

1 Introdução

Sistemas de ajuda *online* têm recebido pouca atenção da comunidade de IHC (Interação Humano-Computador) nos últimos anos.

Esta atitude da comunidade, por um lado, pode ser devida à não utilização destes sistemas por usuários em geral: se os usuários não utilizam a ajuda, a mesma não justifica que se lhe dispense uma maior atenção. Um dos motivos desta “não utilização” é que a maioria dos sistemas de ajuda *online* são relacionados a operações, a instruções que o usuário deve seguir para realizar determinadas ações (copiar, imprimir, etc), sendo que poucos deles são relacionados às tarefas e ao *rationale* (ou lógica de decisões e objetivos) por trás da aplicação. E, sem este tipo de informação, estes sistemas dificilmente conseguem ajudar o usuário a entender o que a aplicação significa e o que ele necessita realizar na mesma (Swartout, 1983).

Por outro lado, muitos membros da comunidade de IHC não esperam que os usuários utilizem a ajuda¹, por ser este uso um sinal – em termos de usabilidade – de um design mal desenvolvido. A idéia de que boas interfaces são intuitivas, auto-explicativas, fáceis de usar, e assim por diante, tem transformado os sistemas de ajuda em algo indesejável, em uma indicação de que, no mínimo, parte desta “intuição” não está funcionando.

Apesar destes posicionamentos, os sistemas de ajuda são, certamente, essenciais a uma aplicação bem projetada. Estes sistemas são um canal privilegiado de comunicação, através do qual os designers² podem falar sobre seus objetivos e decisões aos usuários das aplicações que desenvolveram. Isto pode

¹ Neste trabalho será usado o termo “ajuda” ou “sistema de ajuda”, genericamente, como uma denominação aos recursos de ajuda *online*, independentemente de questões de arquitetura que possam diferenciar um “sistema”, de um “módulo”, de uma “função” de ajuda *online*.

² Neste trabalho o termo “designer” (ou “designers”) refere-se a equipe de design como um todo, podendo ser uma ou mais pessoas.

levá-los a aumentar sua produtividade e a ter uma maior satisfação na interação com esta aplicação. O conteúdo da ajuda deve possibilitar ao usuário corrigir interpretações equivocadas sobre o funcionamento da aplicação, alcançar a lógica de design do designer e ter *insights* sobre novas situações que podem surgir no uso da aplicação. Entretanto, conseguir alcançar este objetivo tem sido uma tarefa difícil: decidir o que falar aos usuários e como fazer isto, assim como escolher uma forma efetiva e eficiente de deixar que os usuários expressem suas necessidades de ajuda, é um grande desafio lançado às equipes de desenvolvimento de software.

O objetivo deste trabalho é fornecer aos designers de software ferramentas epistêmicas que possam apoiá-los na construção dos sistemas de ajuda de suas aplicações. Uma ferramenta epistêmica é usada para aumentar o entendimento de uma pessoa sobre o problema que ela está resolvendo; não para lhe dar diretamente uma resposta ou focar diretamente a sua solução, como é o caso de diretrizes e regras, por exemplo (de Souza, 2002). Estas ferramentas, então, permitirão aos designers explorar o poder comunicativo da ajuda, auxiliando-os a refletir sobre os materiais disponíveis para sua elaboração (modelos de design, lógica de design, entre outros) e, a partir desta reflexão, auxiliando-os tanto na construção do discurso embutido na ajuda quanto das possíveis formas de expressão do usuário frente a este discurso.

Para isto tomou-se por base a teoria da Engenharia Semiótica (de Souza, 1993). A essência desta teoria é que artefatos de software são artefatos de metacomunicação, ou seja, eles comunicam (sobre) comunicação. Estes artefatos são mensagens unidirecionais, enviadas do designer para o usuário, representando a visão do designer sobre o artefato por ele construído e as mensagens que os usuários podem trocar com o sistema a fim de poder utilizá-lo. Isto caracteriza IHC explicitamente como um processo comunicativo envolvendo a comunicação designer-usuário e a interação usuário-sistema. Na visão da Engenharia Semiótica, se o usuário conseguir captar a mensagem do designer sobre os diferentes tipos de mensagens que o sistema está preparado para interpretar, então a interface poderá ser inteligível, usável (embora potencialmente não *ótima*) e consistente para o usuário.

Outra característica desta teoria é que, como o designer não pode estar presente fisicamente na interface, ele é representado através do que é chamado o preposto do designer (Prates et al., 2000). Este preposto deve ser (idealmente) dotado de uma capacidade comunicativa que lhe permite fazer um discurso completo e exclusivamente referente à sua conclusão final sobre as necessidades, preferências, capacidades e oportunidades que ele entende que o usuário tem. Como é impossível antecipar e tratar todo o espectro de interpretações possíveis que cada usuário pode dar para o software, este discurso do designer via preposto desempenha um papel crucial para fazer estas interpretações convergirem em torno de um conjunto consistente com os seus objetivos e decisões. Através de explicações e ilustrações sobre o que o designer fez (e por que motivo), bem como sobre o que ele esperava que seus usuários soubessem, quisessem ou fizessem, ele amplia os recursos de seus reais usuários (em reais situações de uso) para interpretarem e usarem o seu produto. Dada a riqueza e a natureza dos conceitos e informações a serem apresentadas, o sistema de ajuda parece ser, sem dúvida, um local privilegiado para esta meta-comunicação.

Esta estrutura comunicativa de sistemas interativos (a comunicação designer-usuário e a interação usuário-sistema) e a metáfora de um preposto do designer dão os critérios para selecionar os conteúdos que devem estar presentes em sistemas de ajuda. Mas acessar estes conteúdos requer que os usuários sejam capazes de “falar” com este preposto.

Esta “fala” do usuário surgiu na Engenharia Semiótica associada ao método de avaliação de comunicabilidade (Prates et al., 2000), embora em contexto (e com objetivos) distintos dos aqui tratados. Este método é baseado em expressões que caracterizam a atitude do usuário durante a interação com o software. Por exemplo, quando usuários fazem algo que traz um resultado inesperado, a expressão utilizada é “Epa!”. Quando o usuário não consegue atribuir significado a um elemento na tela, a expressão é “O que é isto?”. Quando o usuário abandona um certo curso de ação por outro, na tentativa de realizar uma tarefa, a expressão é “Vai de outro jeito.”³.

³ O conjunto completo destas expressões (em inglês) pode ser visto em (Prates et al., 2000).

No caso da avaliação de comunicabilidade, esta comunicação usuário→preposto é simulada pelos avaliadores, com base nas interações. Ou seja, são “colocadas palavras” (as expressões de comunicabilidade) “na boca dos usuários”, imaginado-se o que eles diriam em determinada situação. Mas estas mesmas expressões, além de serem usadas para avaliar a efetividade e a eficiência da comunicação designer-usuário, podem ser, também, relacionadas a rupturas de comunicação algumas das quais podem ser previstas (e tratadas) pelos sistemas de ajuda.

Previstas estas rupturas de comunicação e contempladas no sistema de ajuda formas de auxílio ao usuário quando de sua ocorrência, estas expressões servem também como meio de comunicação entre o usuário e o preposto do designer, a fim de que seja possível sinalizar tais rupturas durante a interação. Agora o usuário pode escolher (“falar”) uma dentre um conjunto pré-definido de expressões, baseadas não só na pesquisa em avaliação de comunicabilidade, mas também nas dúvidas mais frequentes que usuários têm durante a interação com uma aplicação (Sellen e Nicol, 1990; Baecker et al., 1995). Estas expressões fornecem um ponto de entrada a uma estrutura de discurso coesa, que contém a lógica de design da aplicação e instruções operacionais, táticas e estratégicas sobre como usar esta aplicação, na tentativa de responder às dúvidas dos usuários.

Neste sentido, o trabalho aqui apresentado estende o modelo teórico atual da Engenharia Semiótica (da Silva, 2001) a fim de explicitar a presença do sistema de ajuda e o papel comunicativo do mesmo. Nesta extensão, o sistema de ajuda é conceitualizado como uma parte integrante da aplicação, ela mesma (como já visto) sendo conceitualizada como uma mensagem do preposto do designer para os usuários. Acomoda também uma mensagem do usuário para o preposto do designer, cujo objetivo é requisição de ajuda ou de aprofundamento da ajuda recebida.

O objetivo central desta tese é, portanto, o de avançar a teorização de Engenharia Semiótica no que tange à expressão da meta-mensagem unidirecional dos designers de aplicações interativas para os usuários a que elas se destinam. A forma como este objetivo foi alcançado valeu-se de três bases fundamentais: a análise crítica dos trabalhos relacionados a sistemas de ajuda *online* e a

Engenharia Semiótica; a elaboração de soluções teóricas na forma de conceitos, modelos e arquiteturas (revisados ou criados); e finalmente uma apreciação, por parte de reais ou potenciais usuários destas ferramentas, da concretização de tais soluções sob a forma de ferramentas epistêmicas, dada a qualidade percebida do aporte de tais ferramentas para os processos e produtos de design.

Para refletir os passos no caminho rumo a este objetivo, o texto deste trabalho está estruturado da seguinte forma: o capítulo 2 apresenta a evolução dos sistemas de ajuda, desde sua criação até os dias de hoje, a fim de verificar os problemas e soluções surgidos e discutidos em cada período e de situar a abordagem de ajuda aqui apresentada. Também faz parte deste capítulo uma caracterização dos diferentes padrões interativos hoje disponíveis para sistemas de ajuda *online*. O capítulo 3 discute o papel do sistema de ajuda na teoria da Engenharia Semiótica, apresentando a extensão do modelo teórico acima citado, bem como uma proposta de como capturar a mensagem do designer para, posteriormente, transformá-la no discurso de seu preposto, e de como permitir ao usuário comunicar suas dúvidas quando da ocorrência de alguma ruptura de comunicação durante a interação. O capítulo 4 detalha a concretização do discurso do preposto do designer através da captura dos elementos necessários à construção deste discurso, a concretização do mesmo via uma arquitetura de sistemas de ajuda e o detalhamento desta concretização através de um estudo de caso. Já o capítulo 5 apresenta uma discussão sobre questões levantadas no decorrer deste trabalho sob o ponto de vista de diferentes agentes que atuam no processo de desenvolvimento de software, como desenvolvedores de software, desenvolvedores de sistemas de ajuda e usuários destes sistemas. E, por fim, o capítulo 6 apresenta as contribuições finais deste trabalho e possibilidades de continuidade da pesquisa realizada.