

## Como os *softwares* são fabricados: um olhar ergolinguístico

Aguinaldo Gomes de Souza<sup>1</sup>

### Resumo:

O presente trabalho tem por objetivo descrever, a luz da ergolinguística, o modo de fabricação do *software*. Parte-se do pressuposto de que, ao desenvolver um *software*, o sujeito desenvolvedor deixa nele formas marcadas e não marcadas de sua presença. Nesse sentido, o trabalho em uma fábrica de *software* é sempre de interação e co-construção de ações. Assim, na descrição do processo de fabricação de um artefato tecnológico percebe-se as relações dialógicas que o atravessam, desde o processo de fabricação até o processo de uso.

**Palavras-chave:** *Software*, interface, dialogismo.

### Abstract:

This article aims to describe, according to an ergolinguistics perspective, the way software are produced. We assume that when an individual develops a software he/she leaves marked (and unmarked) forms of his/her presence. Thus, working in a software development enterprise is always interaction and co-building action. Hence, when we describe a technological artifact building process, we realize dialogic relations going through it, from the development process to the using process.

**Keywords:** Software, interface, dialogic.

## Introdução<sup>2</sup>

Nas ciências exatas, são inúmeros os trabalhos (MAIBAUM, 2000; BROOKSHEAR, 2001; REZENDE, 2005; BRAUDE, 2005; FLOYD, 2007 entre muitos) que têm como objeto reflexivo o desenvolvimento e o uso de *softwares*, esses descritos como aplicações complexas baseadas em modelos matemáticos. São numerosas também as investigações sobre a interface do *software* e as interações com os humanos (NORMAN, 1991;

MURPHY, 1998; BARBOSA et al., 2003; PONTES et al., 2004; PONTES et al., 2005 entre muitos ). Tais estudos, sobre a interface, deixam em evidência os processos pelos quais os humanos interagem, com artefatos informatizados, com os *softwares*.

Entre tais investigações podemos encontrar trabalhos cujo escopo é a semiótica os quais se detêm mais especificamente na programação para usuários, na metacomunicação em sistemas interativos, nos modelos conceituais de *design* bem como no método de avaliação de IHC (interação humano-computador). Essa abordagem semiótica considera a interface como um preposto do *designer* que deverá, entre outras coisas, comunicar ao usuário, a visão do *designer* sobre o sistema. Quanto ao próprio sistema computacional, ao *software*, estes transmitem a ‘fala’ dos *designers* aos usuários através de um processo chamado de metacomunicação, ou ainda processo de comunicação sobre a comunicação. Tais estudos colocam em discussão a necessidade de um fator intersubjetivo no desenvolvimento de *softwares*.

Se, por um lado, já existe uma produção de conhecimento razoável sobre *software* e interface no âmbito das ciências exatas, nas ciências da linguagem, e mais especificamente na ciência linguística, tal produção de conhecimento ainda é carente de desenvolvimento e sistematização. Dessa forma, esse trabalho tem como foco central descrever o modo de nascituro de um artefato informatizado, o qual serve de aporte para os textos, hipertextos/enunciados e gêneros. Nas ciências da linguagem, são praticamente inexistentes trabalhos que tenham como ponto de reflexão tal artefato, o *software*.

Não obstante, mesmo diante da carência de literatura sobre o tema proposto, pode-se observar, na prática, que cada vez mais sujeitos interagem com e através de *softwares*. Trata-se então de interações que acontecem, substancialmente, através da língua escrita, às vezes para produzir e obter conhecimento acadêmico, como é o caso, por exemplo, da educação a distância; às vezes para interagir com outros sujeitos situados em comunidades, como as que encontramos na rede social do Orkut ou mesmo por programas mensageiros como o *Messenger*.

Assim, o trabalho em uma fábrica de *software* que aqui será explicitado<sup>3</sup> parte da compreensão de que as relações entre linguagem e trabalho não se resumem apenas as ações languageiras em ambiente de trabalho, mas antes é pensada como a tripartição de Lacoste (1998), ou seja, é pensada como linguagem no/como/sobre trabalho<sup>4</sup>. Esses “três aspectos da linguagem estão em estreita ligação com práticas languageiras na situação de trabalho considerada de maneira global” (NOUROUNDINE, 2002, p. 18).

## 2 Os estudos sobre o trabalho

Para compreendermos o modo de nascituro de um *software*, é necessário atentar para as especificidades (as relações de trabalho que se estabelecem entre os protagonistas da atividade durante o processo de fabricação

de *software*) que norteiam o seu desenvolvimento. Falamos da relação linguagem/trabalho vista a luz da ergolinguística<sup>5</sup> e introduzida aqui a fim de explicar o processo de nascituro de um *software*.

Os estudos sobre o trabalho, inicialmente, desenvolvidos pela Ergonomia, levavam em consideração a dicotomia ‘trabalho prescrito e trabalho real’ (FAÏTA, 2002). Com o avanço em tais estudos a Ergonomia passa a dialogar com outras disciplinas, como, por exemplo, a Ergologia. Desse modo, houve uma revisão acerca dos conceitos de trabalho prescrito e trabalho real. A Ergologia<sup>6</sup> trouxe o homem e toda sua complexidade para os estudos sobre o trabalho, pois até então o trabalhador era visto como um ser despersonalizado cujo papel era desempenhar maquinalmente sua tarefa.

No âmbito dos estudos sobre o trabalho postulou-se o conceito de ‘trabalho prescrito’, tal postulado toma força e incorpora-se às ciências que estudam o trabalho vinculando-se diretamente com as normas institucionais (preconizada pela empresa). O trabalho prescrito, preconizado por Taylor (1976), tem por finalidade apontar as normas, os protocolos que cada profissional ou grupo profissional deverá seguir para executar determinada tarefa. No taylorismo, o sujeito trabalhador é desprovido de sua subjetividade e é visto como um repetidor de normas, padrões e protocolos, existindo claramente uma separação entre o trabalho intelectual e o manual, esse sistema também exclui a linguagem do ambiente de trabalho.

O trabalho real diz respeito à forma particular de execução, de cada sujeito, de sua atividade. O trabalho real é o espaço da subjetividade de cada trabalhador; é nele que o protagonista da atividade fará ‘uso de si’<sup>7</sup>. Com o enfraquecimento da visão taylorista, o homem, e toda a sua subjetividade começam a se sobressair, a linguagem passa a ganhar um novo enfoque, vista não mais apenas como um suporte do trabalho, mas como elemento essencial na construção da ação (FAÏTA, 2002, p. 47).

Na busca de uma sistematização para os estudos da linguagem em situação de trabalho, Lacoste (1998) formula uma tripartição para a linguagem no âmbito da atividade. Para a autora a linguagem pode ser vista de uma tríplice maneira: linguagem como trabalho; linguagem no trabalho e linguagem sobre o trabalho. A relação entre linguagem e trabalho, introduzida no decorrer desse texto; para explicar o modo como um *software* é capaz de ser suporte da linguagem (escrita, iconográfica ou sonora), e ainda ser um produto mediador da linguagem, tem por base as relações languageiras nas quais a linguagem é vista como uma atividade duplamente orientada.

Assim, com Sampaio (2003, p.12) podemos afirmar que a “linguagem não é apenas uma forma de ação. Mais do que isso, ela é uma forma de trabalho, um trabalho linguístico que se realiza nas práticas languageiras”. E é nessa volta ao ambiente de produção de um *software*, nesse olhar atento para as práticas languageiras dos protagonistas que estão engajados no processo de desenvolvimento de um *software*, que poderemos perceber que estas máquinas, *Quod erat demonstrandum*, descrição de máquinas, carregam em si muito da presença de quem as construiu.

Em geral, os *softwares* são desenvolvidos em empresas que se autodenominam de ‘fábricas de *softwares*’. Uma fábrica de *software*<sup>8</sup>, assim como uma fábrica tradicional, é estruturada de modo similar. Ou seja: os processos, as metodologias, os recursos (humanos e materiais) também se fazem presentes nelas. Em uma fábrica de *softwares* é comum que haja departamentos especializados em determinados processos; esses departamentos planejam, produzem, testam e implantam um *software*. Tudo isso de forma integrada e baseada em inter-relações que tomam efeito no produto *software* final. Além disso, a fábrica de *software* é utilizada para manutenções adaptativas e corretivas em sistemas.

O trabalho em uma fábrica de *software* decorre da colaboração, das inúmeras negociações efetuadas entre os diversos atores envolvidos no processo (engenheiros, *designers*, programadores, gerentes de produto). Esses atores são os responsáveis, por assim dizer, pelo produto pronto, o *software* e suas funcionalidades, pelo seu conteúdo, pelos princípios de interação – o que o consumidor final, o usuário poderá fazer com o *software* –, pela interface do *software* etc.

Em todas essas inter-relações direcionadas a um produto que seja adequado ao uso, está um elo mediador: a linguagem. O filósofo russo, M. Bakhtin (1997, p. 279) já observou que “todas as esferas da atividade humana, por mais variadas que sejam, estão sempre relacionadas com a utilização da língua” e que “não é de surpreender que o caráter e os modos dessa utilização sejam tão variados como as próprias esferas da atividade humana”. As relações/o produto *software* que nascem numa fábrica de *softwares* são exclusivamente orientadas pela linguagem<sup>9</sup>.

O aparecimento de um *software* é sempre motivado em torno de alguma problemática. Eles são desenvolvidos para tornar fácil o que antes era penoso. Ele é sempre orientado a alguém (o usuário em potencial), e sempre projetado por alguém – o produtor do *software*. Nesse sentido, o trabalho em uma fábrica de *software* é sempre de interação e co-construção de ações, seja entre os funcionários da empresa, seja entre terceiros que buscam nela a solução para determinado problema. É pela linguagem, na busca do outro através da linguagem, que os profissionais que desenvolvem *softwares* orientam o processo de fabricação de *software*.

Essa retomada do outro, essa busca da subjetividade do outro acontece, num primeiro momento através de um gênero do discurso chamado de *briefing*. O *briefing* é um gênero que busca organizar a ação, confrontar qual a real necessidade do cliente, ajustá-la,

*Briefing* significa, também, o rol de informações dadas pelo cliente a um profissional da comunicação para que ele possa preparar um projeto ou campanha para sua empresa. O *briefing* representa o conjunto de informações colhidas junto à empresa-cliente, seguindo o roteiro dado, por meio de entrevistas. (...) os dados colhidos permitirão fazer a análise do perfil do cliente e formular o diagnóstico e as propostas de atividades adequadas às suas necessidades de comunicação. O *briefing* comporá, portanto, o ponto de partida, o fundamento de todo o trabalho, que começa pela identificação do objeto de estudo, ou do público-alvo: a empresa-cliente. Em sentido amplo, *briefing* pode significar toda a primeira parte do trabalho. (FRANÇA;FREITAS, 2002 p. 121-122)

Essas informações e coordenadas que orientarão o trabalho da equipe da fábrica é apenas um dos componentes da atividade de trabalho. Entretanto, é o componente que guiará todas as outras tarefas dos protagonistas da atividade. O primeiro contato entre o ator social, que busca uma empresa, e a empresa com esse ator, é de fundamental importância para o bom andamento do processo de fabricação: é através desse contato inicial que a empresa passa a conhecer um pouco do trabalho de quem a procura. Conhecer os pontos fortes e fracos, de definir com a ajuda do cliente, uma estratégia de elaboração do produto que chegue mais perto da necessidade.

Daí a urgência de um profissional que tenha a autonomia e a competência necessária e saiba extrair daquele que busca a fábrica, as suas reais necessidades, para então desenvolver um produto que se aproxime ao máximo do que foi outrora definido. As situações de falas em uma entrevista, em um *briefing*, fazem parte da atividade da fábrica e são necessárias para estabelecer o vínculo indispensável ao desenvolvimento de um *software*. São indispensáveis para formular uma relação diferenciada que permita que o trabalho seja executado.

Em si mesmo, esse primeiro contato é a primeira atividade a ser desenvolvida pelos profissionais da fábrica. E é nesse primeiro contato – no ato de comunicar ao outro –, que a atividade se encontra e ganha sentido. É através do discurso do outro, dos desejos das vontades, das suas inquietações, que o protagonista da atividade recebe e percebe os anseios.

O trabalho dos desenvolvedores, dos *designers*, dos funcionários da fábrica é orientado em direção a um ponto de vista comum. É a fala, a conversa ‘informal’, a fala que busca explicar como a atividade poderá ser construída que está em jogo. As informações coletadas no *briefing* gerarão as soluções dos problemas propostos pelo ator que chega à empresa. Os recursos mobilizados, as tecnologias que melhor sirvam ao problema, as demandas necessárias, as equipes envolvidas para a solução, são definidas a partir desse primeiro contato. O papel fundamental do protagonista da atividade (preposto da fábrica) é de transmitir à equipe todas as expectativas do ator social (a empresa/pessoa física contratante).

A coordenação das ações desencadeadas em uma empresa de *software* são estruturadas em conjunto, estando fortemente ligadas com a situação. As situações de trocas – verbais e não verbais – podem ocasionalmente também ocorrer com outros atores que não estejam alocados, instalados em uma localização física comum. As trocas de atividades entre os protagonistas da fábrica, desse modo, passam a ganhar um novo contorno: a interpretação da atividade do outro, a interpenetração na atividade do outro, estende-se de modo dialógico através de enunciados, não só verbais, mas, sobretudo escritos.

A atividade dos protagonistas da fábrica é também condicionada e direcionada para outro protagonista da atividade que pode ou não estar ocupando o mesmo espaço/tempo que ele. O mais interessante é que nessa

falta de corporeidade aparente<sup>10</sup>, os atos de uns não permanecem estrangeiros aos outros. Ao contrário, os atos de uns complementam a atividade dos outros.

### 3 Da orientação para a atividade

Uma vez finalizado o *briefing*, os protagonistas da atividade começam a ganhar papéis definidos dentro da organização. É comum que cada equipe fique responsável por uma atividade e que esta atividade esteja de certa forma co-dependente da atividade de outra equipe. O engajamento entre os protagonistas – os desenvolvedores da fábrica de *software* – depende de um intenso processo de articulação, de trocas, que se materializam em diversos gêneros como: as listas de discussão, os *e-mails*, os bate-papos em comunicadores instantâneos com e sem agendamento, leitura de documentação etc. Desse modo, a produtividade é co-dependente da qualidade das relações, dos vínculos e da cooperação estabelecida entre as equipes envolvidas no projeto.

A fábrica de *software* é um caso *sui generis*: nela os funcionários utilizam como ferramenta de trabalho, um produto que dará origem a outro produto cuja natureza é a mesma (digital), mas cujo acabamento e função serão outros. É nesse ponto que a apropriação dos saberes (saber usar e saber fazer) penetra de forma mais significativa na atividade.

É ao utilizar o *software* para criar um outro *software* que o protagonista da atividade estabelece relação com esses artefatos. O uso de si, nos termos que Schwartz (2000) assinala, passa necessariamente pela capacidade que o protagonista em uma fábrica de *software* tem de apreender outros discursos, de mobilizar saberes e estabelecer relações. Essas relações, mediadas pela linguagem, são, como aponta Peres (2007), vistas não só entre indivíduos, mas também podem ocorrer entre pessoas e máquinas. A autora sustenta a tese da intersubjetividade e observa que os ícones, os elementos das interfaces, os signos em artefatos informatizados que os protagonistas da atividade usam, são enunciações que precederam um determinado produto, são enunciados de um outro desenvolvedor.

*Hic et nunc*, o trabalho na fábrica evoca sempre a presença de outros protagonistas para a atividade. Essa evocação do outro, essas trocas languageiras determinam um papel relevante para o êxito do trabalho prescrito. Permite não só identificar os problemas que estão sendo colocados, como também permite uma aproximação maior entre os protagonistas da atividade.

A heterogeneidade das ações e a presença de outros protagonistas em um trabalho verdadeiramente coletivo são as marcas mais visíveis em uma fábrica. Essa intersubjetividade, materializada em discursos, em vozes que convoca o outro a todo o momento para atividade são, de certa forma, transpostas para o artefato

computacional que está sendo fabricado. Nas fábricas, os atos e as atividades de uns se entrelaçam nos atos/atividades de outros.

O sucesso do trabalho, do objetivo comum, certamente é dependente, em boa parte, dos usos de si para si e do uso de si para o Outro que o protagonista da atividade é capaz de realizar. Certamente relações mais complexas são travadas quando do desenvolvimento de um *software*. A primeira delas diz respeito à própria consciência da atividade, ao acordo tácito – que se intercambia entre o dito e não dito –, entre os protagonistas. Esse acordo diz respeito à própria consciência da atividade, consciência de que o trabalho não é circunscrito a uma equipe, mas que o trabalho é das equipes. Esse deslocamento da intersubjetividade individual a uma intersubjetividade coletiva repercutirá no próprio artefato que está sendo desenvolvido.

#### **4 Das marcas da intersubjetividade no desenvolvimento de *softwares***

As marcas da intersubjetividade do desenvolvedor nos artefatos computacionais, nos *softwares*, aparecerão mais claramente quando do desenvolvimento das interfaces. A intersubjetividade materializada em vozes, ou seja, a marca ideológica de um eu que se vincula a um lugar social encontra aí sua maior ressonância. Daí o motivo de alguns olharem as interfaces dos *softwares* e enxergarem nelas enunciações de outrem. Discursos que guiam ações de uso, que conectam desenvolvedores e usuários, situados em pólos extremos.

É mais especificamente no trabalho da equipe que irá desenvolver a interface do *software* – *Graphical User Interface* (GUI) –, que as questões relacionadas ao estilo, a intersubjetividades, a posições valorativas, a alteridade, se tornam mais expressivas. Esses profissionais de design têm por missão comunicar, ao usuário do *software*, o que ele poderá fazer com um *software*. A pressuposição de um outro a quem o *software* irá servir interfere significativamente no acabamento do produto. Dessa forma, questões como: a quem a mensagem se destina? Que problemas a interface se propõe a resolver? Como interagir com sistemas para resolvê-los? São pensadas pelos protagonistas da atividade enquanto estão desenvolvendo as interfaces dos sistemas. A ideia de comunicar algo a alguém e fazer com que esse outro, em potencial, assumira uma posição axiológica frente a um chamado que se manifesta em um artefato computacional, constitui um dos trabalhos mais sujeitos a interferência de terceiros dentro da fábrica.

O trabalho da equipe de *design* é um trabalho que é constantemente alvo de juízos de valor – estéticos e ideológicos –, por parte de terceiros, como os gerentes de produto ou mesmo os clientes. Essas avaliações ou coerções externas não estão diretamente ligadas ao domínio da técnica. O saber-fazer, por parte do protagonista, que qualquer atividade evoca, mas que se situa no plano do estilo e da instância ideológica, requer, do

protagonista da atividade – o *designer* –, a capacidade de se adaptar às coerções sociais impostas pelo ambiente. Integra, dessa forma, a coesão e o conflito de opiniões na situação de trabalho.

Assim, o grupo social – os companheiros da jornada –, passam a fazer parte da atividade do profissional que projeta a interface. Essas vozes, essas posições avaliativas, que ao mesmo tempo integram e reorganizam o processo de acabamento de um artefato tecnológico, refletirá no próprio artefato em desenvolvimento.

A interface – que vista por um prisma dialógico –, é um enunciado dirigido a alguém, carrega em si muito da intersubjetividade de quem a cria. Veja, por exemplo, a figura 1, quando a interface do *software* Firefox envia uma enunciação para o sujeito utilizador informando o motivo pelo qual não conseguiu restaurar algumas abas e pede para que este tome uma atitude responsiva, escolhendo entre restaurar ou iniciar nova sessão.

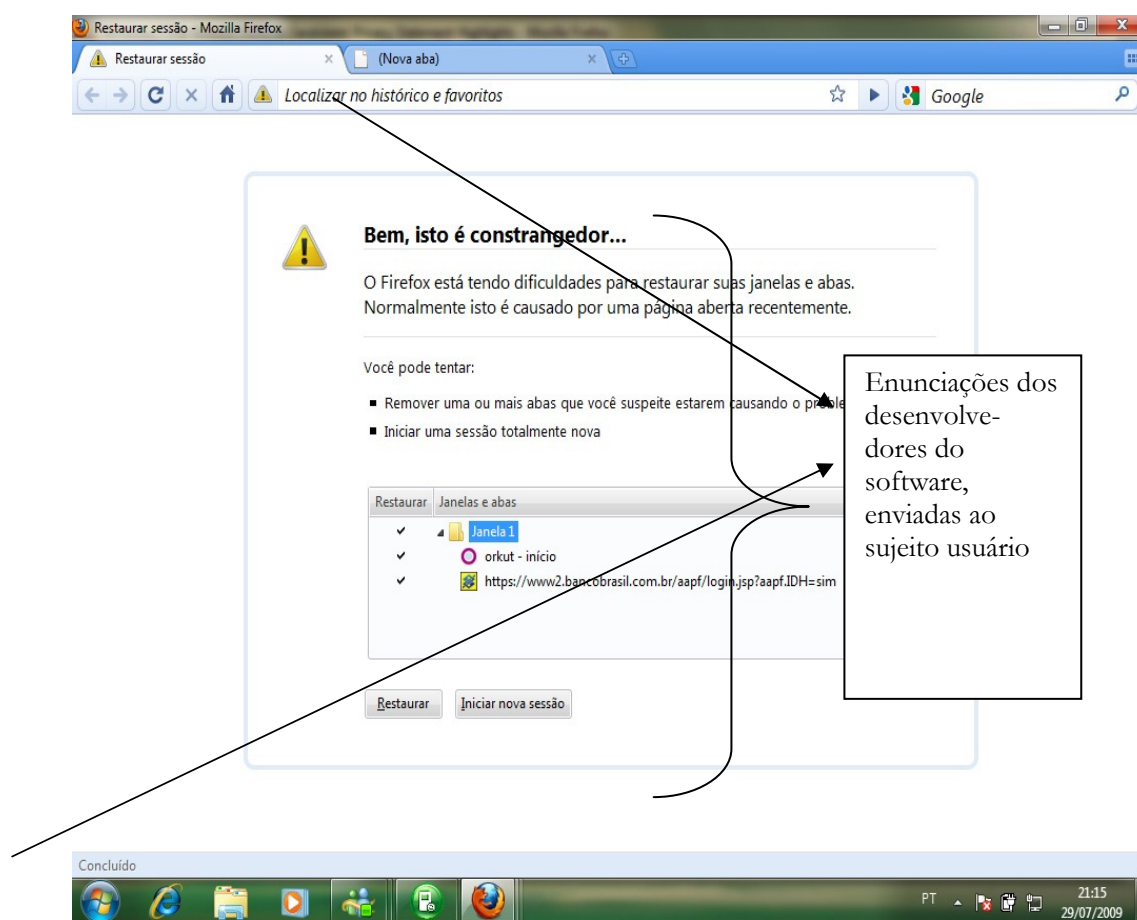


Figura 1 - mensagem dos desenvolvedores na interface *software* Firefox

A interface, assim, está no nível enunciativo enquanto diálogo reflexivo que o protagonista da atividade estabelece com o usuário do *software*. Esse diálogo é materializado através de signos – e.g. o botão restaurar,

iniciar nova sessão, ou ainda o botão colar, copiar, os menus arquivos, ajuda vistos em um *software* como o MS Word –, e também por todos os enunciados que se acoplam a uma interface para guiar ações de uso.

Essas formas de enunciações, que se materializam em uma interface de *software*, constituem-se como material de base sobre o qual o usuário do *software* estabelecerá relação.

Os enunciados dos desenvolvedores, nas interfaces, têm por objetivo mobilizar o usuário do *software*, de orientar suas ações. Essa orientação primária – em primeiro momento facilita o uso de um *software* –, se intercambia a um só tempo com a orientação para com o outro – a comunicação mediada entre dois sociais através de *softwares* que dão origem aos gêneros digitais, como o bater-papo, por exemplo.

Essas formas de enunciações, que se materializam em uma interface de *software*, constituem-se como material de base sobre a qual o usuário do *software* estabelecerá relação. Um bom exemplo dessas marcas de intersubjetividade que os protagonistas da atividade deixam, durante o processo de fabricação de *software*, é as ‘caixas de diálogo’ que aparecem quando o usuário está interagindo com um *software*.

Poderíamos então falar que uma interface de *software* carrega em si um ‘mínimo dialógico’ e que esse mínimo é marcado por uma série de formas de representação do outro que antecedem a criação de um artefato tecnológico. Esse ‘mínimo’<sup>11</sup> pode ser observado também nos embates dialógicos travados entre dois sujeitos situados, através de *softwares* aplicativos, como por exemplo, uma conversa que ocorre através de um comunicador instantâneo como o *Messenger*. Conforme apontam Meira;Peres (2004), este tipo de comunicação, mediada por um artefato tecnológico, é sempre co-dependente da organização sónica da interface.

Para esses autores, o *software* participa desses diálogos, uma vez que os enunciados, proferidos pelos interlocutores, são de certa forma, enunciações dos desenvolvedores e estão submetidas ao funcionamento do próprio *software*, ou mesmo estão embreadas com a própria interface, através de sugestões que aparecem na interface (enunciações dos desenvolvedores).

Dessa forma, o processo de uso de um *software* está diretamente relacionado com o processo de desenvolvimento do *software*. É nessa direção – processo de desenvolvimento/processo de uso –, que a instância humana, a presença de um sujeito historicamente situado, emerge.

Quando evocamos a ideia de que nas interfaces dos *softwares* existem enunciações de outrem dirigidas para alguém (SOUZA, 2009), estamos afirmando que essa pressuposição de um outro interfere significativamente no trabalho prescrito<sup>12</sup> e no acabamento do produto (*software*). O acabamento pode ser estritamente linguístico – ou seja, quais enunciados devem vir na interface para que o sujeito, que irá utilizar o *software*, não tenha problemas de interpretação? – à um acabamento puramente estilístico –, considerado não só como um acabamento do *design* do produto, mas também em relação as marcas linguísticas deixadas nesse produto.

Nesse sentido, o estilo está, de tal forma, relacionado ao conceito de autor/autoria, que para uma análise substancial desse par seria necessário rever as relações constitutivas entre os diversos protagonistas da atividade (os desenvolvedores do *software*) e observar como essa sequência de relações entra em relação dialógica com o outro para quem o *software* é pensando. Assim, as marcas estilísticas em uma interface de *software*, ou melhor, as marcas da gênese estilística, que são marcas deixadas ao longo do processo de fabricação, são uma consequência da interação entre os diversos protagonistas da atividade e da imagem que esses protagonistas fazem do outro a quem o *software* irá servir. Muito embora esse outro só exista, em um primeiro momento, no campo das ideias, de um outro idealizado, essa pressuposição de um outro existe desde o momento de produção ao projeto de trabalho.

## 5 Os programadores e as outras vozes

Dentro da fábrica de *softwares*, um dos trabalhos que está menos sujeito a interferência de terceiros é o dos programadores. Isso se deve ao alto grau de conhecimento técnico especializado – conhecimento de linguagens de programação, de manipulação de *softwares*, de compiladores etc –, que envolve o trabalho desses profissionais. O nível da técnica e do conhecimento, nessa categoria profissional, sobressai-se sobre o nível do estilo e do subjetivo. Enquanto para os *designers* a preocupação central é construir interfaces que dialoguem<sup>13</sup> com humanos de forma mais fácil possível, os programadores têm, por missão, dentro da fábrica, construir códigos que permitam com que o *software* execute as funções necessárias para desenvolver determinada tarefa<sup>14</sup>. Enquanto os *designers* criam vozes para que sejam respondidas pelos usuários, programadores respondem a vozes criadas por outros profissionais.

Quando os fluxos dessas enunciações, que estão materializadas nos *softwares*, não são interrompidos – ou por incapacidade do *software* em executar uma ação ou por incapacidade de quem criou e não conseguiu alocar nele todas as possíveis vozes –, é que podemos perceber como a fluidez no uso de um sistema pode comprometer o trabalho prescrito. Assim, quando o *software* não consegue ser mais suficientemente responsivo<sup>15</sup>, um terceiro é evocado para quem o protagonista da atividade direciona seus enunciados. Esse outro, em potencial, é sempre um dos protagonistas da atividade (companheiro de jornada). O *software* passa então a se constituir como uma terceira pessoa da relação. Nesse ponto da atividade os conceitos de autoria e uso se imbricam.

Quando do momento em que as equipes imbuídas de uma missão (os programadores de desenvolverem o código que permita que o *software* funcione e os *designers* imbuídos na missão de tornar aquele *software* operável

por humanos), se veem na conclusão da tarefa, chega a hora das atividades que foram desenvolvidas se integrarem em uma só. É nessa hora que as trocas linguageiras revelam não só o papel que os atores desempenham em determinada atividade, mas, às vezes, também servem para reforçar o vínculo entre eles.

É na negociação direta, com o outro protagonista, que as melhores soluções são pensadas para finalizar o primeiro processo do trabalho. É neste contexto que a dimensão coletiva do trabalho coletivo, na atividade, começa a ganhar forma: o que deve ser feito e como deve ser feito esse alinhamento entre o código de programação e o *design* do produto passa necessariamente pela dimensão do coletivo.

Por vezes o trabalho prescrito, nessa atividade, é negociado e renegociado: às vezes por limitação técnica, outras vezes por questões estilísticas ligadas ao *design* do produto. Nesse contexto, o papel regulador, do gerente da fábrica, é de certa maneira constitutivamente regulador da atividade. Na realidade, para o trabalho prescrito, a atividade corresponde, em boa medida, aquilo que não se pode fazer, ou ao que se deveria fazer, mas que, por alguma limitação, seja tecnológica ou estética, não se faz.

## Considerações Finais

Na orientação para o outro, o *software*, quando é finalizado, é submetido a uma bateria de testes com usuários leigos. Em geral esses testes servem para verificar se de fato o produto *software* responde bem as expectativas do outro, se o produto *software* de alguma forma apresenta lacunas. É nessa atividade, impelida a um terceiro, que o produto poderá ser melhorado.

Nesse processo, o outro, que até então era idealizado passa a assumir uma corporeidade e interferir significativamente no processo de construção de um *software*. Nesse sentido, construir um *software* é sempre uma re-construção. Com isto não estamos dizendo que a atividade, realizada na fábrica, que a atividade que precede o momento do teste de *software*, se encontra em um segundo plano, pois são nessas realizações que surgem novas formas ou maneiras de dar acabamento ao trabalho prescrito. Certamente, um olhar externo sobre o produto fabricado, às vezes ajuda os protagonistas da atividade a repensarem a própria atividade, de ajustá-la a real necessidade do cliente. Os protagonistas da atividade, desse modo, passam a experimentar duas situações de trabalho, uma regulada – a planejada, a concebida, a que deveria seguir determinados protocolos –, e a experiência vivida.

Ao separar as etapas de fabricação de um *software* podemos vislumbrar as vozes de cada um dos autores envolvidos no processo de fabricação: desde o cliente, que chega com certas expectativas, passando pelo gerente de projeto que tem por missão traduzir para equipe da fábrica, o que o cliente deseja. Essa teia de inter-relações

em sua maioria vividas através das trocas languageiras, em situação de trabalho, leva-nos a questionar se as atividades de trabalho, em especial numa fábrica de *softwares*, não seriam atividades para os outros.

Assim, como bem aponta Clot (1999<sup>16</sup> apud Santos, 2006), a atividade em um fábrica “(...) é triplamente dirigida (...). Na situação vivenciada, ela não é somente dirigida pela conduta do sujeito ou dirigida através do objeto da tarefa, ela é também dirigida para os outros”. Esse trabalho dirigido ao outro toma forma e constrói-se quando da criação das interfaces desses *softwares*: é a partir do contato com a interface do *software* que o outro social –, a quem o trabalho foi endereçado, para quem o produto foi pensando – que a atividade dos protagonistas da fábrica ganha e começa a produzir sentido.

A heterogeneidade das condutas dos atores, engajados no processo de fabricação de um *software*, parece dar, ao produto final ‘*software*’, certas particularidades. O papel do fator humano, na criação e co-participação de um artefato tecnológico, revela o quanto de nós existe nessas máquinas, vistas como mediadoras da linguagem e construídas em uma linguagem de programação.

Percebemos, então, como o processo de construção de um *software* é tomado e formado a partir dos discursos de outrem. Esse encontro dialético e, porque não, dialógico, com o outro, possibilita integrar o nível da técnica ao nível da subjetividade. Saber dominar uma ferramenta, saber programar em determinada linguagem de programação constitui apenas um dos componentes da atividade dos protagonistas em uma fábrica.

O trabalho em uma fábrica de *software*, aqui descrito, teve como ponto de partida a ideia que as relações entre linguagem e trabalho não se resumem apenas as ações languageiras em ambiente de trabalho. Pensamos a linguagem como Lacoste (1998), ou seja, vimos que, em uma fábrica, a linguagem pode ser observada de três formas: sobre/no/como trabalho. Quando nos voltamos para o protagonista da atividade (sujeito desenvolvedor), verificamos que, no *software*, o estilo é também pré-dado e esse estilo pré-dado, por vezes, limita o uso desse *software*. Verificando o modo como um *software* é fabricado, percebemos que as relações dialógicas o atravessam, desde o processo de fabricação até o processo de uso.

Assim, percebemos que ao produzirem as interfaces os desenvolvedores deixam nelas formas marcadas e não marcadas de suas presenças, o que chamamos de marcas da intersubjetividade. Nessa fase da investigação o que descrevemos foram as formas de representação do outro (a alteridade). A análise desse outro que ocupa um papel primordial no desenvolvimento de *softwares* levou-nos a indagar como se dá a construção do estilo tanto quando do uso de um *software*.

Entretanto, salientamos que as marcas de representação do outro, aqui descritas, foram guiadas a luz dos postulados da ergolinguística, ou seja, o outro ao qual estamos nos referindo diz respeito ao sujeito trabalhador e toda sua individualidade e as marcas as quais nos referimos dizem respeito ao produto de seu trabalho ou seja, aos atos languageiros em situação de trabalho.

E é ao observar o processo de desenvolvimento de um *software* que podemos dizer ser o *software* um suporte. Ele é ao mesmo tempo o lugar onde a escrita está materializada, o lugar onde mantemos relação com a escrita, um produto que consegue ser um portador do texto e um objeto capaz de executar funções mediadas por humanos (salvar, enviar), bem como executar funções que independem da interferência de humanos como, por exemplo, comunicar-se com outros *softwares*.

## Referências bibliográficas

BAKHTIN, M. *Estética da criação verbal*. São Paulo: Martins Fontes, 1997.

BARBOSA, S.D.J.; de Souza, C.S.; Paula, M.G. *The Semiotic Engineering Use of Models for Supporting Reflection-in-Action*. In: HCII2003 - HCI International, Creta. Human-Computer Interaction: Theory and Practice (Part I). Mawah, NJ : Lawrence Erlbaum, 2003.

BRAUDE, Eric. *Projeto de software: da programação à arquitetura: uma abordagem baseada em Java*. Porto Alegre: Bookman, 2005

BROOKSHEAR, J.Glenn. *Ciência da Computação: Uma visão abrangente*. 5ª Edição. Porto Alegre: Bookman Companhia Editora, 2001

CLOT, Y. *La fonction psychologique du travail*. Paris: PUF. 1999

FAÏTA, D. *Análise das práticas linguageiras e situações de trabalho: uma renovação metodológica imposta pelo objeto*. In: SOUZA-E-SILVA, M. C.; FAÏTA, D. (orgs.). *Linguagem e trabalho: construção de objetos de análise no Brasil e na França*. Trad. Ines Polegatto e Décio Rocha. São Paulo: Cortez. 2002

FRANÇA, Fábio; FREITAS, Sidinéia Gomes. *Manual da qualidade em projetos de comunicação*. São Paulo: Pioneira, 2002.

FLOYD, THOMAS. *Sistemas Digitais: fundamentos e aplicações* 9 ed. Bookman. Porto Alegre, 2007

LACOSTE, M. *Fala, atividade, situação*. In: DUARTE, F.; FEITOSA, V. (orgs.). *Linguagem e trabalho*. Rio de Janeiro: Lucerna, p. 15-36. 1998

- MAIBAUM, Tom. *Fundamental approaches to software Engineering*. Third International Conference, FASE 2000 held as part of the joint European Conferences on theory and practice of software, Berlin, Germany, March/Apri 2000
- MEIRA, Luciano; Peres, Flavia (2004): *A dialogue-based approach for evaluating educational software*. In *Interacting with Computers*, 16 (4) pp. 615-633
- NOUROUDINE, A. 2002. *A linguagem: dispositivo revelador da complexidade do trabalho*.  
In: SOUZAESILVA, M. C.; FAÏTA, D. (orgs.). *Linguagem e trabalho: construção de objetos de análise no Brasil e na França*. Trad. Ines Polegatto e Décio Rocha. São Paulo: Cortez.
- PERES, Flávia. *Diálogo e autoria: do desenvolvimento ao uso de sistemas de informação*. 204 f, Tese de Doutorado, Departamento de Psicologia, Curso de Pósgraduação em Psicologia Cognitiva, Universidade Federal de Pernambuco. 2007
- PONTES, A.M.; Leitão, C.F.; Barbosa, S.D.J.; de Souza, C.S.; Quental, V.S.T.D.B. *Estudo do impacto do design e das formas de uso sobre a recuperação de informações em fóruns de discussão online*. In: VI Simpósio sobre Fatores Humanos em Sistemas Computacionais - IHC 2004, 2004, Curitiba. VI Simpósio sobre Fatores Humanos em Sistemas Computacionais - IHC 2004: Mediando e Transformando o Cotidiano - Anais do Simpósio. Porto Alegre : SBC, 2004
- PONTES, A.M.; Barbosa, S.D.J.; de Souza, C.S *Organização Conversacional: Inspeção das Representações na WikiPedia*. In: CLIHC2005 - Latin American Conference in Human-Computer Interaction, Cuernavaca, Mexico. Proceedings of CLIHC, 2005.
- REZENDE, Denis Alcides. *Engenharia de software e sistemas de informação* 3.ed. Rio de Janeiro: Brasport, 2005
- SAMPAIO, M.C.H. *A Ergologia e os Estudos da Linguagem e das Práticas Linguageiras em Situações de trabalho*. Recife, Arte Comunicação, n. 8, p.149-167. 2003
- SANTOS, M. *Análise psicológica do trabalho: dos conceitos aos métodos*. Laboreal, 2, (1), 34-41. 2006. Disponível em: <http://laboreal.up.pt/revista/artigo.php?id=37t45nSU54711226516545:581>
- SOUZA, A.G. *Software, hipermídia, hipertexto e gêneros digitais: observações preliminares*. Encontro internacional de texto e cultura – anais – página 42-55. CD-ROM ISSN: 2176-199X. Protexto - UFC, Fortaleza, Ceará 2009.
- SCHWARTZ, Yves *Trabalho e uso de si*. In: *Pró-Posições*. Campinas, Vol,11 n.2 (32), p.34-50. Tradução de Maria Lúcia da Rocha Leão, Revisão Técnica de Maria Inês Rosa, UNICAMP. 2000
- TAYLOR, Frederick Winslow. *Princípios de Administração Científica*. Tradução de Arlindo Vieira Ramos. Editora Atlas, São Paulo. 1976

## Notas

---

<sup>1</sup> **Aguinaldo Gomes de Souza, Prof. Ms**

Mestre em Letras/ Linguística pela Universidade Federal de Pernambuco

Professor da Pós-Graduação em Linguística Aplicada ao Ensino de Línguas- Faculdades Integradas da Vitória de Santo Antão (FAINTVISA)

[aguinaldo@souza.pro.br](mailto:aguinaldo@souza.pro.br) / web site: [www.souza.pro.br](http://www.souza.pro.br)

<sup>2</sup> As reflexões contidas neste artigo fazem parte da dissertação de mestrado “Software: esboço de um estudo para as Ciências da Linguagem” defendida na Universidade Federal de Pernambuco em fevereiro de 2010.

<sup>3</sup> Vale ressaltar que para as considerações feitas sobre a fábrica de *software*, não houve coleta de dados in loco, servimo-nos de nossas memórias enquanto protagonistas da atividade para teorizar sobre o objeto.

<sup>4</sup> A linguagem no trabalho (uma linguagem circundante); A linguagem como trabalho (uma linguagem que faz); A linguagem sobre o trabalho (uma linguagem que interpreta)

<sup>5</sup> Ergolinguística é a palavra utilizada para se referir aos estudos sobre Linguagem e Trabalho

<sup>6</sup> Perspectiva científica e filosófica que surgiu na França, cujo objetivo é estudar o trabalho humano em todas as suas dimensões, levando em consideração as complexidades das situações de trabalho

<sup>7</sup> O uso de si diz respeito a como o trabalhador se engaja na atividade, como e o que ele faz para realizar a atividade, para maiores detalhes a respeito do tema, consultar Schwartz (2000)

<sup>8</sup> A explanação aqui iniciada diz respeito ao desenvolvimento de *softwares* de código fechado

<sup>9</sup> Cabe mais uma vez a ressalva: não confundir linguagem com linguagem de programação. Embora homônimos os termos, o que temos em ambos é apenas uma identidade de nome, e não de substância. Para evitar confusões quando fizermos referência à linguagem em que os *softwares* são criados, faremos com o predicado ‘de programação’ ao final da expressão linguagem

<sup>10</sup> Essa não presença física na empresa, mas uma presença remota, já está se tornando comum nesse meio. Uma dos principais argumentos para os adeptos desse tipo de trabalho é a melhora significativa na qualidade de vida dos colaboradores e nos rendimentos e produtividades elevadas que o trabalho em casa proporciona à empresa. Além disso, para as empresas, os benefícios incluem também menos gastos com os escritórios, já que o espaço não precisa ser grande devido à menor quantidade de funcionários, com o auxílio transporte etc.

<sup>11</sup> Chamamos de mínimo dialógico aquele elemento que faz com que o sujeito, que está utilizando o *software*, passe a tomar uma atitude responsiva frente a um chamado. O botão *fechar*, presente na interface do *software* de e-mail ou de um processador de textos, é um exemplo desse mínimo.

<sup>12</sup> Com isso estamos querendo dizer que a ideia de um outro a quem o software irá servir, interfere significativamente na atividade dos desenvolvedores de *softwares*.

<sup>13</sup> Os *designers* são os responsáveis por desenvolver a forma arquitetônica do *software*

<sup>14</sup> É nesse momento que a forma mecânica do *software* é desenvolvida.

<sup>15</sup> Conforme já apontou Peres (2007)

<sup>16</sup> Clot, Y. *La fonction psychologique du travail*. Paris: PUF. 1999