspeção permite que se analise o processo de uma maneira sistemática, ao mesmo tempo em que considera suas particularidades.

O objetivo do método de inspeção é identificar as deficiências de comunicação e oportunidades para melhorias, para que a conversação com os usuários resulte, em última instância, em modificações efetivas no produto, considerando a visão de toda a equipe.

É importante ressaltar que a aplicação do método proposto não tem impacto apenas no produto, mas no processo que gera tal produto. O controle de qualidade, neste caso, é realizado no processo. Pode-se fazer uma analogia com uma linha de produção em que, sempre que se identifica um defeito em um produto, o processo é interrompido e a causa do erro é investigada. O processo é então modificado e, se a modificação é bem sucedida, todos os produtos gerados a partir de então tendem a não apresentar o defeito observado. O percurso comunicativo tem o mesmo propósito e sua aplicação torna o processo mais estável (i.e., bem consolidado) potencializando a criação de produtos de melhor qualidade.

7.2 Perspectivas futuras

Os processos de design apresentados no capítulo 2 são baseados em uma visão de usabilidade que focaliza o desempenho de indivíduos interagindo com um artefato. Nessa visão de usabilidade, métricas são fundamentais, pois o ciclo design-implementação-avaliação-modificação é repetido iterativamente até que se atinja os objetivos quantitativos estabelecidos.

Entretanto, críticas a essa abordagem têm surgido progressivamente (Ehn, 1997). O foco excessivo em realizar medidas pode acarretar em uma concentração

em aspectos facilmente mensuráveis sem que se observe elementos mais essenciais para a interação.

Particularmente no caso de algumas categorias de software, os parâmetros correntemente considerados (relativos ao desempenho dos usuários) não resumem os benefícios pretendidos com o aumento de usabilidade. Primeiramente, eles focalizam o diálogo entre cada usuário (individualmente) e um sistema. Em um contexto de trabalho, no qual cada funcionário de uma empresa é responsável por uma parcela cujo resultado final irá contribuir para a produção do serviço ou produto que a empresa oferece, existem outras dimensões de usabilidade.

Duas situações distintas podem ser identificadas que poderiam beneficiar-se do modelo de comunicação e método de inspeção propostos neste trabalho:

- vários grupos distintos de usuários cooperam para a obtenção de um produto final utilizando diferentes sistemas monousuário. Em algumas organizações, cada usuário ou grupo de usuários é responsável por determinada etapa de um processo. É comum uma realidade na qual diferentes etapas são executadas com o auxílio de diferentes sistemas monousuário. Neste caso, a saída de um sistema é, potencialmente, entrada de outro. No projeto de rede de telecomunicações, por exemplo, atividades como digitalização do mapeamento urbano e análise do mercado precedem o projeto de criação ou expansão da rede de telecomunicações. Essas atividades são, tipicamente, executadas com o auxílio de sistemas específicos (em função de sua complexidade) e a sua saída é entrada para os sistemas de projeto de redes. Assim, existe uma comunicação indireta entre usuários de diferentes seções através dos sistemas que utilizam.

Nesse caso, benefícios de usabilidade ultrapassam atributos relacionados ao desempenho individual. O tempo poupado por cada usuário é, sem dúvida, importante para a organização. Entretanto, como o custo para a expansão da rede é alto, a qualidade do projeto final da rede é que determina as perdas ou ganhos mais significativos para a empresa. Neste caso, o conceito de usabilidade deve se estender de tal forma que, além de focalizar o desempenho de indivíduos interagindo com sistemas específicos deve também focalizar a produtividade da organização. Pode-se investigar novas métricas de usabilidade para: o conjunto de sistemas que suportam as diversas etapas das atividades de uma organização e o desempenho conjunto dos usuários desses sistemas;

- sistemas multiusuário. Neste caso, duas ou mais pessoas trabalham conjuntamente, ainda que separadas por espaço ou tempo, com o auxílio de um sistema computacional. Aqui também a produtividade do grupo é um atributo tão fundamental quanto o desempenho dos usuários individualmente.

Nessa e em outras situações, é preciso entender usabilidade de uma forma mais ampla. O método de inspeção proposto é flexível para incorporar metodologias que focalizem atributos mais subjetivos. Algumas abordagens como o design participativo (Muller, 1997) e contextual (Beyer e Holtzblatt, 1998) focalizam aspectos mais qualitativos de usabilidade. Assim como o percurso comunicativo foi aplicado ao processo de design de um teste de usabilidade, ele pode ser aplicado a técnicas propostas nesses métodos.

No nível macroscópico, assim como foi aplicado ao ciclo de vida da ORG, o percurso comunicativo pode ser projetado sobre diferentes paradigmas de design de software, para que se possa entender o processo de comunicação que envolvem, o que focalizam na conversação com os usuários, em que medida associam a qualidade do produto à qualidade do processo de criação.

Whiteside e Wixon colocam que:

“For us, software usability refers to the extent to which software supports and enriches the ongoing experience of people who use that software. This direct emphasis on experience is at variance with ‘standard’ definitions of usability that concentrate on, as primary, measurable manifestations of usability such as productivity, preference, learnability or throughput. Although these may be interpreted as important properties of usability, for us they are not primary. Primary is the person’s experience at the moment experienced” (Whiteside e Wixon, 1987, p. 18).

Por outro lado, o design emerge de um ambiente social em que há uma série de fatores e pressões além das considerações dos designers, como aponta Norman:

“When you are asked to solve a problem, look beyond it. Ask why that particular problem arose in the first place. Search beyond the technical: Question the business model, the organizational structure, and the culture. The path to a solution seldom lies in the question as posed: the path appears only when we are able to pose the right question.” (Norman, 1996, p. 233).

Caminhar rumo ao entendimento das questões, sobre como o software estende as possibilidades humanas, foi nosso objetivo durante o desenvolvimento dessa pesquisa e esse desejo orienta nossos passos futuros. Para tanto, temos procurado

entender o design e a complexidade inerente a ele: as conseqüências individuais e sociais provocadas por um artefato e a forma como o design reflete o próprio contexto do qual emerge.

Bibliografia

- (Adler e Winograd, 1992) Adler, Paul S. e Winograd, Terry A. (ed.). *Usability: turning technologies into tools*. Oxford University Press, 1992.
- (Andersen, 1997) Andersen, Peter BØgh. *A theory of computer semiotics*. Cambridge University Press, 1997.
- (Baecker et al., 1995) Baecker, Ronald M., Grudin, Jonathan, Buxton, William A. S., Greemberg, Saul (ed.). *Human-Computer Interaction: Toward the Year 2000*. Morgan Kaufmann Publishers, 1995.
- (Baranauskas et al, 1998) Baranauskas, M. Cecília C., Rosslet, Flavia e de Oliveira, Osvaldo Luiz. *Uma Abordagem Semiótica à Análise de Interfaces: um estudo de caso*. Anais do IHC 98, Maringá. 1998.
- (Beyer e Holtzblatt, 1998) Beyer, Hugh e Holtzblatt, Karen. *Contextual design: defining customer-centered systems*. Morgan Kaufmann Publishers, 1998.
- (Bias e Mahew, 1994) Bias, R. G. e Mahew, D. J. (ed.) *Cost-justifying usability*, Academic Press, 1994.
- (Boehm, 1988) Boehm, W. *A Spiral Model of Software Development Enhancement*. IEEE Computer 21(2), pp. 61-72, Maio 1988.
- (Brown, 1995) Brown, Gillian. *Speakers, listeners and communication: exploration in discourse analysis*. Cambridge University Press, 1995
- (Bush, 1996) Bush, Vannervan. *As We May Think*. *Interactions*, 1996, v. 3, n. 2. Reprinted from The Atlantic Month, julho, 1945.
- (Coelho Neto, 1996) Coelho Neto, José Teixeira. *Semiótica, informação e comunicação*. 4ª ed., São Paulo, Editora Perspectiva.
- (Cunningham, 1998) Cunningham, D. J. *Cognition as semiosis: the role of inference*. <http://php.ucs.undiana.edu/~cunningh/theory.html> , 1998.

- (de Oliveira e Baranauskas, 1998a) de Oliveira, Osvaldo Luiz e Baranauskas, M. Cecília Canalani. *A semiótica e o design de software*. RT IC-98-09. Unicamp, 1998.
- (de Oliveira e Baranauskas, 1998b) de Oliveira, Osvaldo Luiz e Baranauskas, M. Cecília Canalani. *Semiotic proposals for software design: problems and prospects*. RT IC-98-10. Unicamp, 1998.
- (de Souza, 1993) de Souza, Clarisse S. *The semiotic engineering of user interface languages*. International Journal of Man-Machine Studies, v. 39, p. 753-773. 1993.
- (de Souza, 1999a) de Souza, C. S.. *Leading users from interaction into programming: The teaching-centered approach of semiotic engineering*. INTERACT99, 1999.
- (Deely, 1990) Deely, John. *Basics of semiotics*. Indiana University Press, 1990.
- (Denning e Dargan, 1996) Denning, Peter e Dargan, Pamela. *Action-Centered Design*. In Winograd, Terry (ed.). *Bringing Design to Software*. ACM Press. 1996.
- (Draper e Norman, 1985) Draper, Stephen W. e Norman, Donald A. *Software Engineering for User Interfaces*. IEEE Transactions on Software Engineering, vol. SE-11, no. 3, Março de 1985.
- (Eco, 1990) Eco, Umberto. *O Signo*, 4^a ed., Editorial Presença, Lisboa, 1990.
- (Eco, 1976) Eco, Umberto. *Tratado Geral de Semiótica*. Trad. Antônio de Pádua Danesi e Gilson César Cardoso de Souza (A Theory of Semiotics. Bloomington: Indiana University Press). São Paulo, Editora Perspectiva, 1980.
- (Ehn, 1997) Ehn, Pelle. *Design for Quality-in-use: Human-Computer Interaction Meets Information Systems Development*. In M. Helander, T.K. Landauer e P. Prabhu (eds.). *Handbook of Human-Computer Interaction*. 2^a ed., Elsevier, 1997.
- (Familiant e Detweiler, 1993) Familiant, M. E. e Detweiler, M. C. *Iconic reference: evolving perspectives and an organizing framework*. International Journal of Man-Machine Studies 39, 705-728, 1993.

- (Gaines e Shaw, 1986) Gaines, Brian. R. e Shaw, Mildred L. G.. *From timesharing to sixth generation: the development of human-computer interaction*. Int. J. Man-Machine Studies. v. 24, p. 1-27. 1986.
- (Gould, 1995) Gould, J. *How to Design Usable Systems*. In Ronald M. Baecker, Jonathan Grudin, William A. S. Buxton, Saul Greemberg (ed.). *Human-Computer Interaction: Toward the Year 2000*. Morgan Kaufmann Publishers. 1995, p. 93-121.
- (Gould et al., 1991) Gould, J., Boies, S. e Lewis, C. *Making Usable, Useful, Productively-Enhancing Computer Applications*. Communications of the ACM 34(1), pp. 74-85, 1991.
- (Gould e Lewis, 1985) Gould, J. e Lewis, C. *Designing for Usability: Key Principles and What Designers Think*. Communications of the ACM 28(3), pp. 300-311.
- (Gray e Salzman, 1998) Gray, Wayne D. e Salzman, Marilyn C.. *Damage Merchandise? A Review of Experiments that Compare Usability Evaluation Methods*. Human-Computer Interaction, 1998, vol. 13, pp. 203-261.
- (Grudin, 1989) Grudin, Jonathan. *The case against user interface consistency*. Communications of the ACM, outubro, 1989.
- (Grudin, 1991) Grudin, Jonathan. *Interactive systems: bridging the gaps between developers and users*. IEEE Computer, abril, 1991.
- (Grudin, 1993) Grudin, Jonathan. *Interface: an evolving concept*. Communications of the ACM, abril, 1993.
- (Hartson, 1998) Hartson, H. Rex. *Human-computer interaction: interdisciplinary roots and trends*. Journal of Systems and Software. 43 (1998) 103-118.
- (Hartson e Hix, 1989) Hartson, H. Rex e Hix, Deborah. *Toward empirically derived methodologies and tools for human-computer interface development*. Int. J. Man-Machine Studies, vol. 31, pp. 477-494, 1989.
- (Hix e Hartson, 1993) Hix, Deborah e Hartson, H. Rex. *Developing user interfaces: ensuring usability through product and process*. John Wiley & Sons, 1993.

- (Jakobson, 1960) Jakobson, Roman. *Closing Statement: Linguistics and Poetics*, in: Sebeok, A, ed., *Style in Language* (Cambridge, Mass.: The MIT Press, 1960).
- (Jakobson, 1991) Jakobson, Roman. *Lingüística e comunicação*. 16^a edição. Cultrix, São Paulo, 1991.
- (Jorna e Van Heusden, 1996) Jorna, René e Van Heusden, Barend. *Semiotics of the user interface*. *Semiotica* 109-3/4, 237-250, 1996.
- (Kapor, 1996) Kapor, Mitchell. *A software design manifesto*. In Winograd, Terry (ed.), *Bringing design to software*. Addison-Wesley, 1996.
- (Karat, 1994) Karat, Clare-Marie. *A business case approach to usability cost justification*. In Bias, R. G. e Mayhew, D. J., (ed.), *Cost-justifying usability*, Academic Press, 1994.
- (Kyng, 1991) Kyng, Morten. *Designing for cooperation: cooperating in design*. *CACM*, 34(12), pp. 65-73, 1991.
- (Laurel, 1993) Laurel, Brenda (editor). *Computers as Theatre*. Addison-Wesley Publishing Company. 1993.
- (Laurel, 1990) Laurel, Brenda (editor). *The art of human-computer interface design*. Addison-Wesley Publishing Company. 1990.
- (Lavery et. al., 1997) Lavery, Darryn, Cockton, Gilbert e Atkinson, Malcolm P. *Comparison of evaluation methods using usability problem reports*. *Behaviour & Information Technology*, 1997, vol. 16, no. 4/5, pp. 246-266.
- (Leite, 1998) Leite, Jair C. *Modelos e Formalismos para a Engenharia Semiótica de Interfaces de Usuário*. Tese de doutorado. PUC Rio. 1998.
- (Liu, 2000) Liu, Kecheng. *Semiotics in information systems engineering*. Cambridge University Press, 2000.
- (MacIntyre et al., 1990) MacIntyre, F., Estep, K. W., e Sieburth, J. M.. *Cost of user-friendly programming*. *Journal of Forth Application and Research* 6 (2), 103-115, 1990 .

- (Mandelbrot, 1983) Mandelbrot, Benoit B. *The fractal geometry of nature*. W. H. Freeman, Nova York, 1983.
- (Martin e McClure, 1983) Martin, J. e McClure, C.. *Software Maintenance: The Problem and its Solution*. Prentice Hall, New Jersey, 1983.
- (Mayhew, 1992) Mayhew, Deborah J. *Principles and Guidelines in software user interface design*. Prentice Hall, New Jersey. 1992.
- (Mcluhan, 1964) Mcluhan, Marshall. *Os meios de comunicação como extensões do homem*. Cultrix, São Paulo, 1964.
- (Muller, 1997) Muller, M. *Participatory Practices in the Software Lifecycle*. In: Helander *et al.* (eds.) *Handbook of Human-Computer Interaction*. Elsevier Science, 1997.
- (Nadin, 1988a) Nadin, Mihai. *Interface Design and Evaluation – Semiotic Implications*. In Hartson, H. Rex and Hix, Deborah (ed.). *Advances in Human-Computer Interaction*. vol. 2. Ablex Publishing Corporation. 1988.
- (Nadin, 1988b) Nadin, Mihai. *Interface design: A semiotic paradigm*. *Semiotica* 69-3/4, 269-302, 1988.
- (Nielsen, 1993) Nielsen, Jakob. *Usability Engineering*. AP Professional. 1993.
- (Nielsen e Mack, 1994) Nielsen, Jakob and Mack, Robert L. *Usability Inspection Methods*. John Wiley & Sons. 1994.
- (Norman, 1986) Norman, Donald A. *Cognitive Engineering*. In Norman, Donald A. e Draper, Stephen W. (editors). *User centered system design: new perspectives on human-computer interaction*. Lawrence Erlbaum Associates. 1986.
- (Norman, 1988) Norman, Donald A.. *The Psychology of Everyday Things*. Basic Books, New York, 1988.
- (Norman, 1993) Norman, Donald A.. *Things that make us smart: defending human attributes in the age of machine*. Addison-Wesley. 1993.

- (Norman, 1996) Norman, Donald A.. *Design as Practiced*. In Winograd, Terry (ed.) Bringing design to software. ACM Press, 1996.
- (Norman, 1998) Norman, Donald A. *The invisible computer: why good products can fail, the personal computer is so complex, and the information appliances are the solution*. MIT Press, 1998.
- (Norman e Draper, 1986) Norman, Donald A. e Draper, Stephen W. (editors). *User centered system design: new perspectives on human-computer interaction*. Lawrence Erlbaum Associates. 1986.
- (Peirce, 1931-1935) Peirce, Charles, S. *Semiótica*. Trad. José Teixeira Coelho Neto (Collected papers of Charles Peirce. 3rd printing. Harvard University Press, Cambridge, 1974). Editora Perspectiva.
- (Polson et al., 1992) Polson, P. G., Lewis, C., Rieman, J. e Wharton, C. *Cognitive walkthroughs: a method for theory-based evaluation of user interfaces*. International Journal of Man-Machine Studies 36, 741-773, 1993.
- (Prates et al., 1998) Prates, R. O., Leite, J. C. e de Souza, C. S. *Semiotically based user interfaces design languages*. In Anais do IHC98 Workshop. Maringá, Brasil. 1998.
- (Prates, 1998) Prates, Raquel O. *Engenharia Semiótica de Linguagens e Interface Multi-usuário*. Ph.D. Thesis. Puc Rio. 1998.
- (Prates et al., 2000) Prates, R. O., de Souza, C. S. e Barbosa, S. *A Method for Evaluating Communicability of User Interfaces*. Interactions v. 7 (1), Janeiro de 2000.
- (Press, 1995) Press, Larry. *McLuhan Meets the Net*. CACM, vol. 38, No. 7, Julho, 1995.
- (Pressman, 1995) Pressman, Roger. *Engenharia de Software*. 3^a ed. Makron Books, S. Paulo, 1995.
- (Rheinfrank et al., 1992) Rheinfrank, John J., Hartman, William R. e Wasserman, Arnold. *Designing for Usability: Crafting a Strategy for the Design of a New*

- Generation of Xerox Copiers*. In Adler, Paul S. and Winograd, Terry A. (ed.). Usability: turning technologies into tools. Oxford University Press, 1992.
- (Rosenberg, 1989) Rosenberg, D. *A cost benefit analysis for corporate user interface standards: what price to pay for consistent look and feel?* In Coordinating User Interfaces for Consistency (J. Nielsen, ed.), Academic Press, 1989.
- (Salles *et al.*, 2000a) Salles, Juliana, Baranauskas, M. Cecília C. e Bigonha, Roberto da Silva. *A communication model for interface design process*. Workshop in Semiotics, CHI2000, <http://peirce.inf.puc-rio.br/chi2000ws/ppapers/fr-salles.htm>, visitado em Fevereiro de 2001.
- (Salles *et al.*, 2000b) Salles, Juliana, Baranauskas, M. Cecília C. e Bigonha, Roberto da Silva. *Towards a communication model applied to interface design process*. Forthcoming In KBS, Elsevier.
- (Santaella, 1990) Santaella, Lúcia. *O que é semiótica*. 8^a ed., Editora Brasiliense, 1990.
- (Sebeok, 1994) Sebeok, Thomas A. *Signs: an introduction to semiotics*. University of Toronto Press, 1994.
- (Searle, 1969) Searle, John R., *Speech Acts*, Cambridge: Cambridge University Press, 1969.
- (Shackel, 1969) Shackel, B. *Man-Computer Interaction – The Contribution of the Human Sciences*. Ergonomics, 1969, vol. 12, no. 4, 485-499.
- (Shannon e Weaver, 1949) Shannon, C. L. e Weaver, W. *The mathematical Theory of Communication*, Urbana, Univ. of Illinois Press, 1949.
- (Shneiderman, 1992) Shneiderman, Ben. *Designing the user interface: strategies for effective human-computer interaction*. Addison-Wesley, 2^a edição, 1992.
- (Shneiderman, 1998) Shneiderman, Ben. *Designing the user interface: strategies for effective human-computer interaction*. Addison-Wesley, 3^a edição, 1992.

(Westly e MacLehan, 1957) Westley, B. H. e MacLehan, M. S.. *Un modelo conceptual para la investigación en comunicaciones*. In A. G. Smith (org.), *Comunicación y cultura*. Nueva Visión, Buenos Aires, 1972.

(Whiteside e Wixon, 1987) Whiteside, J. e Wixon, D. *The dialectic of usability engineering*. In Bullinger, H. J., and Shackel, B. (eds.) *Human-computer interaction – Interact 87'*, pp. 17-20. Amsterdam: Elsevier.

(Winograd, 1996) Winograd, Terry (ed.). *Bringing Design to Software*. ACM Press, 1996.

(Yourdon, 1989) Yourdon, Edward. *Structured walk-throughs*. Yourdon Press, 4^a edição, 1989.