

7. Discussão dos Resultados da Pesquisa

Todos os signos de uma interface gráfica são indiciais na medida em que têm uma relação de contigüidade causal com seus objetos. Em outras palavras, o resultado da ação de operar um ícone, um botão ou um comando de um menu é quase instantâneo, signo e objeto estão distantes um do outro por apenas poucos segundos de espera.

Entretanto, não é eficiente, nem seguro, clicar em tudo o que se vê até obter o resultado esperado. Neste caso, tentativa e erro não é uma boa política, portanto é necessário depreender o significado de um signo antes de usá-lo. Em um complexo sistema de signos vagos e contextuais, essa é uma difícil tarefa.

Tanto mais difícil quanto for a inexperiência do usuário, pois no caso de uma interface gráfica, sua interação será guiada por juízos visuais, e como Nojima (1999, p. 17) salienta, “os signos visuais não alfabéticos são de baixa definição, e a informação que deles decorre pode ser rica, porém pouco saturada em relação à precisão de seus dados”.

Como foi mostrado no Capítulo 5, a natureza simbólica dos signos da interface do *Windows XP*[®] faz com que não seja simples compreender seu significado *a priori*. Sem mais informações sobre um botão, por exemplo, o usuário é levado a deduzir leis gerais a partir de leis mais particulares, no sentido de “se funcionou para aquele, deve funcionar para este”. Em relação a legissignos simbólicos, cada verificação da lei em um fenômeno é singular, o que vale para um não vale para outro, ou para todos.

Usuários experientes, que já conhecem os princípios fundamentais que regem o uso da interface, podem operar inferências com maior possibilidade de sucesso, pois, como quer Eco (1998), “somente se eu já conheço a regra geral segundo a

qual ‘se fumaça, então fogo’ eu sou capaz de tornar significativas as informações sensoriais”. Os usuários experientes constroem, em termos de teoria da comunicação, um referencial sobre o potencial de sucesso que tem uma inferência, com base naquilo que *abduziram* da interface ao longo do tempo. Mais abduções significa maior possibilidade de se pressupor corretamente se uma operação inferencial na interface terá ou não sucesso.

De fato, após uma atenta análise dos seus signos e semiose, a interface gráfica do *Windows XP*[®] demonstrou ser, do ponto de vista lingüístico, muito semelhante às interfaces de linha de comando. Ambas têm semântica e sintaxe rigorosas, isto é, seus significados são estritos, assim como as relações formais entre seus signos.

As duas interfaces têm uma interpretação limitada — por limitações computacionais — da semiose (ilimitada) dos usuários, mas somente para os usuários da interface de linha de comando isso fica evidente¹.

A grande diferença entre elas, a princípio, é que ao contrário do que acontece com as interfaces de linha de comando, aos usuários das interfaces gráficas não é fornecido um dicionário de “vocábulos”, e deles é esperado que interajam com a interface sem qualquer conhecimento prévio dos significados e do emprego dos signos.

Isto complica o quadro, porque se aos usuários de interfaces de linha de comando é esclarecida de início a restrita capacidade de interpretação do sistema, e a conseqüente necessidade de restringirem sua expressão a palavras-chave “válidas”, aos usuários de interfaces gráficas fica a impressão de que podem fazer o que quiserem. Não podem, sua comunicação com o sistema tem que se limitar aos mesmos termos-chave “válidos”, muito mais complexos do que palavras porque são operações visuais, espaciais e gestuais.

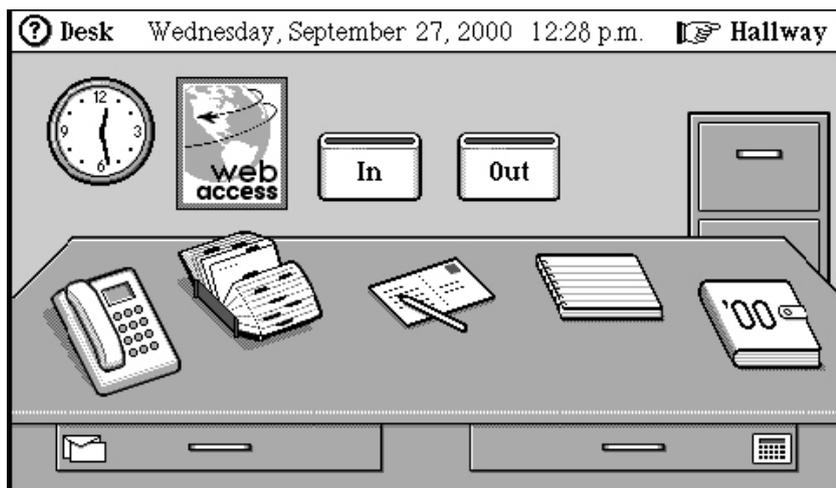
¹ Ver a noção de *habitabilidade* em Watt (1968), segundo a qual o sistema leva o usuário a pensar que porque ele (sistema) pode interpretar e responder bem em certas situações de conversa, pode fazer o mesmo em qualquer outra.

O aprendizado do uso de interfaces gráficas pode ser mais rápido, mas a retenção de informações é menor. Há grande reincidência de erros dos usuários, e como frequentemente eles não compreendem porque erraram, se é que percebem seu erro, é necessário um número maior de mensagens de erro, e que sejam mais *esclarecedoras*. Em outras palavras, não basta avisar que houve um erro ou qualquer outro problema com a interface, é necessário educar o usuário.

Supõe-se que, porque interfaces gráficas são intuitivas, seus usuários não necessitam de maiores instruções. Acima de tudo, interfaces gráficas não são intuitivas. Elas são, em última análise, a linguagem que o usuário utiliza para se comunicar com o sistema, e, como em toda linguagem, é absolutamente necessário que os usuários sejam “alfabetizados”.

Supõe-se também que utilizando elementos gráficos que sejam familiares ao usuário, ele não precisará de uma instrução básica. Mas, ainda que esses elementos lhe sejam familiares, sua relação com o objeto “computacional”, isto é, com aquilo que representam na interface não é evidente nem funciona da mesma maneira como no mundo real.

Desta forma, não cabe levar as metáforas às últimas conseqüências, como alguns propõem. Além de aumentar a opacidade do sistema, quanto mais realistas as metáforas mais será exigido delas que se comportem (e signifiquem) como seus paralelos no mundo real. Quanto mais se avança no mundo da metáfora visual, mais incompreensível vai ficando a relação do signo com o objeto. Por um lado, essa relação será mais restringida pelas limitações computacionais, e de outro lado, pela distância do objeto (real) a que as metáforas se referem e o objeto a que os signos se referem na interface.



Schneiderman (1993) sugere que interfaces devem ser cognitivamente compreensíveis, consistentes, previsíveis e controláveis. Interfaces gráficas não serão nada disso até que se explicita aos usuários a limitação interpretativa do sistema, e, desta forma, a impossibilidade de efetuarem *qualquer* tipo de operação com a interface. Por conseguinte, assim como existe em interfaces de linha de comando, é importante que tanto designers quanto usuários entendam a necessidade de uma “gramatização” dos signos de uma interface gráfica. Para isto, o “discurso do preposto do designer”, proposto por Souza (2005b), tem função chave. É ele que deve dizer ao usuário como e por que utilizar a interface. Sem algo mais que cumpra este papel, os usuários só terão dificuldades.