

UNIVERSIDADE FEDERAL DE PERNAMBUCO  
CENTRO DE ARTES E COMUNICAÇÃO  
DEPARTAMENTO DE LETRAS  
PÓS-GRADUAÇÃO EM LETRAS

SOFTWARE: ESBOÇO DE UM ESTUDO PARA AS  
CIÊNCIAS DA LINGUAGEM

AGUINALDO GOMES DE SOUZA

RECIFE, 2010

# **Livros Grátis**

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

UNIVERSIDADE FEDERAL DE PERNAMBUCO  
CENTRO DE ARTES E COMUNICAÇÃO  
DEPARTAMENTO DE LETRAS  
PÓS-GRADUAÇÃO EM LETRAS

## SOFTWARE: ESBOÇO DE UM ESTUDO PARA AS CIÊNCIAS DA LINGUAGEM

AGUINALDO GOMES DE SOUZA

Dissertação apresentada ao Programa de Pós-graduação em Letras da Universidade Federal de Pernambuco como requisito parcial para obtenção do Grau de Mestre em Linguística.

Orientadora: Prof<sup>a</sup> Dr<sup>a</sup> Maria Cristina Hennes Sampaio

RECIFE, 2010

**Souza, Aguinaldo Gomes de**

**Software: esboço de um estudo para as ciências da linguagem / Aguinaldo Gomes de Souza. – Recife: O Autor, 2010.**

**134 folhas: il., fig.**

**Dissertação (mestrado) – Universidade Federal de Pernambuco. CAC. Letras, 2010.**

**Inclui bibliografia.**

**1. Linguística. 2. Software. 3. Subjetividade. Título.**

**801  
410**

**CDU (2.ed.)  
CDD (22.ed.)**

**UFPE  
CAC2010-58**

ATA DA REUNIÃO DA COMISSÃO EXAMINADORA PARA JULGAR A DISSERTAÇÃO INTITULADA: "**SOFTWARE: ESBOÇO DE UM ESTUDO PARA AS CIÊNCIAS DA LINGUAGEM**", DE AUTORIA DE: **AGUINALDO GOMES DE SOUZA**, ALUNO DESTES PROGRAMA DE PÓS-GRADUAÇÃO EM LETRAS.

O julgamento ocorreu às 14h do dia 10 de fevereiro de 2010, no Centro de Artes e Comunicação/UFPE, para julgar a Dissertação de Mestrado intitulada: *Software: Esboço de um Estudo para as Ciências da Linguagem*, de autoria de Aguinaldo Gomes de Souza, aluno deste Programa de Pós-Graduação em Letras. Presentes os membros da comissão examinadora: **Prof.ª. Dr.ª. Maria Cristina Hennes Sampaio** (Orientadora), **Prof.ª. Dr.ª. Evandra Grigoletto**, **Prof. Dr. Júlio César Rosa de Araújo**. Sob a presidência da primeira, realizou-se a arguição do candidato. Cumpridas as disposições regulamentares, foram lidos os conceitos atribuídos ao candidato: Prof.ª. Dr.ª. Maria Cristina Hennes Sampaio: **Aprovado**, Prof.ª. Dr.ª. Evandra Grigoletto: **Aprovado**, Prof. Dr. Júlio César Rosa de Araújo: **Aprovado**. Em seguida, a Prof.ª. Dr.ª. Maria Cristina Hennes Sampaio comunicou ao candidato Aguinaldo Gomes de Souza, que sua defesa foi aprovada pela comissão examinadora. E, nada mais havendo a tratar eu, Jozaiás Ferreira dos Santos, Secretário do Programa de Pós-Graduação em Letras, lavrei a presente ata que assino com os demais membros da comissão examinadora.

Recife, 10 de fevereiro de 2010.

*Jozaiás Ferreira dos Santos*  
- Prof.ª. Maria Cristina Hennes Sampaio  
- Araújo  
- Evandra Grigoletto

A banca, por unanimidade, indica a dissertação para publicação.

Para  
Maria Helena  
e Eurico Gomes (*in memoriam*)

## AGRADECIMENTOS

À Profª Drª Maria Cristina Hennes Sampaio, pela orientação segura e competente bem como pelas aulas na disciplina de Ergolinguística, as quais trouxeram enorme contribuição tanto para minha formação geral enquanto sujeito aprendiz, quanto para o desenvolvimento desse trabalho.

Ao Professor Dr. Júlio César Araújo pelos diálogos travados comigo durante o período em que eu ainda dava meus primeiros passos na graduação e por aceitar compor a banca de avaliação. Sua presença muito me honra.

À professora Drª Ângela Paiva Dionísio por muito gentilmente ter aceitado participar da banca de avaliação bem como pelas contribuições na pré-banca.

À professora Drª Evandra Grigoletto por sua participação em minha Banca de Defesa

Ao professor Dr. Marcos Galindo Lima por ter aceitado ser um dos avaliadores externos

A todos que fazem o programa de Pós-Graduação em Letras da Universidade Federal de Pernambuco, representados no exercício do ano de 2010 pela Professora Dra. Ângela Paiva Dionísio.

Aos professores: Drª Nelly Carvalho, Drº Marlo Pessoa, Drª Gilda Lins (*in memoriam*), Drª Virginia Leal, Drª Kazue Saito e Drª Doris Arruda pela formação no ano de 2008.

À Gilberlande Pereira dos Santos por ter semeado, ainda na graduação, a semente que iria germinar, dando origem à realização desse trabalho.

Aos professores das Faculdades Integradas da Vitória de Santo Antão pelo exemplo de profissionalismo, aqui representados nominalmente por: Hugo Monteiro, Cíntia Tavares, Verônica Campos, Edilene Soares e Flávia Ferreira

Aos meus amigos: Bruno, Hugo e Eduardo, pelo carinho e incentivo. Também pelo fato de serem amigos.

À Marco Aurélio que desde sempre esteve torcendo pelo meu trabalho, pelo apoio que sempre deu. Todo o meu afeto.

À CAPES pelo auxílio financeiro durante o curso.

Aos meus colegas de curso no ano de 2008, representados aqui nominalmente por: Carlos, Leonardo, Cleber, Ludmila, Misael, Sônia, Margareth, Daniel e Edigar.

## LISTA DE FIGURAS

Figura 1 Mozilla Firefox customizado .....	33
Figura 2 Internet Explorer .....	35
Figura 3 Apple II .....	42
Figura 4 WordPerfect.....	43
Figura 5 Ms-Dos .....	44
Figura 6 Lisa -primeira interface baseada em janelas. ....	50
Figura 7 interface baseada em linhas de comando .....	51
Figura 8 janelas .....	53
Figura 9 o <i>software</i> de e-mail acoplado no software de navegação Mozilla.....	58
Figura 10 <i>software</i> Internet Explorer .....	61
Figura 11 mensagem dos desenvolvedores na interface <i>software</i> Firefox.....	70
Figura 12 topologia das redes .....	77
Figura 13 diferença entre ponte e roteador .....	78
Figura 14 protocolos de redes .....	80
Figura 15 primeiro <i>software</i> de navegação na rede Internet: Mosaic.....	81
Figura 16 webmail .....	83
Figura 17 signos e função .....	90
Figura 18 ‘tela do webmail’ .....	93
Figura 19 formato de um e-mail.....	101
Figura 20 estruturação de um e-mail.....	102
Figura 21 processo de envio de e-mail.....	103
Figura 22 user agent.....	104
Figura 23 área de texto.....	105
Figura 24 <i>software</i> cliente de e-mail.....	106
Figura 25 <i>software</i> reconhecido como ‘e-mail’ acoplado ao Firefox.....	107
Figura 26 troca simbólica entre desenvolvedor e usuário.....	112

## RESUMO

No presente trabalho consideramos que o *software* muito mais que servir de aporte para textos, hipertextos ou gêneros, configura-se em estratégia gradual de construção de sentidos. Para sustentar tal argumento separamos o *software* em dois estados: o estado processo e o estado produto, sendo o último objeto de nossa análise. Analisou-se o modo como um *software* é fabricado, procurando compreender e descrever como os processos sequenciais que formam um *software* (a interface, funcionalidades, base hipertextual e o corpo sígnico), o tornam um suporte. Nesse sentido, observamos e descrevemos como a interface do *software*, interage com o sujeito utilizador, ao enviar chamados (enunciações que aparecem na interface) dos autores/desenvolvedores desses *softwares*. Foram também observadas e descritas as relações de alteridade que se estabelecem entre os protagonistas da atividade, interface-sujeito-utilizador com base nos postulados teóricos da Teoria Dialógica da Linguagem (BAKHTIN, 1997) e da Ergolinguística (SCHWARTZ, 2002 a;b e FAÏTA, 2002). Como caminho metodológico de análise revisitou-se a origem e evolução dos *softwares* enquanto sistemas operacionais, com suas respectivas interfaces, bem como os *softwares* aplicativos. Embora sabendo da interdependência dos programas, optamos pela distinção *software* sistema operacional e *software* aplicativo, detendo-nos no segundo para evidenciar a heterogeneidade enunciativo-discursiva que atravessa o desenvolvimento de um artefato digital. Verificamos ainda aspectos relativos à subjetividade dos sujeitos envolvidos no processo de fabricação de *software*, concluindo que os *softwares* carregam em si, na sua interface, formas marcadas e não-marcadas da presença de Outrem.

(237 palavras)

Palavras-chave: Software; interface; signos; intersubjetividade

## RESUMEN

En ese trabajo, consideramos que el software mucho más que servir de aporte para textos, hipertextos o géneros, se constituye en una estrategia gradual de construcción de sentidos. Para sostener tal raciocinio separamos el software en dos estados: el estado proceso y el estado producto. Para fines de análisis llevamos en consideración sólo el segundo. Tras esa separación encaminamos nuestro mirar para el modo como un software es fabricado. Con eso, buscamos comprender y describir los procesos que forman un software, a saber: la interfaz, las funcionalidades, la estructura hipertextual y los signos. En ese sentido, observamos y describimos como la interfaz del software, interage con el sujeto utilizador, al enviar llamados (enunciaciones que aparecen en la interfaz) de los autores/desarrollistas de esos softwares. Bien como observamos y describimos las relaciones de alteridad que se establecen en la relación a los protagonistas de la actividad, interfaz-sujeto-utilizador. Tomamos como basamento teórico los postulados de Bajtín (1997) y las ideas advenidas de la Ergolinguística (SCHWARTZ, 2002 a;b y FAÏTA, 2002). Como camino para análisis, optamos por visitar el origen de las primeras computadoras bien como la origen y evolución de los softwares sistemas operativos con sus respectivas interfaces así como los softwares aplicativos. Aunque sabiendo de la interdependencia de los programas, separamos software sistema operativo de software aplicativo a fin de nos detengamos sólo en el segundo. Buscamos de esa forma, poner en evidencia la heterogeneidad discursiva que se atraviesa en el desarrollo de un artefacto digital. Nosotros concluimos que esos softwares cargan en sí, en su interfaz, mucho de nuestra presencia.

(260 palabras)

Palabras-llave: software, interfaz, signos, intersubjetividad

## SUMÁRIO

LISTA DE FIGURAS.....	vi
RESUMO.....	vii
RESUMEN.....	viii

INTRODUÇÃO.....	11
-----------------	----

### CAPÍTULO 1

#### BREVE RETOMADA SOBRE A QUESTÃO DO SUPORTE NAS CIÊNCIAS HUMANAS; FUNDAMENTAÇÃO, METODOLOGIA E DESCRIÇÃO DE UM CORPUS PARA ANÁLISE QUALITATIVA

1 Os suportes da escrita como objetos de estudos.....	17
1.1 A tela como suporte da escrita.....	23
1.2 O <i>software</i> como lugar da inscrição da escrita.....	26
1.3 O Círculo bakhtiniano e a Ergolinguística.....	28
1.4 Constituição e descrição dos <i>softwares</i> .....	32

### CAPÍTULO 2

#### O SISTEMA COMPUTACIONAL E A METALINGUÍSTICA

2 Linguagem de programação, <i>hardware</i> e Sistemas operacionais.....	39
2.1 O que são <i>softwares</i> .....	46
2.1.1 A interface do <i>software</i> .....	48
2.2 Um olhar da metalinguística sobre <i>softwares</i> e interfaces.....	55
2.3 Como os <i>softwares</i> são fabricados: um olhar ergolinguístico.....	62
2.3.1 Da orientação para a atividade.....	67
2.3.2 Das marcas da intersubjetividade no desenvolvimento de <i>softwares</i> .....	68
2.4 Os programadores e as outras vozes.....	71

### **CAPÍTULO 3**

#### **DA NOÇÃO DE *SOFTWARE* COMO OBJETO SÍGNICO; A CONSTRUÇÃO DE SENTIDOS NOS *SOFTWARES* E OS PONTOS DE ALTERIDADE**

3 Implicações para o estudo dos gêneros e dos hipertextos.....	76
3.1 Os signos e os <i>softwares</i> .....	84
3.2 Signos na interface do <i>software</i> . ....	88
3.3 As funcionalidades dos signos na interface do <i>software</i> . ....	91
3.4 Hipermodalidade.....	96

### **CAPÍTULO 4**

#### **CONSIDERAÇÕES SOBRE A INTERAÇÃO COM *SOFTWARES*. APLICAÇÃO DA BASE TEÓRICA DESENVOLVIDA AO LONGO DESSE ESTUDO.**

4 Aplicação da base teórica.....	101
4.1 Do usuário e do <i>software</i> .....	108
4.2 Relações de sentido. ....	112
<b>5. CONCLUSÕES.</b> .....	117
5.1 Conclusões e apontamentos futuros. ....	118

<b>REFERÊNCIAS.</b> .....	123
---------------------------	-----

## INTRODUÇÃO

'Haja hoje para tanto ontem'  
**Paulo Leminski**

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 observar e descrever o modo de funcionamento 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*. Assim, acreditamos que o presente estudo possa contribuir para suscitar o início de um profícuo debate sobre o tema.

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 a da rede social do Orkut ou mesmo por programas mensageiros como o Messenger. Com isso queremos destacar que se faz urgente, no campo da Ciência Linguística, uma teorização sobre o *software*.

Nesse sentido, no presente trabalho, temos como objetivo geral analisar os processos sequenciais que engendram um *software*, ou seja, compreender a interface, as funcionalidades, a base hipertextual que lhe é própria bem como compreender o corpo de signos que o constitui. Estaremos atentando, assim, não só para o seu todo (sua forma arquitetônica), mas, mais ainda, para os aspectos particulares do *software*.

Nessa direção iremos (1) descrever como a interface do *software*, interage com o sujeito utilizador, ao enviar chamados (enunciações que aparecem na interface) dos autores/desenvolvedores desses *softwares*; iremos observar e (2) descrever também as relações de alteridade que se estabelecem na relação dos protagonistas da atividade – interface-sujeito-utilizador –, e (3) interpretar os efeitos de sentido produzidos nos enunciados concretos materializadas em uma interface de *software*. Assim, no presente trabalho iremos

- 1- observar, descrever e interpretar as marcas da gênese estilística, deixadas ao longo do processo de fabricação, entre os diversos protagonistas da atividade, bem como a imagem que esses protagonistas fazem do outro a quem o *software* irá servir;

- 2- observar, descrever e interpretar como o outro interfere significativamente no trabalho prescrito e no acabamento do produto *software*;
- 3- observar e descrever de que forma os aspectos arquitetônicos e mecânicos, que formam o *software* – interface, funcionalidades, base hipertextual, signos –, influenciam o processo formativo dos hipertextos, enunciados e gêneros digitais.

Uma análise dessa natureza, no entanto, não pode deixar de fazer um duplo movimento teórico. Sendo assim, para situar as observações sobre a interface tomaremos como base os postulados teóricos da Teoria Dialógica da Linguagem do Círculo de Bakhtin, a saber: dialogismo, enunciado e enunciação, signos. Tais postulados serão utilizados aqui na confluência da Ergolinguística, enquanto disciplina dedicada ao estudo do trabalho humano expresso pela linguagem enquanto atividade (trabalho), forjada em situação de trabalho real. Pretende-se, pois verificar, nas interfaces, as formas de presença do outro, formas que se fazem representar através de signos e enunciados encontrados nas interfaces.

Assim, partimos da hipótese que tais enunciados deixados nas interfaces dos *softwares*, enunciados que servem para que sujeitos historicamente situados possam se guiar, enunciados que guiam ações de uso e que governam o uso de um *software*, dizem respeito não só ao diálogo instaurado entre os designers dos *softwares* e o usuário, mas dizem respeito às relações entre um eu (desenvolvedores, designers, engenheiros, cientistas da computação, gerentes de fábricas de *softwares*) e um outro (o ‘nós’ que corresponde à pessoa na qual podem desaparecer todos os outros, o ‘eu’ inclusive<sup>1</sup>).

Entretanto, para construção da proposição preliminar cuja demonstração prévia é necessária para demonstrar a tese principal<sup>2</sup> que pretendemos advogar, no Capítulo 1 desse trabalho iremos fazer uma breve retomada sobre a questão do suporte nas ciências humanas. Assim, iremos visitar alguns estudos que tratam do

---

<sup>1</sup> O enunciado entre parênteses é de Sampaio (2002)

<sup>2</sup> A tese que iremos tratar nesse trabalho e que detalharemos melhor na página 26 é a de que o *software* é o lugar da inscrição da escrita.

suporte no ambiente impresso e tipográfico. Com isso poderemos observar que os objetos que serviram de inscrição para escrita condicionaram, de alguma forma, os usos da escrita. Iremos observar também que são tais objetos que nos permitiram fazer uso da escrita.

Ainda no Capítulo 1 veremos que os primeiros estudos sobre os gêneros digitais e sobre os hipertextos sequer faziam menção ao *software* ser um produto que aporte os (hiper)textos e os gêneros em ambiente digital. Tais estudos, quando tratavam do objeto que servia para inscrição da escrita em ambiente digital, creditavam, à tela, o *status* de suporte. Nesse capítulo observaremos também que nas primeiras investigações havia a preocupação de encontrar pontos semelhantes entre o ambiente digital e o impresso. Assim, iremos discutir questões de ordem teórica e a escolha do método e do *corpus* escolhido para ser analisado.

Após situar o quadro geral dos estudos que tratam da questão do ‘suporte’ bem como situar o leitor sobre nossa proposta de análise, fundamentação teórica e metodologia, iremos iniciar o Capítulo 2. Nele trataremos do sistema computacional. Inicialmente faremos uma breve retomada sobre os *softwares* sistemas operacionais para só então nos determos nos *softwares* aplicativos. Julgamos relevante fazer essa retomada uma vez que é o sistema operacional que controla todo o computador, incluindo os *softwares* aplicativos, objetos de nosso estudo.

Assim poderíamos dizer que o Capítulo 2 trata da questão de produção: (a) discutem-se os modos de fabricação de *softwares*, tais como: linguagens de programação, *hardware* e sistemas operacionais; conceituaremos ainda o *software* no âmbito das ciências da linguagem, estudando sua interface para, em seguida, nos determos no modo de fabricação de *softwares*. Assim, veremos que (b) ao produzirem as interfaces os desenvolvedores deixam nelas formas marcadas e não marcadas de suas presenças, o que chamamos de marcas da subjetividade. Nessa fase da investigação o que estamos buscando é identificar e descrever 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*<sup>3</sup>.

Após verificarmos que nos *softwares* existem marcas da subjetividade, desenvolveremos o Capítulo 3 a fim de analisar detalhadamente a interface do *software*. Para tanto iremos decompô-la em micro-unidades: signos e enunciações, detendo-nos especialmente no primeiro. Em relação aos primeiros, faremos um breve apanhado das escolas e doutrinas que tratam da questão, situando nossa filiação às ideias bakhtinianas sobre signos, ou seja, como ponto de tensão entre um eu e um outro. Ao abordarmos os signos pela óptica bakhtiniana, dirigimos nosso olhar para um fenômeno cunhado de hipermodalidade.

O Capítulo 4 desse trabalho será dedicada à aplicação da base teórica, desenvolvida ao longo de nosso estudo, à utilização do *software*, ou seja, como um sujeito historicamente situado interage com *softwares*. Para tanto, levamos em consideração as formas de representação da presença do outro, que descrevemos na segunda parte do trabalho, bem como a construção da alteridade que se materializa em signos na interface do *software*. Vimos, desse modo, que os processos sequenciais que engendram um *software* (as funcionalidades do *software*, os signos, as enunciações presentes em uma interface, a base hipertextual, a base hipermediática) colaboram para com o processo de uso de um *software*, seja em interações mediadas por outros *softwares* que originam gêneros digitais, seja em interações com outros *softwares*.

Nas conclusões faremos uma breve retomada do percurso teórico inicial, tirando algumas conclusões e apontando possíveis lacunas sobre os *softwares* que servirão de base para futuros questionamentos sobre o lugar de inscrição da escrita em ambiente digital.

---

<sup>3</sup> As marcas de representação do outro, aqui analisadas, serão 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. Embora tenhamos ciência que uma análise a luz das ideias de Authier-Revuz (1998) em nada inviabilizaria as observâncias sobre essa questão.

## **CAPÍTULO 1**

Breve retomada sobre a questão do suporte nas ciências humanas; fundamentação, metodologia e descrição de um *corpus* para análise qualitativa.

## 1 Os suportes da escrita como objetos de estudos

Como observa Araújo (2008), é inegável o fato de que a escrita é uma tecnologia que tem necessidade de um suporte e a relação entre eles constitui uma perfeita simbiose. Nesse sentido, reiteramos Marcuschi (2005, p.26) quando sugere que a escrita é “um modo de produção textual-discursiva para fins comunicativos com certas especificidades materiais e caracterizada pela sua constituição gráfica”. Como bem salienta Marcuschi (2005), ao utilizarmos a expressão ‘gráfica’ não estamos apenas equiparando a escrita com a sua forma de realização alfabética, mas, antes, estamos generalizando, incluindo, na concepção de escrita, os modos de realização alfabéticos, imagéticos e ideográficos, entre outros.

Todos os objetos que servem para inscrição da escrita são indispensáveis para o reconhecimento do gênero que nele é fixado, incluindo o acesso e o arquivamento desses textos, desses enunciados inscritos os quais são dependentes de tais objetos. Foi assim com o códice – predecessor do livro impresso – que permitiu o nascimento do volumem. Foi assim com o livro impresso que se configura como um suporte para os mais diferentes gêneros<sup>4</sup>.

O mundo impresso, o mundo dos textos impressos e todos os objetos impressos que nos são familiares – e.g. uma revista impressa, um cardápio, um dicionário impresso, um outdoor etc –, são produtos de uma tecnologia que permitiu, ao homem, avançar de um estágio da existência a outro estágio. Esse mundo passou, e ainda passa, por inúmeras modificações. Muito dessas modificações deve-se ao aparecimento do primeiro computador<sup>5</sup>. Computadores, telefones celulares, *iphones*, câmeras fotográficas digitais, *palm*s, calculadoras digitais, MP10/MTV *playes*, os terminais de auto-atendimento bancários, etc., são máquinas que, de um modo ou de outro, fazem parte da existência do homem contemporâneo

---

<sup>4</sup> Livro enquanto suporte de gêneros foi tratado por Marcuschi (2003) no ensaio ‘A questão dos suportes dos gêneros textuais’. E também por Bezerra, (2007).

<sup>5</sup> Consideramos aqui como o primeiro computador eletromecânico o MARK I, desenvolvido em 1944. Por ser este o primeiro computador que era controlado por um programa (*software*). Embora ainda seja possível recuar um pouco mais e visualizar a máquina de Hollerith que funcionava com o auxílio de cartões perfurados. Poder-se-ia dizer que esses cartões perfurados que eram lidos por máquinas, são os predecessores dos atuais *softwares* digitais que conhecemos.

e que estão tão entranhadas em nossa cultura letrada que já não se pode prescindir da existência delas.

Os fundamentos científicos que permitiram o desenvolvimento dessas máquinas são um pouco mais antigos, remontando centenas ou milhares de anos antes de seu aparecimento. Não obstante o valor dado à máquina física – o *hardware* –, nesse trabalho iremos nos deter na máquina digital que funciona com o auxílio da máquina física, por entendermos que é nela que enunciamos, que mantemos relação com a escrita, ou seja, iremos nos deter na questão do *software*.

Ao retomarmos o estado da arte dos estudos em linguagem e tecnologia, (KOCH, 2003; POSSENTI, 2002; RIBEIRO, 2003, 2005; COSCARELLI, 2003, 2005; XAVIER, 2002, 2007; ARAÚJO, 2004; KOMESU, 2005 entre outros<sup>6</sup>) verificamos que os estudos iniciados quando do aparecimento, na sociedade humana, do computador conectado à rede Internet, não faziam menção à possibilidade de um artefato tecnológico, como o *software*, ser um objeto que aporte, em si, hipertextos e gêneros.

Entretanto, o reconhecimento dessa proposição, como verdadeira, produz não apenas mudanças de perspectivas como também de ações: a forma de tratamento dos enunciados, como, por exemplo, os hipertextos<sup>7</sup>, e todas as ações sociais que o uso de um *software* pode desencadear. No entanto, mesmo que haja indícios que possam sustentar tal afirmação, pouco ou quase nada foi produzido a partir dela no âmbito das ciências da linguagem.

Afirmar que o *software* aporta, em si mesmo, gêneros, hipertextos, etc, e que é no *software* que mantemos relação direta com a escrita, significa dizer que esse produto tecnológico é formado por uma série de inter-relações observáveis desde o

---

<sup>6</sup> Esses pesquisadores representam o início dos estudos em linguagem e tecnologia na linguística brasileira. Nessa época, era o que acontecia da tela para fora que os interessavam e não o contrário.

<sup>7</sup> Sobre o hipertexto como enunciado e a hipertextualidade como enunciação, lembraria aqui os recentes trabalhos de Lobo-Sousa (2009) e Lobo-Sousa; Araújo e Pinheiro (2009).

seu estado embrionário, quando um sujeito historicamente situado<sup>8</sup> o projeta, até as relações desencadeadas quando outro sujeito social começa a utilizar esse *software*. Assim, observa-se que os modos enunciativos, em ambiente digital, os gêneros, os hipertextos, as hipermídias, são partes dele, do *software*, e estão nele materializados<sup>9</sup>.

Quanto à escrita, ela é utilizada em contextos sociais básicos da vida cotidiana, na escola, no trabalho, na família etc. Em cada um desses contextos (cf. MARCUSCHI, 2005), o uso e os objetivos da escrita são variados e fazem surgir gêneros. A história mostra que desde a escrita cuneiforme à escrita egípcia hieroglífica, dos manuscritos ao surgimento da imprensa com Gutemberg, de Gutemberg à sociedade digital, o ser humano sempre interagiu com textos em suportes.

Em qualquer um dos contextos sociais, são inevitáveis as relações do gênero com o suporte. Uma vez que os suportes são condições de possibilidade de se tornar os gêneros materializados, de segmentá-los, de inseri-los em novos contextos, de permitir-nos manipulá-los e de engendramos novas formas de contato com eles (cf. ARAÚJO, 2008). No evoluir da história material do suporte, este sempre condicionou as possibilidades e as liberdades da inscrição da escrita. Se um gênero é resultado de determinadas práticas sociais, a evolução (mudança) de práticas sociais implica necessariamente uma transformação nos gêneros e o surgimento de outros.

Na medida em que fenômenos novos vão surgindo, como o aparecimento dos gêneros digitais, parece-nos natural que os cientistas da linguagem busquem em outra área da atividade humana, pontos análogos para compreender o fenômeno novo que se apresenta. É assim que buscam estudar os gêneros digitais, os hipertextos, fazendo sempre comparações, ou tendo, como correspondência, algo semelhante e que faça parte de um mundo já conhecido: o mundo impresso.

---

<sup>8</sup> Indivíduo inserido em um ambiente e em dado período histórico, isto é, em um contexto que determina certas características, inclusive, de seu modo de pensar e agir diante das mais diversas circunstâncias do cotidiano.

<sup>9</sup> Matéria como aquilo que é capaz de sofrer mudança substancial.

Foi assim que os primeiros estudos sobre o gênero *chat* o viram como resultado da transmutação da conversa oral cotidiana para uma conversa teclada (ARAÚJO, 2003; 2004; 2006); do mesmo modo, pesquisas acerca do e-mail encontravam, nesse gênero digital, similitude com um gênero impresso chamado carta (JONSSON, 1997; PAIVA, 2004; MARCUSCHI, 2004). Nesse sentido, são memoráveis as palavras de Paiva (2004 *on-line*) quando assegura que o e-mail

é um gênero eletrônico escrito, com características típicas de memorando, bilhete, carta, conversa face a face e telefônica, cuja representação adquire ora a forma de monólogo ora de diálogo e que se distingue de outros tipos de mensagens devido a características bastante peculiares de seu meio de transmissão, em especial a velocidade e a assincronia na comunicação entre usuários de computadores

Refletindo sobre o e-mail, Marcuschi (2004) lembra que estes são correspondências pessoais que possuem um formato textual muito próximo ao de uma carta, em outros casos o e-mail tem uma estrutura típica de um bilhete. Assim, prossegue o autor (2004, p.41):

o e-mail é um gênero que, como as cartas, tem respostas (mas não sempre). Na falta de resposta pode-se supor que o destinatário não recebeu ou não quer responder, ou recebeu e não respondeu. Mas há casos em que o endereço do remetente não funciona na recepção de respostas. O fluxo é determinado não só por decisões pessoais, mas também por condições tecnológicas. Uma das vantagens dos *e-mails* é sua transmissão instantânea encurtando o tempo de recebimento.

Foi dessa forma que os primeiros estudos vislumbravam o blog mais como um espaço análogo a um diário íntimo (PRANGE, 2002; SIBILA, 2003; SCHITTINE, 2004; KOMESU, 2004; 2005). Ou ainda o fato de dizerem ser todo texto um hipertexto (KOCH, 2003; COSCARELLI, 2005; RIBEIRO, 2008). O raciocínio ora exposto denota um modo de fazer ciência por analogia, a qual pressupõe a

inferência de que num conjunto não muito extenso de objetos, se estes estão em concordância sob vários aspectos, podem muito provavelmente estar em concordância também sob um outro aspecto. Por exemplo, a Terra e Marte estão em concordância sob tantos aspectos que não parece improvável que possam concordar também quanto ao fato de serem habitados. (PEIRCE, 2003 p.06)

Um dos pensamentos análogos, no Brasil – quiçá no mundo – que têm influenciado o modo de pensar a escrita eletrônica é o do historiador francês Roger Chartier. Interessa-nos rever as ideias de Chartier (1994a; 1994b; 2002; 2004) uma

vez que são delas que se valem os cientistas da linguagem<sup>10</sup> para balizar suas falas no tocante ao lugar da inscrição da escrita, tanto em meio impresso quanto digital, e todas as inter-relações desencadeadas por esse produto que serve para inscrição da escrita.

De um lado, o que temos em Chartier (1994a; 2002; 2004) é um axioma, ou seja, a reprodução de um raciocínio baseado em analogias – no sentido apontado por Peirce (2003) –, o qual conduz a uma premissa admitida como universalmente verdadeira, sem exigência de demonstração. E, de outro, o que observamos é uma tentativa de encontrar um método que balize as ideias sobre suportes, sobre hipertextos<sup>11</sup>, questões essas que são importantes para a atual Ciência Linguística.

Em Chartier (1994; 2002; 2004), temos uma retomada histórica acerca do processo de leitura e escrita, a qual assinala que essas práticas – tanto na era do manuscrito quanto do impresso –, desenvolveram-se a partir do momento em que os objetos, que serviam de inscrição para escrita, e as técnicas de reprodução, evoluíram. Essa evolução, que ele chama de suporte<sup>12</sup>, na técnica e no nível da prática de leitura, configura-se como uma revolução tecnológica que transforma radicalmente o modo como um texto chega ao leitor.

Com a revolução tecnológica, todos os objetos que permitiam a inscrição da escrita e que davam forma à cultura impressa, estão agora em um único suporte, a tela do computador. Assim, o livro, a revista, o jornal, transpostos para era digital, possuem como suporte a tela do computador, a qual permite ao leitor uma leitura descontínua e uma hipertextualidade marcada por múltiplos vínculos<sup>13</sup> (CHARTIER

---

<sup>10</sup> Pelo menos nas produções brasileiras a fala do historiador, no tocante a importância do lugar da inscrição da escrita (suporte) para recepção do texto, do gênero que nele é fixado, é bastante difundida.

<sup>11</sup> Chartier chama de textos eletrônicos.

<sup>12</sup> Outros autores também tratam da questão do suporte no ambiente impresso. É o caso, por exemplo, de Bezerra (2006; 2007) que assegura que suportes participam da construção do sentido das mensagens escritas; de Távora (2008) que, em tese doutoral, esboçou uma proposta para o conceito de suporte de gêneros textuais de modo a torná-lo operacional para analisar a relação suporte/gênero. Também de Maingueneau (2002) que, através de uma perspectiva discursiva, considera os suportes como força material dos discursos, objetos que garantem práticas discursivas únicas, ou ainda como Marcuschi (2003), de uma perspectiva textual, argumenta que o suporte tem a função de fixar e mostrar o gênero.

<sup>13</sup> O que deve ser combatido, já que a não-linearidade é algo constitutivo da cognição humana e não dos suportes de escrita.

1994; 2002; 2004). O computador “faz surgir diante do leitor os diversos tipos de textos tradicionalmente distribuídos entre objetos diferentes. Todos os textos, sejam eles de qualquer gênero, são lidos em um mesmo suporte (a tela do computador) e nas mesmas formas” (CHARTIER, 2002, p.22).

É, no entanto, necessário assinalar, sem nenhum demérito ao trabalho desenvolvido por Chartier (1994; 2002; 2004), que houve um notável desvio de enfoque nos seus escritos, pois, enquanto a analogia era feita entre objetos que possuíam uma mesma natureza (o manuscrito e o impresso), ela respondia bem às perguntas do problema. Quando, contudo, ele passou a buscar pontos semelhantes entre dois elementos disformes (impresso e digital), o resultado do seu esforço foi infrutífero uma vez que não levou em consideração diferenças entre um ambiente e outro.

Para Chartier (1994a, p.187),

a revolução do nosso presente é, com toda certeza, mais que a de Gutenberg. Ela não modifica apenas a técnica de reprodução do texto, mas também as próprias estruturas e formas do suporte que o comunica a seus leitores. O livro impresso tem sido, até hoje, o herdeiro do manuscrito: quanto à organização em cadernos, à hierarquia dos formatos, do *libro da banco* ao *libellus*; quanto, também, aos subsídios à leitura: concordâncias, índices, sumários etc. Com o monitor, que vem substituir o códice, a mudança é mais radical, posto que são os modos de organização, de estruturação, de consulta do suporte do escrito que se acham modificados.

Como pode-se observar, da citação acima, o autor, deixa de constatar que os objetos da cultura impressa – o livro, o jornal, a revista, etc –, não foram substituídos pelo monitor do computador. A tela do computador é um objeto que nos permite apenas enxergar o que está dentro dele. Já em relação àquilo que está materializado no *software*, *a tela nem pode dar forma ao escrito como também não possui a capacidade de armazenar o escrito*<sup>14</sup>. *Ela apenas nos permite enxergar o que está em um software, se ela não existisse outro dispositivo poderia servir para isso, como por exemplo, um data show.*

---

<sup>14</sup> Conforme já apontou Távora (2008)

## 1.1 A tela como suporte da escrita

Alguns estudiosos, como Xavier (2002); Távora (2008) e Soares (2002)<sup>15</sup>, provavelmente, inspirados nos estudos de Chartier (1994; 2002; 2004), atribuem, à tela do computador, o rótulo de suporte da escrita. Nesse sentido, é provável que a premissa da tela, enquanto suporte da escrita, do hipertexto, assumida pelos autores anteriormente citados deva-se ao fato de desconsiderarem que o computador seja uma máquina composta por dois elementos: o *hardware* e o *software*. E, além disso, por tratarem o *software* apenas como uma linguagem de programação.

É importante ressaltar também que tais estudos, ainda que partindo de uma premissa, a nosso ver, equivocada, são importantes, na medida em que lançam luzes sobre fenômenos ainda pouco explorados. Entretanto, a proposição de que a tela é o suporte da escrita<sup>16</sup>, traz alguns problemas para os estudos enunciativos, em ambiente digital.

Tais problemas, como veremos, repercutirão nos estudos dos gêneros digitais, como, por exemplo, da hipermodalidade, da hipertextualidade e do hipertexto<sup>17</sup>. Essa orientação parece-nos, pois, inadequada, tanto do ponto de vista linguístico – como é possível um único *suporte* abranger a infinidade de gêneros que circulam, por exemplo, na Internet? –, quanto do ponto de vista técnico-informático. Pois, se percorrermos brevemente a origem e a evolução dos computadores, veremos que, nos primórdios da computação moderna, a programação dos computadores era feita através de inúmeros cartões perfurados. Esses cartões de papel, inseridos na máquina, funcionavam como dispositivos de entrada, como dispositivo de inserção de dados e comandos no computador.

A saída das informações, processadas pelo computador, era feita por impressoras conectadas a ele. Com a evolução do método de programar, o

---

<sup>15</sup> Soares (2002) não é propriamente uma pesquisadora da área de Linguística e sim da Educação. Entretanto, seus trabalhos sobre letramento na cyber cultura têm cada vez mais espaço nos estudos em linguística aplicada sobretudo em trabalhos que tratam a questão do letramento digital.

<sup>16</sup> Chamaremos essa linha de raciocínio de proposição da tela.

<sup>17</sup> A maior prova disso é o fato de que os primeiros trabalhos na área – em sua maioria – não diferenciam hipertexto de hipertextualidade nem separam hipertexto de hipermídia, nem consideram os elementos sógnicos presentes em uma interface. Nesse sentido, é oportuno lembrar que trabalhos recentes já fazem essa distinção, como é o caso de Lobo-Sousa (2009).

dispositivo de saída (a impressora) precisou ser substituído por outro dispositivo (a tela) que permitisse que o trabalho de programar fosse feito com maior agilidade. Para isso, nos idos de 1970, os cientistas da computação conectaram um aparelho de TV ao computador e deram a esse conjunto o nome de terminal de computador. Desse modo, era possível digitar um comando diretamente no computador e receber deste uma resposta imediata.

Com o advento da tela, os programas passaram a ser inseridos diretamente na máquina, em uma linguagem de programação. Essas instruções enviadas ao processador do computador, em princípio, se constituía em um trabalho penoso, pelo fato de que os programadores necessitavam escrever os comandos diretamente em linguagem de máquina. Com o evoluir da ciência da computação, programas foram criados – os compiladores – para facilitar a atividade de programar. Esses compiladores permitiam que os programadores pudessem criar *softwares* com maior facilidade.

O que antes era feito com o auxílio de cartões perfurados e uma impressora, passou a ser feito com o auxílio de programas digitais e visto em um monitor. O salto quantiquantitativo, no modo de como programar o computador, possibilitou a criação de *softwares* com interfaces homem-máquina cada vez mais amigáveis, ou seja, permitiu a criação de interfaces em que o humano pudesse interagir com máquinas não só para programá-las, mas, sobretudo, para executar atividades (as mais variadas) de forma simples e intuitiva. A atividade de programar, com a inserção da tela, transformou-se substancialmente. Antes, o que demorava semanas para ser feito, uma vez que era necessário escrever o programa em inúmeros cartões perfurados para só depois inseri-los no computador, passou a levar dias, horas.

Por não atentarem (CHARTIER, 1994; XAVIER, 2002; SOARES, 2002; TÁVORA, 2008; entre muitos) para o fato de que a tela é apenas um dispositivo de saída de informações, como a impressora, ou o *data show*, que os trabalhos que fazem referência ao lugar da inscrição da escrita, em ambiente digital, creditam, a ela, o *status* de suporte<sup>18</sup>. É por adotar tal premissa, que, por exemplo, Xavier (2002,

---

<sup>18</sup> Entendemos a tela do computador como um *locus* de fixação ou como um lugar de atualização de informações.

p. 9), postula a tela do computador como lugar da inscrição do hipertexto: “o hipertexto é um *modo de enunciação digital*, resultado do amálgama, integração e superposição dos vários modos de enunciação (verbal + visual + sonoro), em um mesmo suporte digital de leitura: a tela do computador”. Para Souza (2007b, p.3), se olharmos

[...] a premissa da tela como suporte, de um ponto de vista técnico-informático, de imediato percebe-se a invalidade dessa tese, vejamos, o computador pode ser dividido em partes: HD, memória, placa mãe, processador e placa de vídeo. O ‘processador’ é considerado o cérebro do computador e é definido como um circuito integrado de controle das funções.

Os programas e os dados que ficam armazenados no HD (disco rígido), são transferidos para a memória RAM da CPU. Uma vez estando na memória, o processador pode executar os programas e processar os dados. Na memória RAM esses programas que já foram processados no processador do computador são transferidos para a placa de vídeo que envia as informações para a tela. A tela possui um dispositivo, que recebe sinais elétrico, chamado de ‘canhão de elétrons’. As intensidades desses sinais são transmitidas e decodificadas pelo ‘controlador de vídeo’ que comanda a voltagem do canhão de elétrons e sincroniza o sistema de placas defletoras (horizontal e vertical) fazendo aparecer assim às imagens. Se a tela não existisse, essas informações poderiam ser mostradas em uma parede, por exemplo, ou até mesmo no corpo de alguém bastava para isso que outro dispositivo de saída fosse acoplado ao computador, por exemplo, o data show.

Adotando a perspectiva da tela como suporte, seria um contra-senso estudar os gêneros digitais, por exemplo, a partir da idéia bakhtiniana de estilo, uma vez que o estilo está diretamente ligado ao suporte em que o enunciado escrito se manifesta, é o suporte que nos permite manter contato direto com a escrita, que nos permite fazer uso da escrita.

Aspecto ainda a ser considerado na perspectiva empreendida por Chartier (1994) e seus seguidores, diz respeito a uma homogeneização do suporte, o que, em tese, traria homogeneidade aos gêneros suportados. A observação de Chartier, com relação ao suporte da escrita em ambiente digital, embora empiricamente possa parecer correta, não oferece embasamento teórico suficiente para se analisar, por exemplo, um gênero digital.

Reconhecemos a relevância dos estudos sobre a origem e a evolução das tecnologias da escrita, desenvolvidos por Chartier (1994; 2004); entretanto, acreditamos que a analogia criada, para explicar o processo da leitura e da escrita,

em ambiente digital, analogia aceita como verdadeira por quase todos que se dedicam ao tema, falhou ao esquecer as diferenças relevantes (hipertextualidade, hipermodalidade, interatividade, etc) entre o ambiente digital e o impresso. Falhou ao desconsiderar o *software* como um objeto capaz de dar forma ao escrito. Falhou ao não atentar para o fato de que *lemos através da tela o que um suporte nos possibilita e não que lemos na tela. Falhou ao não considerar que a tela só atualiza e, portanto, só apresenta o material enunciativo*<sup>19</sup>.

## 1.2 O *software* como lugar da inscrição da escrita

A posição que defendemos nesse trabalho é a de que o *software* é o lugar da inscrição da escrita. Ele não é só um conjunto de instruções isoladas, enviadas ao processador do computador para que este desempenhe determinada tarefa. Numa proposição mais simplista, poderíamos dizer que olhar o *software* como conjunto de instruções isoladas (algoritmos) equivale a levar em consideração o *processo de fabricação do software e não o produto software*.

A distinção entre o *processo* de fabricação do *software* e o *produto software* (no qual mantemos contato, no qual enunciamos, nos comunicamos com outros sujeitos situados, etc.), faz-se necessária na medida em que passamos a considerar o *software* como suporte da escrita<sup>20</sup>, dos gêneros digitais (SOUZA, 2007b).

<i>Software</i> = produto ≠ processo = linguagem de programação
---

<sup>19</sup> 'a tela só atualiza e, portanto, só apresenta o material enunciativo' é uma premissa abduzida por nós de Távora (2008). Embora a utilizemos aqui para dizer o oposto do que o autor diz, ou seja, que o *software* é o suporte da escrita.

<sup>20</sup> Entendemos com Marcuschi (2005, p.26) a escrita como um modo de produção textual-discursiva para fins comunicativos com certas especificidades materiais e se caracterizando por sua constituição gráfica, podendo envolver também recursos de ordem pictórica e outros. Quanto a sua manifestação na sociedade, ela pode estar presente desde o ponto de vista de sua tecnologia, por unidades alfabéticas, ideográfica ou unidades iconográficas, sendo que no geral, não temos uma dessas escritas puras. A palavra escrita aqui utilizada diz respeito a todos os tipos de escrita, sejam eles alfabéticos ou ideográficos, entre outros.

Assim, como vimos estudando (SOUZA, 2006; 2007; 2009a; 2009b), passamos então a considerar os *softwares* como objetos que aportam, em si, enunciados (hipertextos), textos e gêneros digitais que *vemos através da tela de um computador*. Além disso, muito do que o usuário faz com um gênero digital é atribuído ao *software*, ou seja, muito da hipertextualidade – entendida aqui como propriedade do hipertexto – nasce do e no *software*. *Não existe hipertexto eletrônico sem que ele esteja materializado em um software*. Esses hipertextos, esses enunciados inscritos no *software*, os gêneros digitais, poderão se mesclar com uma infinidade de outras mídias, constituindo assim organismos hipermodais<sup>21</sup>.

O *software* é um produto cada vez mais presente na sociedade atual: “além de ser uma entidade de natureza mecânica, é também uma entidade descritiva, complexamente hierarquizada, cognitivo-linguística e historicamente concebida através de esforços coletivos durante um considerável período de tempo” (FERNANDES, 2003, p. 29). Assim, passamos a estudar o *software* como lugar de inscrição da escrita, um artefato mediador da linguagem e mediado pela linguagem, levando sempre em consideração os processos sequenciais que engendram um *software*, ou seja: a interface, as funcionalidades do *software*, a base hipertextual que lhe é própria, bem como o corpo de signos que o constitui.

Admitir, por conseguinte, que os textos, hipertextos, gêneros, sons e vídeos estão materializados em *softwares*, significa assumir, como primeira tarefa, uma discussão das propriedades desse objeto. Admitamos, portanto, por empirismo e também por categorização, que os *softwares* são entidades que se diferem das outras que servem para inscrição da escrita<sup>22</sup>, diferenciando-se, por exemplo, dos papiros quando estes ainda serviam de suporte para a inscrição da escrita, do papel, etc.

Assim, ao considerarmos o *software* como lugar onde os hipertextos e gêneros se inscrevem, é útil termos presente a ideia de que um *software* é concebido para dialogar (no sentido lato da palavra) com o usuário (PERES; MEIRA,

---

<sup>21</sup> Discutiremos melhor essa questão na parte 3 deste trabalho

<sup>22</sup> Não é preciso fazer um grande levantamento para perceber que o *software* se difere de qualquer outro suporte, basta atentar para sua natureza, digital, e para a natureza dos outros objetos que servem de inscrição para escrita.

2006). Esse é um ponto de grande relevância o qual nos conduz a repensar, de imediato, tanto a relação que o *software* mantém com o gênero como a relação que o usuário mantém com o *software* e com o gênero, relação essa fundada no uso.

### 1.3 O Círculo bakhtiniano e a Ergolinguística

É na confluência das ideias do Círculo bakhtiniano que a teorização sobre *software* e interface aqui tratada têm seu ponto de apoio, uma vez que partimos do pressuposto teórico consensual de que a linguagem é também uma forma de atividade. As teorias sobre a atividade são, por constituição, multidisciplinares e, na linguística, tem seu ponto mais forte na *Teoria Dialógica da Linguagem* de Bakhtin (1997<sup>23</sup>). Para o teórico russo e o seu Círculo (formado por V. N. Volochinov e P. N. Medvedev), dialogismo é o modo de funcionamento real da linguagem e é também seu princípio constitutivo.

No Círculo, a linguagem é pensada como produto da vida social e a noção de linguagem passa, necessariamente, pela relação eu/outro, relação de alteridade. No entanto, o eu e o outro são universos de valores distintos. O eu só existe em interação com o outro. As relações eu/outro, mergulhadas na noção de dialogismo, são desenvolvidas mais explicitamente quando da questão dos enunciados concretos. O enunciado, para o Círculo, é tomado não como uma forma isolada da língua, mas como unidade real da comunicação discursiva. É delimitado através da alternância dos sujeitos falantes. Numa concepção bakhtiniana, é impossível pensar a língua/linguagem sem levar em consideração as relações dialógicas como um fenômeno concreto.

O dialógico em Bakhtin (1997) vai definir um traço constitutivo da linguagem, que toma forma através de enunciados. O discurso, o enunciado, é sempre dirigido a alguém que é capaz de compreendê-lo e dar-lhe uma resposta. O enunciado tem uma orientação social. E é a partir dessas relações, dessa orientação social, que

---

<sup>23</sup> Utilizamos aqui a tradução de: *Estetika slovesnogo tvorchestva*, 1979, feita pela Livraria Marfins Fontes Editora.

atua como uma força viva e constitutiva, que a forma estilística do enunciado e a estrutura estritamente gramatical se organizam. Além da orientação social, dialógica, todo enunciado comporta um sentido, um conteúdo.

É na ideologia do cotidiano, no diálogo cotidiano, na interação social ou aquilo que Bakhtin (1996) denomina de colóquio ideológico em grande escala, que nascem as relações dialógicas. Assim, como aponta Bakhtin (1997), às relações dialógicas não podem ser reduzidas a relações de ordem lógica, linguística (no sentido estrito do termo), psicológica, mecânica ou natural. São relações de sentido que se estabelecem entre enunciados ou mesmo no interior dos enunciados. Para haver relações dialógicas é preciso que qualquer material linguístico tenha entrado na esfera do discurso. Tenha sido transformado em um enunciado, tenha fixado a posição de um sujeito real.

Relações dialógicas<sup>24</sup> são possíveis não só entre enunciados completos (relativamente completos); uma abordagem dialógica é possível em relação a qualquer parte significativa de um enunciado, mesmo em relação a uma só palavra, caso aquela palavra seja percebida não como uma palavra impessoal da língua, mas como um signo da posição semântica de um outro alguém, como representante do enunciado de outra pessoa, isto é, se ouvirmos nela a voz de outro alguém. Assim, relações dialógicas podem permear o interior do enunciado, mesmo o interior de uma só palavra, desde que nela duas vozes colidam dialogicamente. Bakhtin (1981)

As relações dialógicas nascem então das relações de sentido que se estabelecem entre dois enunciados. O enunciado possui certas particularidades, como, por exemplo: é carregado de juízo de valor e é sempre destinado a alguém. É caracterizado pela alternância de sujeitos falantes. Os enunciados possuem um objeto de discurso e formas típicas de acabamento. Todo enunciado é dialógico e seu sentido também é dialógico e as relações dialógicas se dão entre pessoas que produzem enunciados.

---

<sup>24</sup> Convém não confundir relações dialógicas com diálogo no sentido estrito do termo, o diálogo face-a-face. Como observa Faraco (2003, p.60) o Círculo olha para o diálogo face-a-face como evento da grande interação sociocultural de qualquer grupo humano, como espaço de vida da consciência socioideológica; como eventos atravessados pelas mesmas forças dialógicas.

As noções de enunciado/enunciação têm papel central na concepção de linguagem que rege o pensamento bakhtiniano, justamente por que a linguagem é concebida de um ponto de vista histórico, cultural e social. Enunciado/enunciação se vinculam com signo, ideologia, palavra, comunicação, interação, gêneros discursivos, texto, tema e significação, discurso, discurso verbal, polifonia, dialogismo, ato/atividade/evento e demais elementos constitutivos do processo enunciativo discursivo.

Bakhtin (1997) e sua teoria dialógica leva-nos a refletir sobre o papel do outro na produção de um artefato informatizado e como esse outro, ou melhor, as enunciações desses outros interferem no acabamento do produto bem como no uso desse produto *software*.

Tomaremos também como fundamentação teórica para explicar o trabalho e o retrabalho numa fábrica de *software*, embasamento nos postulados da Ergolinguística, uma disciplina pluridisciplinar que, de acordo com Sampaio et al (2006a, [on-line]),

se ocupa de temáticas originadas nas relações linguagem e trabalho na vida social, e é forjada na confluência da ciência linguística com a ergologia e a ergonomia, aproximando esta nova forma de fazer pesquisa linguística com outras disciplinas com as quais dialoga (história, filosofia, sociologia, psicologia, gerontologia, etc).

A ergonomia foi durante muito tempo a disciplina que deu base aos estudos sobre o trabalho que, conforme Faïta (2002, p. 45-46), “até então só levava em consideração o trabalho prescrito e o trabalho real”. Com o avançar dos estudos sobre o trabalho, a ergonomia passa a dialogar com outras disciplinas, como por exemplo, a ergologia. Este diálogo instaurado permitiu uma revisão acerca da díade ‘trabalho prescrito e trabalho real’. Assim, os estudos posteriores começaram a levar em consideração o trabalho não como uma repetição mecânica, mas como espaços de encontro de várias subjetividades.

Com a ergologia surge o conceito de atividade, esta vista como arena de debates entre “normas heterodeterminadas que antecedem seu andamento e as experiências recriadoras dos sujeitos, as renormatizações que se processam em cada gesto e em cada situação” (FRANÇA, 2004, p.124). Assim, para uma análise

do trabalho é levado em consideração os usos de si que o trabalhador faz, (cf. SCHWARTZ, 2000). Desse modo, a díade ‘trabalho prescrito e trabalho real’ cedeu lugar aos usos de si pelos outros e os usos de si por si mesmo.

O trabalho em uma fábrica de *software* que aqui será explicitado<sup>25</sup> parte da compreensão de que as relações entre linguagem e trabalho não se resumem apenas as ações linguageiras 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>26</sup>. Esses “três aspectos da linguagem estão em estreita ligação com práticas linguageiras na situação de trabalho considerada de maneira global” (NOUROUNDINE, 2002, p. 18).

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<sup>27</sup>, 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*.

Como caminho para análise<sup>28</sup> do objeto aqui em questão, optamos por revisitar a origem dos primeiros computadores bem como a origem e a evolução dos *softwares* sistemas operacionais com suas respectivas interfaces assim como os *softwares* aplicativos. Embora sabendo da interdependência dos programas, separamos *software* sistema operacional de *software* aplicativo a fim de nos determos apenas no segundo. Isso não significa que houve um esquecimento do primeiro. Ao contrário, isso foi feito para que pudéssemos guiar nossa atenção para

---

<sup>25</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>26</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>27</sup> Quando falamos que a escrita está materializada em *softwares*, estamos presumindo que o *software* possua uma forma material, entretanto, quando falamos em matéria, estamos nos referindo a um conceito abduzido de Aristóteles, e esse conceito, essa ideia inicial, em nada se assemelha ao conceito de matéria utilizado, por exemplo, pela física. Para Aristóteles, matéria é aquilo – seja o que for – de que é feito algo.

<sup>28</sup> A decomposição do todo em suas partes

os *softwares* em que a maioria dos gêneros digitais estão imbricados (os *software* aplicativos que permitem navegação na web).

A partir deste ponto, partimos para verificar o modo como um *software* é fabricado, para só então decompô-lo em micro-unidades de análise, a saber: interface, enunciações de desenvolvedores, signos e funcionalidades. Com isso desviamos nosso olhar do código de programação pra observar e descrever o modo de funcionamento do produto *software*.

#### 1.4 Constituição e descrição dos *softwares*

O *corpus* de análise será constituído por dois *softwares* aplicativos que possibilitam a navegação na rede Internet, denominados de 'Firefox' e 'Internet Explorer'. A escolha justifica-se pelo fato de que estes *softwares* são os mais utilizados no mundo<sup>29</sup>: o primeiro detém 20,78% do mercado de *browsers* e o segundo 69,77%. Ambos foram coletados nos sites das empresas que os fabricam.

Esses *web browsers*, distribuídos livremente por seus fabricantes (Mozilla Foundation e Microsoft), estabelecem ligações entre servidores web, processam os mais variados tipos de arquivos e acoplam<sup>30</sup> a maioria dos gêneros digitais que vemos na Internet.

Quando a rede Internet começou a ser operada comercialmente, existia apenas um *software* que permitia o acesso à WWW: o Netscape Navigator<sup>31</sup>. O monopólio do *browser* durou até o final da década de 90, quando houve o lançamento de outro *software* de navegação batizado de Internet Explorer e que já vinha instalado em um sistema operacional chamado de Windows, fabricado pela

---

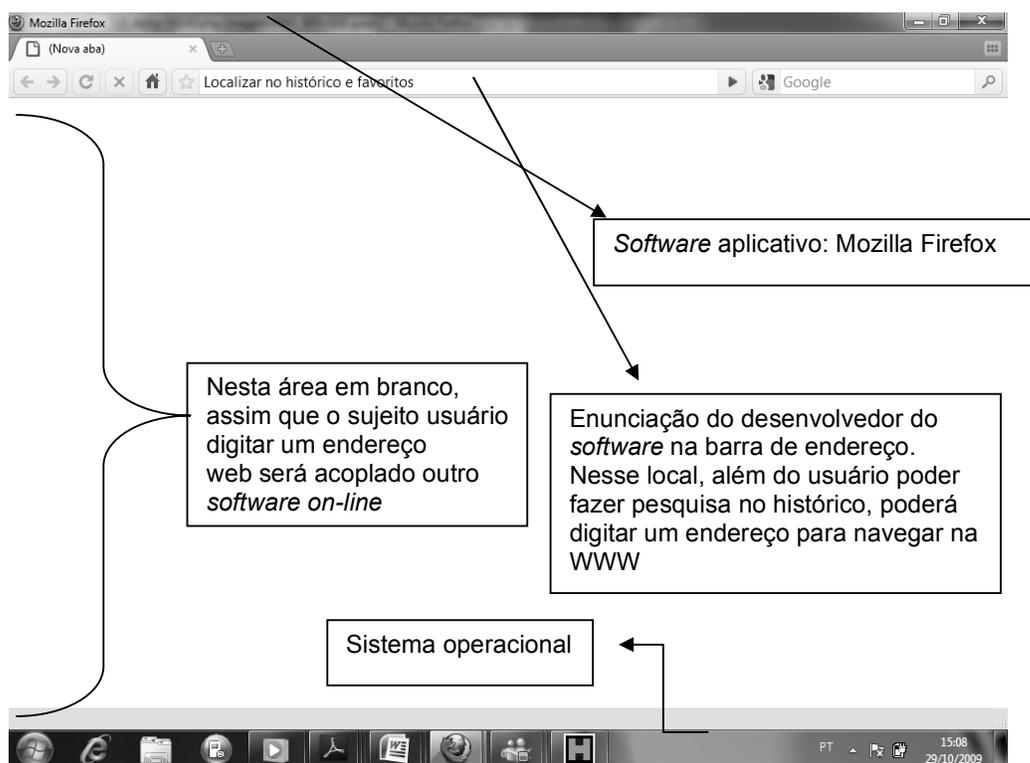
<sup>29</sup> Esses são os *softwares* mais utilizados para navegar na rede internet, segundo pesquisa divulgada pela Net Application em 01 de dezembro de 2008, disponível para consulta em: <http://www.netapplications.com/newsarticle.aspx?nid=45>

<sup>30</sup> Quando dizemos que um *software* acopla outro estamos dizendo que ele pode se juntar, se ligar fisicamente a outros corpos ou objetos, formando um único corpo e passando a funcionar de forma integrada, complementar ou coordenadamente com outros *softwares* ou objetos que a eles se liguem. A expressão 'acoplar' utilizada por nós em nada se assemelha a utilizada na ciência da computação.

<sup>31</sup> O Netscape utiliza boa parte do seu código do Mosaic, este considerado o primeiro navegador web. Foi com o Mosaic que o mercado de *browsers* ganhou vida. Ele foi o primeiro a possuir uma enorme capacidade gráfica a qual permitia a distribuição e a captação de imagens pela internet. Além disso, a esse *software* foram incorporados conceitos da multimídia o que possibilitava também a captação de som e vídeo na grande rede. O Netscape é a evolução do Mosaic.

Microsoft. A entrada do Internet Explorer no mercado de *browsers* desencadeou o que a história chama de guerra dos *browsers*<sup>32</sup>. Esse fato fez com que a Netscape sucumbisse, passando a criar uma fundação chamada de Mozilla<sup>33</sup>, a qual ofereceu aos programadores do mundo todo, o código fonte do seu produto, o Netscape Navigator.

Um dos aspectos que permitiram o sucesso do Mozilla Firefox (ver figura 1) foi o fato deste ser um *software* fabricado em comunidade. Hoje a fundação Mozilla funciona aos moldes da Wikipédia, uma rede de colaboradores espalhados pelo mundo trabalha de forma voluntária para a melhoria dos produtos oferecidos por ela.



**Figura 1 Mozilla Firefox customizado**

<sup>32</sup> Diz respeito ao período de 1995 a 1999 quando a Netscape perdeu a liderança no mercado de navegadores para a Microsoft. A Microsoft foi processada por *truste* uma vez que incorporava ao seu sistema operacional líder de mercado (Windows) um navegador. Para mais informações a respeito ver Lewis (2000)

<sup>33</sup> Fundação Mozilla é nome dado a esse empreendimento sem fins lucrativos que tem como missão dar apoio ao desenvolvimento de *software* livre

A arquitetura do web *browser* Firefox tem por base o uso de extensões, o que na prática permite que um usuário leigo possa customizar o *software*, instalando *plugins*. Além disso, o programa possui alta compatibilidade com os padrões da web como, por exemplo, compatibilidade com DOM<sup>34</sup>, com HTML, SHTML, AJAX<sup>35</sup>, JavaScript<sup>36</sup> entre muitos. Por ser um *software* de código aberto, qualquer um pode modificar seu código fonte, inserindo com isso novas funções ao *software*. Além do mais, o *browser* é projetado para se adaptar a multiplataformas, ou seja, pode ser instalado em qualquer sistema operacional (Windows, Linux, Mac OS X ou Chrome Os).

O nosso segundo *software* aplicativo para análise é o Internet Explorer. Esse *software* de navegação na rede Internet produzido pela Microsoft (ver figura 2) foi incorporado gratuitamente ao sistema operacional Windows no ano de 1995. Ele possui licença proprietária, ou seja, o código de fabricação é fechado e com isso o usuário não pode modificar o código ou simplesmente ver o código<sup>37</sup>. Por alguns anos o *software* de navegação dava suporte a outras plataformas como o Macintosh, o Solaris etc.

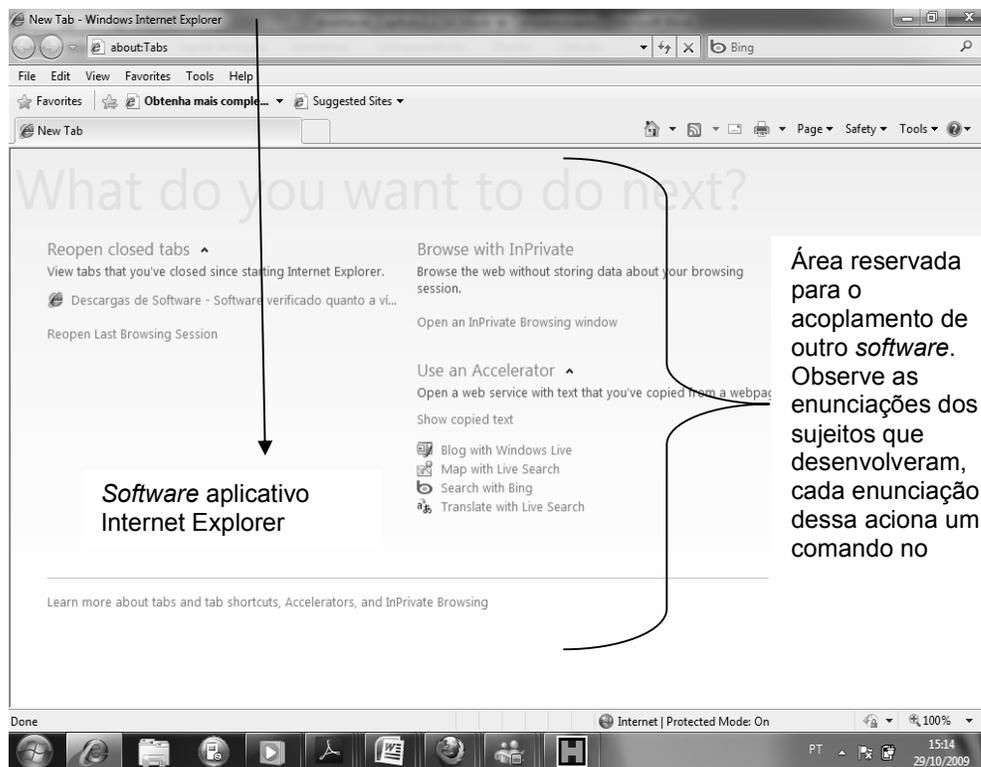
---

<sup>34</sup> DOM (Document Object Model - Modelo de Objetos de Documentos)

<sup>35</sup> AJAX (acrônimo em língua inglesa de Asynchronous Javascript And XML) é o uso metodológico de tecnologias como Javascript e XML, providas por navegadores, para tornar páginas Web mais interativas com o usuário, utilizando-se de solicitações assíncronas de informações. Foi inicialmente desenvolvida pelo estudioso Jessé James Garret e mais tarde por diversas associações. Fonte: [http://pt.wikipedia.org/wiki/AJAX\\_\(programação\)](http://pt.wikipedia.org/wiki/AJAX_(programação))

<sup>36</sup> Linguagem de programação criada pela Netscape em 1995 e utilizada na maioria dos *softwares* aplicativos para Internet

<sup>37</sup> Embora seja esse um fator determinante para diferenciar os dois *softwares*, nesse trabalho não iremos fazer distinção entre *software* de código fechado e aberto.



**Figura 2 Internet Explorer**

Hoje em dia, ele roda apenas em dois sistemas operacionais: Windows e Linux. Uma das críticas a esse *software* é o fato dele ser apontado como portador de inúmeras falhas de segurança, as quais possibilitam, por exemplo, o ataque por vírus ou por terceiros ao computador que esteja navegando na rede com ele.

De posse desse *corpus* faremos uma abordagem qualitativa dos dados, com base na Teoria Dialógica (BAKHTIN, 1996; 1997) e nos postulados da Ergolinguística, no âmbito da atividade. Assim, o *corpus* coletado foi descrito e interpretado tanto em relação às suas unidades menores (*software*, interface e funcionalidades do *software*) como em relação ao seu todo (conjunto), a fim de entender todas as variáveis interligadas ao processo de fabricação e uso de um *software*.

Descreveremos como o *software* e sua interface carregam, em si, formas marcadas e não marcadas da presença de outrem, e como essas marcas da presença de outrem, deixadas durante o processo de fabricação de *software*,

interfere no processo de uso de um *software*, por conseguinte de um gênero digital, de um hipertexto, bem como interpretar-se-á os efeitos de sentido produzidos.

Para tanto, serão utilizados os dados obtidos no exame do fenômeno concreto (*software* e interface). Assim iremos revisitar as concepções teóricas em torno de *software* e interface para alterar essa concepção e, em seguida, voltar a esta com outra compreensão, sem a circularidade das teorias fechadas, o que será feito com base na síntese bakhtiniana: o dialogismo. Este, por sua vez, consoante Sobral (2008, p.136), “envolve pelos menos dois elementos em interação, dado que seus momentos são tese-tese-síntese, o que supõe uma permanente atividade de síntese”.

O exame dos fenômenos concretos, ou seja, o *exame* daquilo que *vemos através de uma tela* (*software* e interface), será feito com o uso de uma abordagem intersubjetiva a qual nos ajudará a interpretar e descrever os dados concretos de maneira qualitativa. Desse modo, levamos em consideração, ao fazer tal análise, as micro-unidades que compõem um *software* (enunciações dos desenvolvedores, enunciações dos sujeitos usuários, signos e funcionalidade<sup>38</sup>).

Assim, partimos da evidência de que a escrita, para se materializar, necessita de um objeto que a fixe e a firme. A partir daí, voltamos a essa ideia inicial e estudaremos o fenômeno apresentado. Isso feito com dados que não poderiam ser descritos como coisas isoladas, mas como fenômenos que se interrelacionam, trata-se então de uma pesquisa qualitativa onde as complexidades de oposições dos dados tratados (*software*/interface) formam um único dado coerente.

Após tecermos essas considerações e tendo em vista os dois primeiros objetivos específicos, iremos iniciar o capítulo 2 deste trabalho. Nele, abordaremos a parte mecânica do *software* (o código de programação), veremos também como os primeiros sistemas operacionais surgiram. Ao atentar para a parte mecânica do *software*, percebemos que é ela a responsável por controlar a forma arquitetônica do *software*. Na ciência da computação, o estudo das interfaces está polarizado em duas direções: uma semiótica e outra cognitiva. Desse modo, faremos uma breve ressunta dessas escolas, para, após isso, e com base nos postulados de Bakhtin

---

<sup>38</sup> Nossas categorias de análises

(1997) e da Ergolinguística, olharmos para essas interfaces como organismos que possuem uma ancoragem na metalinguística.

Buscamos, dessa forma, em tal capítulo, colocar em evidência a heterogeneidade discursiva que atravessa o desenvolvimento de um artefato digital. Pretende-se ainda verificar aspectos relativos à intersubjetividade dos sujeitos envolvidos no processo de fabricação de *software* bem como se esses *softwares* carregam em si, na sua interface, a nossa presença enquanto sujeitos historicamente situados.

## **CAPÍTULO 2**

O sistema computacional e a metalinguística; Definição preliminar sobre o que é *software*; Estudo da interface do *software* e as formas marcadas e não marcadas da presença de outrem; Breve descrição do processo de fabricação de *software* a luz da Ergolinguística

## 2- Linguagens de programação, *hardware* e Sistemas operacionais<sup>39</sup>

Os primeiros computadores eram máquinas tão grandes que chegavam a ocupar salas inteiras. Inicialmente eles foram construídos para funcionar com válvulas e painéis, cabendo aos programadores enviar as instruções através de fios, chaves e cartões. Nessa fase da computação, os *softwares* sistemas operacionais ainda não existiam. Na segunda era da computação, entre os anos de 1955 a 1965, foi desenvolvido o primeiro sistema em lote (batch systems), a base para esse sistema operacional, ainda embrionário, era um programa que monitorava o enfileiramento das tarefas a serem desenvolvidas pelo computador. Foi nessa geração que houve o aparecimento do que hoje chamamos de *softwares*.

Esses programas primitivos eram escritos em cartões perfurados<sup>40</sup>, que dependiam, para serem carregados pela máquina, de um compilador (COBOL ou Fortran) e de um ser humano que inseria e trocava esses cartões. Nessa época, o desenvolvimento de sistemas operacionais em nada se assemelhava com a produção industrial que vemos hoje, poder-se-ia dizer que a produção de sistemas operacionais era praticamente artesanal. Esse problema (produção artesanal) estava diretamente ligado a arquitetura dos computadores e das diferentes linguagens de máquina utilizadas, até então, cada computador só funcionava com um sistema operacional feito diretamente para ele.

Após essa fase, os cientistas da computação começaram a pesquisar formas de fabricar sistemas operacionais que automatizassem a troca de tarefas e que fossem compatíveis com várias máquinas, não importando a arquitetura das mesmas. Um desses primeiros sistemas foi o CTSS (Compatible Time-Sharing System), desenvolvido pelo MIT (Instituto de Tecnologia de Massachusetts -EUA) em seguida vieram outros como o *Multics*, desenvolvido pelos laboratórios Bell da

---

<sup>39</sup> Um sistema operacional é um *software* que permite o funcionamento de um *hardware*, além de fazer com que os outros *softwares* aplicativos instalados no sistema operacional, funcionem. O SO comunica-se constantemente com o *hardware* e com os *softwares* aplicativos. Interessa-nos rever o modo de nascituro desses *softwares* sistemas operacionais para entendermos os usos que deles fazemos na atualidade.

<sup>40</sup> Daí a metáfora utilizada pelos programadores: 'escrever um *software*'

AT&T/General *Electric* e o sistema UNIX. Todos esses programas eram fabricados numa linguagem de montagem denominada de *assembly*.

Apesar do passo quantificativo dado na forma de construir programas, a linguagem de montagem estava longe de oferecer um ambiente adequado para programação de computadores, uma vez que as premissas utilizadas nesse tipo de instrução eram as mesmas das linguagens de máquina. Isso forçava o programador a ter que rearranjar o código do programa em função da máquina que iria receber o *software*, uma vez que as configurações dos registradores da máquina nova não conseguiam compreender o programa transportado (BROOKSHEAR, 2001).

É atribuído a Dennis Ritchie (s.d), o desenvolvimento de uma linguagem de programação de terceira geração, chamada de C. A linguagem C é estruturada, padronizada e de alto nível. Linguagens de programação de alto nível são aquelas que permitem ao programador programar na linguagem humana, não necessitando entender e escrever em linguagem de máquina. A característica básica desse tipo de linguagem é a não necessidade, por parte do programador, de conhecer o processador do computador para poder programar. Linguagens de terceira geração, como é o caso da C, utilizam-se de tradutores que transformam a linguagem escrita em instruções de máquinas. Esses programas tradutores são conhecidos como compiladores. Um programa escrito em linguagem de terceira geração, em tese, pode ser utilizado em qualquer máquina (*hardware*), bastando para isso que haja a escolha adequada do compilador para tal máquina.

Entretanto, como nos mostra Brookshear (2001, p. 203), “quando o compilador é projetado para traduzir uma linguagem, certas restrições impostas pela máquina subjacente, podem fazer com que uma “mesma” linguagem apresente diferentes características”. Dessa forma, é necessário que o programador faça alguns pequenos ajustes no programa, antes de movê-lo de uma máquina para outra. A causa deste problema, prossegue Brookshear (2001, p. 203), “reside na falta de concordância em relação à correta composição da definição de uma linguagem particular”. Para resolver o problema de portabilidade (transferência de programa entre máquinas), o Instituto Norte-americano de Padrões e a Organização

Internacional de Padrões (ISO), órgãos reguladores da área de informática, adotaram e publicaram padrões para muitas linguagens de programação.

Após resolver, quase que completamente, o problema da incompatibilidade entre máquinas distintas, através da adoção dos padrões propostos pelos órgãos supracitados, os cientistas da computação puderam construir o “primeiro” sistema operacional que se tem notícia: o Unix. O Unix<sup>41</sup> foi o primeiro sistema operacional multiusuário que funcionava em uma grande variedade de computadores (*hardwares*).

Até o desenvolvimento do Unix, os *softwares* eram fabricados para funcionar em *hardwares* enormes (chamados de *mainframes*). Estes, em geral eram alocados em grandes centros de segurança militar ou em universidades. O primeiro *hardware* para uso pessoal foi batizado de Altair. Esse computador, embora não tenha obtido sucesso entre o público leigo, foi um marco na então indústria da computação. A partir dele houve um notável interesse das empresas em desenvolver *hardwares* para o público em geral. O Altair, por exemplo, marcou o início de uma sucessão de máquinas como a Commodore PET e a Radio Shack TRS-80, as quais foram evoluindo até chegarem à arquitetura dos atuais computadores pessoais que temos hoje (CASTELLS, 2003).

O computador doméstico, com teclado e monitor, só foi vendido comercialmente nos anos de 1970. Batizado de Apple II<sup>42</sup>, esse *hardware* de uso doméstico, possuía um sistema operacional fácil de ser operado e já não se fazia necessário ter conhecimentos avançados para manusear um computador. Ele vinha com um gabinete capaz de gerar gráficos e contava com drive de disquete de 5 ¼" além de um sistema operacional que rodava em um processador de 6502 a 1 MHz. O sistema operacional do Apple II foi batizado de AppleSoft BASIC, esse sistema ainda não possuía propriamente uma interface gráfica mas era capaz de produzir gráficos em 6 cores com uma resolução de 280x192 pixels.

---

<sup>41</sup> Outro *software* sistema operacional, fabricado logo após a fabricação do Unix, foi o BSD. Este por sua vez era quase completamente baseado no Unix, com a diferença que o BSD foi projetado para funcionar em máquinas específicas, de alto desempenho.

<sup>42</sup> O Apple II é a evolução natural do Apple I computador bastante primitivo, mais se parecia com uma máquina de escrever, desenvolvido por Steve Jobs para uso pessoal.

No final da década de 1970, os empresários da Apple tomaram conhecimento de um computador, inovador, produzido pela empresa Xerox PARC. Esse computador possuía um sistema operacional com uma interface gráfica (GUI) muito mais moderna e intuitiva que a interface do Apple. A partir de então, a Apple<sup>43</sup>, passou a se empenhar em um projeto batizado de “Lisa”. Esse projeto, apesar de seu fracasso nas vendas, modificou os conceitos de computador pessoal, uma vez que introduziu no mercado de computadores domésticos três acessórios que mudariam os rumos da informática: um monitor de alta resolução, uma interface gráfica e um mouse.



**Figura 3 Apple II**

Fonte: <http://netdna.webdesignerdepot.com/uploads/2009/01/apple-ii2.jpg>

Com o fracasso do ‘Lisa’, o idealizador da Apple é obrigado então a dedicar-se a outro projeto denominado de Macintosh. O Macintosh era um computador que

---

<sup>43</sup> A Apple Computer Inc é a empresa fundada por Steve Jobs, empresário, que recebeu o mérito de ter introduzido no mercado mundial os computadores pessoais com teclado mouse e monitor

possuía 128KB de memória RAM, resolução gráfica de 512x384, drive de disquete, mouse, recursos sonoros, além de um sistema operacional batizado de MacOs.

Com a evolução dos *hardwares* e *softwares* sistemas operacionais, aconteceu também uma explosão no desenvolvimento de *softwares* aplicativos, que se propagou até a década de 1980. Nessa ocasião, surgiu o primeiro processador de textos, batizado de *WordPerfect* bem como a primeira planilha eletrônica batizada de *VisiCalc*. Conforme vemos na figura 4:



**Figura 4 - WordPerfect**

Fonte: <http://www.guidebookgallery.org/splashes/wordperfect>

Apesar de a história tributar à Apple o *status* de pioneira no mercado de computadores para uso pessoal, é necessário dizer que ela não estava sozinha nessa caminhada. Concomitantemente a todo progresso realizado pela Apple, a IBM juntamente com a Microsoft já representavam forte concorrência. Nessa luta por consumidores, a IBM e a Microsoft saíram como vencedoras. Esse fato deve-se a uma falha no projeto de um novo computador produzido pela Apple, falha essa que fez a empresa substituir milhares de peças danificadas por superaquecimento. Esse erro no projeto 'Apple III' foi o suficiente para que a IBM e a Microsoft conseguissem conquistar o mercado consumidor (LIVINGSTON, 2009)

O computador pessoal, produzido pela IBM, tinha como padrão um modelo de negócio diferente de sua concorrente: o *hardware* produzido era de padrão aberto e

possuía um sistema operacional, MS-DOS, fabricado por outra empresa, a Microsoft (Ver figura 5). O sistema DOS, da Microsoft, foi desenvolvido para processadores da família de microprocessadores que possuíam chips da Intel e seu funcionamento era muito próximo ao UNIX. Ou seja, ele era baseado em comandos de textos e, por isso mesmo, pouco intuitivo para usuários leigos. Entretanto, com o poder econômico da IBM, o PC produzido por ela e que tinha como sistema operacional, o sistema da Microsoft, logo começou a ser um dos mais vendidos do mercado. Além desta parceria inicial, a Microsoft e a IBM firmaram outra, cujo objetivo era o desenvolvimento de um sistema operacional multitarefa, chamado de OS/2. Entretanto essa parceria durou pouco e a IBM seguiu sozinha no desafio de aprimorar o sistema operacional.

```
[Saved Games] [Searches] [Tracing] [Videos]
0 File(s)      0 bytes
14 Dir(s)     38.607.437.824 bytes free

C:\Users\Aguinaldo>scandisk
'scandisk' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\Aguinaldo>label
Volume in drive C: has no label
Volume Serial Number is 0417-072D
Volume label (32 characters, ENTER for none)?
Access Denied as you do not have sufficient privileges.
You have to invoke this utility running in elevated mode.

C:\Users\Aguinaldo>copy
The syntax of the command is incorrect.

C:\Users\Aguinaldo>label
Volume in drive C: has no label
Volume Serial Number is 0417-072D
Volume label (32 characters, ENTER for none)?

C:\Users\Aguinaldo>copy
The syntax of the command is incorrect.

C:\Users\Aguinaldo>
```

Figura 5 – MS-Dos

Na história da computação, o *hardware* sempre evoluiu juntamente com o *software*. Foi assim com os primeiros computadores, até os atuais, que utilizam conceitos como multiprogramação (a possibilidade de mais de um programa ser executado por vez em um computador). Assim ao mudar a arquitetura das máquinas, é também obrigatoriamente mudada a maneira de operá-las.

Apesar dos *softwares* serem produtos comercializados por empresas, eles são também fruto de uma nova era da humanidade: a era digital. E é exatamente nesse momento da história que surgem pessoas como o filandês Linus Benedict Torvalds que, em 1990, mudaria para sempre o conceito de fabricação de *softwares*.

Torvalds é o criador de um kernel que daria origem ao mais conhecido sistema operacional de código aberto que se tem notícia: o Linux. (FLYNN; MCHOES, 2002)

O kernel de um *software* é a parte central deste, ou seja, é a parte do *software* que se comunica diretamente com os dispositivos eletrônicos do computador, como por exemplo, o processador da máquina, a memória etc. O kernel do Linux é distribuído livremente e uma das particularidades desse sistema operacional é a possibilidade de qualquer um, com conhecimento suficiente em programação de máquinas, modificar o código fonte.

Desse modo, de forma colaborativa, o Linux é atualizado por indivíduos e grupos organizados espalhados pelos “quatro cantos do globo”. (BROOKSHEAR, 2001).

Seja um sistema operacional que possua o código de fabricação fechado (MacOS; Windows) ou um que possua o código aberto (Linux<sup>44</sup>), é certo que ambos funcionam quase que do mesmo modo. Ou seja, os processos e as funções realizadas se assemelham. Dessa forma, não faremos distinção, nesse trabalho, entre *softwares* sistemas operacionais de código livre e fechado, uma vez que em ambos vemos um funcionamento semelhante. Ambos possuem módulos que controlam o gerenciamento de processos no computador, módulos que controlam a memória, outros que controlam os arquivos e a entrada e saída de dados etc. Também não faremos distinção entre *software* aplicativo de código livre (Firefox) e *software* aplicativo de código fechado (Internet Explorer).

Com relação à divisão, os sistemas operacionais estão divididos de acordo com o tipo de *hardware* (computador) que controlam e os tipos de *softwares* aplicativos que “rodam” neles. Assim, eles podem ser: monousuários monotarefas;

---

<sup>44</sup> Apesar de não ser esse o enfoque do nosso trabalho, não poderíamos deixar de dizer, como observação, que é nesse ponto que o Linux se diferencia, por exemplo, do DOS, do MacOS ou do Windows: enquanto que estes últimos possuem apenas um autor de direito (a empresa que o desenvolveu) o Linux possui vários autores. Os *softwares*, como toda obra da criação humana, são protegidos por direitos do autor, para usar um *software* cujo código de fabricação é fechado e o usuário recebe uma licença de uso. No caso do Linux, o código é aberto: qualquer um pode utilizar modificar, reescrever sem permissão prévia do autor ou autores. É claro que as questões relacionadas a autor e autoria, aqui apontadas, estão diretamente ligadas ao ramo do jurídico. No entanto, acreditamos que nada invalidaria uma análise feita à luz dos escritos desenvolvidos por Bakhtin e o Círculo a respeito do tema autor/autoria.

monousuários multitarefas; em tempo real ou multiusuário. Os *softwares* sistemas operacionais organizam o *hardware* e todos os outros *softwares* aplicativos que nele estão instalados<sup>45</sup>.

Os sistemas operacionais, hoje em dia, estão presentes em outros *hardwares* que em nada se assemelham ao computador pessoal, tais como: o telefone celular, o *smartphone* etc. Entretanto, possuir um sistema operacional não é condição para que um computador funcione, se quisermos recuar um pouco no tempo veremos que o Eniac<sup>46</sup> não possuía um sistema operacional e mesmo assim funcionava a contento. Outro exemplo de computador que funciona sem um sistema operacional é o atual forno de microondas.

Um sistema operacional, como qualquer *software*, está sujeito a um ciclo de vida, apesar de ele não ser um produto que se desgasta. O que em computação chama-se de ciclo de vida do *software*, diz respeito ao momento em que um *software* foi desenvolvido, utilizado e modificado. Este modo de funcionamento do produto *software*, conforme nos mostra Brookshear (2001), acontece porque erros que não foram eliminados durante o processo de desenvolvimento, exigem durante o uso do programa, que sejam feitas modificações correspondentes em partes do *software*.

Julgamos relevante atentar para o sistema operacional do computador, uma vez que é o sistema operacional que controla a máquina bem como controla todos os outros *softwares* que são instalados nos *hardware*. Se pudéssemos estabelecer uma hierarquia para os *softwares*, poderíamos dizer que o sistema operacional estaria no topo da pirâmide e todos os outros *softwares* seriam dele dependentes.

## 2.1 O que são *softwares*?

Dado ao ineditismo do tema, não encontramos estudos na Ciência Linguística que trate do objeto em questão. Desse modo, as contribuições para a construção de

---

<sup>45</sup> Daí a importância em se conhecer mais da história de um sistema operacional, apesar de, para fins de análise, nos interessar apenas o *software* aplicativo.

<sup>46</sup> O Eniac, fabricado em 1946, é considerado o primeiro computador eletrônico de grande porte.

um conceito coerente sobre o lugar da inscrição da escrita em ambiente digital, sobre os *softwares*, virão, de forma substancial, de estudos realizados em outras áreas do conhecimento humano: da ciência da computação, do *design* voltado para objetos digitais etc.

A primeira definição encontrada na área é a de *software*, o qual, para Fernandes (2003), é uma descrição de máquina cujo funcionamento depende de outra máquina que carregue e interprete as informações que ele está formatado. Essa máquina de ordem superior é representada na figura do *hardware*, que pode ser desde um computador pessoal, a uma câmera digital, um terminal de auto-atendimento bancário ou mesmo um telefone celular.

É também à luz da ciência da computação que buscamos entender a organização dos *softwares*, uma vez que é dela que separamos (cf. SOUZA, 2007b) os dois estados do *software*: o estado produto e o estado processo. O *estado processo* refere-se à *linguagem de programação* escolhida para criar um *software*. Essa ferramenta (linguagem) foi desenvolvida para facilitar o processo de fabricação de *softwares*. Desse modo, os programadores poderiam fazer a descrição de máquinas de forma mais rápida, não precisando escrever o código diretamente em linguagem de máquina.

O outro *estado* do *software* é o que denominamos de *produto*, no qual o *software* é visto em sua *integralidade*, ou seja, é visto não só como um código, mas como um produto que é capaz de unir a si outros tipos de objetos: desde uma interface até outro *software* e estabelecer com eles, além das relações mediadas por humanos, relações automáticas (sem interferência de humanos). Além disso, verificamos a *dupla divisão dos softwares*, ou seja, vemos que um *software* pode ser dividido em *software sistema* e *software aplicativo*.

Nesse trabalho iremos nos ocupar do *software* em seu estado produto e levar em consideração, para análise, os *softwares* aplicativos. Conforme nos mostra Brookshear (2001), são exemplos de *softwares* aplicativos: planilhas eletrônicas, sistema de edição de textos como o *MS Word*, jogos, navegadores web como o Firefox ou Internet Explorer (*softwares* no qual a maioria dos gêneros digitais estão

acoplados) e *softwares* de desenvolvimento de programas. Um *software* sistema executa aquelas tarefas que são vistas nos sistemas operacionais – no Windows, no Linux entre outros – de certo modo, o *software* sistema operacional desenha o ambiente no qual o *software* aplicativo se desenvolve.

Dentro dos *softwares* sistema operacional podemos encontrar módulos de *softwares* utilitários, (são exemplos desses *softwares* os programas de comunicação por modem, *softwares* que formatam o computador etc) a distinção entre *software* aplicativo e *software* utilitário é bastante vaga e por vezes são tratados como homônimos.

O *software* sistema operacional tem o papel de se comunicar com o usuário do *software* e com os outros componentes – os *softwares* aplicativos –, para coordenar a execução das atividades dentro das máquinas. Todo *software* possui um shell. Shell é a parte do *software* que define a interface entre o *software* e o usuário, permitindo que haja uma comunicação natural entre ambos. É através dessa parte do sistema, que o sujeito, que utilizará o *software* poderá manipular signos, enviar comandos ao sistema etc. Em computação moderna, os shells executam essas funções através de interfaces gráficas, também conhecidas como GUI (graphical user interface). Essas interfaces gráficas nasceram após o desenvolvimento dos primeiros sistemas operacionais baseados em janelas e servem para permitir que haja uma interação natural entre o usuário e o computador.

### **2.1.1 – A interface do *software***

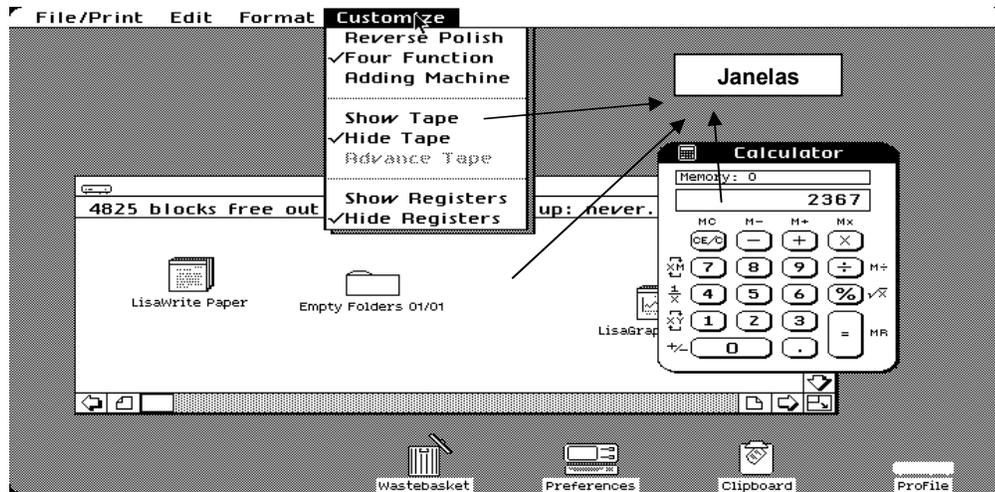
A *interface* gráfica de um *software* é um *elemento introdutório que auxilia o usuário a utilizar o software*. Ela é composta por elementos textuais, imaginéticos e sonoros e por sequências de controle que auxiliam o ser humano a interagir com os *softwares*. É através da interface gráfica (GUI) que nós, enquanto sujeitos utilizadores desses artefatos, poderemos manter relação com o *software* – poderemos digitar um artigo, escrever um e-mail, enviar um formulário etc. É também através da interface gráfica do *software* que reconhecemos os diversos tipos de *softwares* que existem.

A história da interface se confunde com a história dos primeiros *softwares* modernos, sabe-se que a interface gráfica como conhecemos hoje foi desenvolvida no início de 1972 pelos pesquisadores da Xerox PARC; como aponta Johnson (2001), a Xerox era um lugar intelectualmente desafiante<sup>47</sup>. Nessa empresa foi desenvolvida não só a primeira interface gráfica como também o mouse, as *janelas* e o mapeamentos de bits o qual nos deu a imagem visual para a organização das informações.

O desenvolvimento de janelas mudou substancialmente o modo como operar um computador. Aos poucos, essa invenção, viabilizada na prática pelos profissionais da Xerox, causou um impacto tão grande no campo da informática que hoje é praticamente impossível pensar em *softwares* sem interface gráfica baseada em janelas. Elas estabelecem limites entre os programas, através delas a delimitação do espaço do *software* é feito. Janelas são metáforas computacionais desenvolvidas para que o computador execute mais de um *software* ao mesmo tempo com alternância quase que simultânea. Elas representaram uma evolução tão grande no conceito de como operar um computador, que um dos sistemas operacionais mais usados no mundo recebeu o nome de Windows (janelas em inglês).

---

<sup>47</sup> Ao mesmo tempo em que era um centro de inovação, era também um centro incapaz de comercializar o que desenvolviam, é o caso, por exemplo, do mouse e da GUI que só saíram dos laboratórios da Xerox por conta da APPLE



**Figura 6** Lisa – primeira interface baseada em janelas.

Fonte: [http://www.harding.edu/fmccown/gui/Apple\\_Lisa\\_Office\\_System\\_1.0.png](http://www.harding.edu/fmccown/gui/Apple_Lisa_Office_System_1.0.png)

Os *softwares* e os documentos (diretórios) que antes eram acessados com comandos de textos (ver figura 7), na verdade com linhas de comandos ou sequências de letras, passaram a ganhar um novo modo de operação: comandos enviados através de janelas. Uma interface gráfica que tem por base janelas, ao invés de linhas de comando, permite ao usuário do computador executar inúmeras tarefas mais facilmente.

```

Directory of D:\
25/04/2009  18:16    <DIR>          _paulo_austran_-_4_seculos_de_poesia
18/03/2009  01:54          63.960.916  _paulo_austran_-_4_seculos_de_poesia.rar
08/04/2009  00:47             2.745      3_3_5v.gif
27/02/2009  16:23          330.360    758047077_08-03-2009.pdf
25/04/2009  18:16    <DIR>          adobe photoshop cs3 br<com plugins>
02/05/2009  17:31    <DIR>          aguinaldo_arquivos
11/04/2009  19:27             1.217      anotacoes.txt
25/03/2009  00:23          514.069    as_tecnologias_da_inteligencia.txt
01/12/2009  16:16    <DIR>          aulas
18/04/2009  22:03             648      celpe.log
18/04/2009  22:03             70.978     celpe07042009.pdf
23/04/2009  22:28        147.964.196  confli.5e11.death.suite.gfsilva.www.therebel
s.de(1).rmvb
13/04/2007  12:30          32.489     confli.5e11.death.suite.gfsilva.www.therebel
s.de(1).srt
16/07/2009  23:10    <DIR>          converteroutput
07/04/2009  08:53          2.286.510  cris.pdf
01/01/2008  02:51          699.659    da4.jpg
24/07/2009  17:01          846.336    disnova.doc
25/07/2009  00:21    <DIR>          downloads
17/11/2009  12:25    <DIR>          downloadsp
15/04/2009  22:57          314.368    ergo_aguinaldosouza.doc
09/11/2009  00:42    <DIR>          fotos
19/04/2009  13:43          33.141     introdução indios.docx
07/04/2009  23:06          148.570    linguagem_e_trabalho.pdf
12/01/2010  10:04    <DIR>          livros_de_bakhtin
17/12/2009  21:32    <DIR>          meu_desktop
25/04/2009  20:31    <DIR>          meus_arquivos_recebidos
27/04/2009  00:38    <DIR>          ms-office-2007
25/04/2009  20:11    <DIR>          musicas_soulseek
29/11/2009  10:56    <DIR>          new_folder
29/11/2009  10:55    <DIR>          nova_pasta
20/04/2009  17:43          174.776    novo_regimento.pdf
11/02/2009  01:17          534.098.021  offent2007-baixebr_2.zip
23/04/2009  12:42          277.464     oo.jpg
07/01/2010  15:43    <DIR>          pendrive
09/04/2009  01:46          1.512.051  phpejkjzp.pdf
23/04/2009  21:34          393.333    pict0405.jpg
10/04/2009  15:32          128.250    projeto_aguinaldo_reescritura.docx
26/03/2009  12:39          320.512    projeto_aguinaldo_revisandomarço2009.doc
14/04/2009  23:28          310.784    projeto_revisado.doc
26/04/2009  14:14          318.464    projeto_revisado.sampaio_25.4.09_(2).doc
14/04/2009  23:32          311.296    projeto_revisado_notasexplicativas.doc
20/04/2009  17:44          101.411    regimento_pgletas.pdf
23/04/2009  23:08          1.188.913  uuu.jpg
12/12/2009  15:10    <DIR>          www
          27 File(s)          756.341.477 bytes
          17 Dir(s)         52.272.050.176 bytes free

D:\> aguinaldo
'aguinaldo' is not recognized as an internal or external command,
operable program or batch file.

```

Figura 7 – interface baseada em linhas de comando

Embora tenha sido a equipe da Xerox que elaborou a primeira interface baseada em janelas, que deu vida ao conceito de desktop<sup>48</sup>, é atribuído ao presidente da Apple<sup>49</sup>, Steven Jobs, o mérito de ter libertado o conceito de janelas

<sup>48</sup> Segundo o Houaiss 2.0a versão digital: desktop é a parte da interface gráfica de sistemas operacionais que exibe representações de objetos usualmente presentes nas mesas de trabalho, como documentos, arquivos, pastas e impressoras

<sup>49</sup> A Apple Inc. é uma empresa multinacional norte-americana que atua no ramo de aparelhos eletrônicos e informática famosa principalmente pela fabricação do computador de marca registrada,

dos laboratórios da Xerox PARC. O desktop inicialmente foi vendido em conjunto com o Lisa<sup>50</sup> e logo em seguida com o Macintosh. Esse último, com seu desktop, mudou radicalmente o conceito de experiência de usuário. Antes dele o computador era uma máquina inerte, um objeto misterioso entre o usuário e o microprocessador (JOHNSON, 2001).

A interface baseada em janelas se tornou uma entidade autônoma, um meio de comunicação. Tudo que veio depois do desenvolvimento da primeira janela pode ser considerado como evolução natural, não havendo praticamente mudanças nos padrões utilizados. Com as interfaces gráficas, o usuário do computador começou a se libertar de vários paradigmas, como por exemplo, o paradigma da exclusão, se antes quando o usuário desejava excluir um arquivo ou um programa, teria que digitar isso para que o computador pudesse interpretar, com o desenvolvimento das interfaces ele apenas arrasta para a lixeira. A interface gráfica baseada em janelas utiliza-se de manipulação direta Ver figura 8.

---

Macintosh, com seu próprio sistema operacional, Mac OS, entre outros produtos. Fundada por Steve Wozniak e Steve Jobs com o nome de Apple Computers em 1976. Fonte: [http://pt.wikipedia.org/wiki/Apple\\_Inc](http://pt.wikipedia.org/wiki/Apple_Inc).

<sup>50</sup> Lisa e Macintosh são os nomes dos computadores desenvolvidos pela Apple. O lisa foi um fracasso de vendas ao contrário do Macintosh que até hoje é um sucesso.

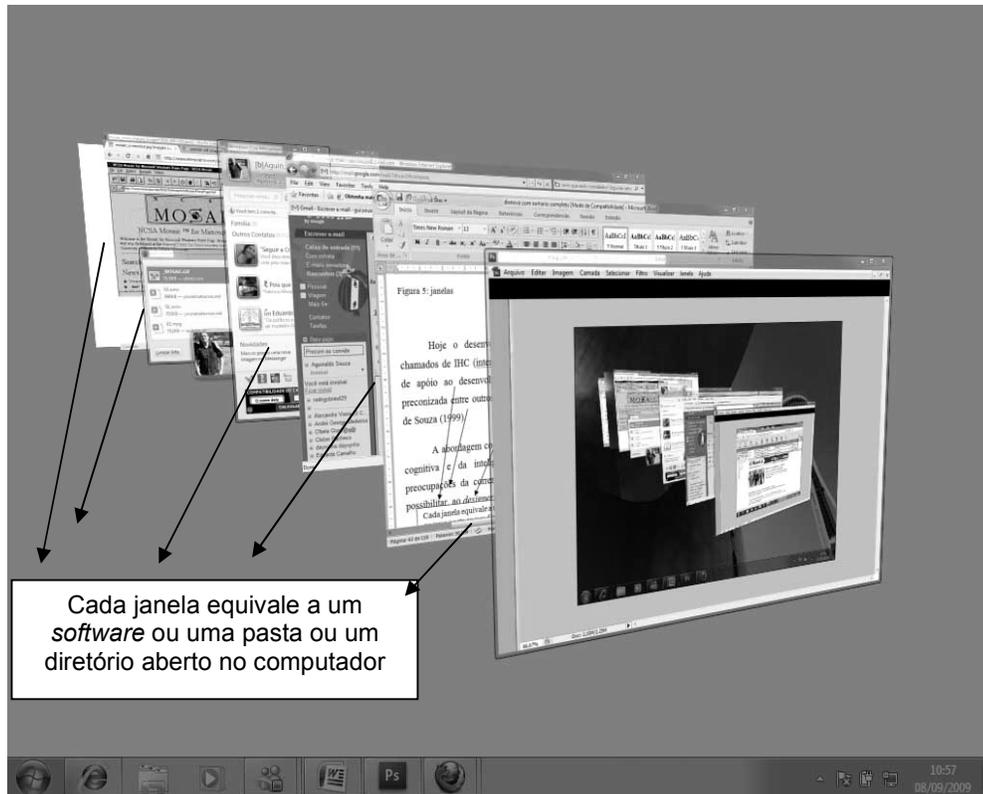


Figura 8 – janelas

Hoje o desenvolvimento de interface encontra-se no campo dos estudos chamados de IHC (interação humano-computador). Existem duas correntes que servem de apoio ao desenvolvimento de interface baseada em janelas: uma cognitivista preconizada entre outros, por Norman (1991) e uma de base semiótica, apreçada por De Souza (1993; 2005).

A abordagem cognitivista advém diretamente da psicologia cognitiva, da ciência cognitiva e da inteligência artificial as quais estudam a cognição. Uma das preocupações da corrente cognitivista, quando do desenvolvimento das interfaces, é possibilitar, ao *designer*, entender como ocorrem os processos cognitivos humanos.

A ideia básica é que modelos cognitivos que descrevem os processos e estruturas mentais (e.g. recordação, interpretação, planejamento e aprendizado) podem indicar para pesquisadores e projetistas quais as propriedades que os modelos de interação devem ter de maneira que a interação possa ser desempenhada mais facilmente pelos usuários. Como estas abordagens adotam uma perspectiva centrada nos aspectos

cognitivos do usuário, o *design* feito com base nelas é chamado de *design* de sistemas centrado no usuário (User Centered System Design – UCSD). (DE SOUZA et al., 1999, p. 13)

O *designer* de artefatos digitais deve ser capaz de construir um sistema (uma interface) que permita que o usuário, ao interagir com esse artefato, crie um modelo mental consistente com o modelo projetado pelo *designer*. Para ajudar os *designers* a entenderem como os usuários leigos utilizam uma interface, Norman (1991) argumenta que os *designers* precisam entender os processos mentais pelos quais os humanos interagem com a interface, propondo uma teoria da ação, segundo a qual a interação usuário-sistema é desempenhada num ciclo-de-ação.

Assim, os estudos voltados para interação homem-máquina, de base cognitivista, veem a interação com a interface, como uma travessia de dois golfos: o de execução e o de avaliação. A travessia do primeiro golfo está correlacionada à sequência de ações que o usuário terá que estabelecer para interagir com o sistema computacional; a segunda está correlacionada com a percepção da interface, a interpretação e a avaliação da meta a ser seguida para interagir com a máquina.

A crítica maior, feita à teoria cognitiva, é que ela focaliza apenas a interação usuário-sistema, desconsiderando o processo de *design* bem como o produto final do processo de *design*. Com o argumento de que a teoria de base cognitivista nega o rastro da subjetividade deixada pelos produtores dos *softwares*, De Souza (1999; 2005) propõe outra teoria para o campo da IHC: a da engenharia semiótica, que toma como base a semiótica de Peirce ([1839-1914] 2003).

Assim, como salienta De Souza et al., (1999, p.16) “nestas abordagens toda aplicação computacional é concebida como um ato de comunicação que inclui o *designer* no papel de emissor de uma mensagem para os usuários dos sistemas por ele criados”. Desse modo, a engenharia semiótica torna-se complementar a engenharia cognitiva, uma vez que enfatiza que o *design* não está só buscando construir a imagem do sistema, mas buscando comunicar ao usuário sua própria visão do sistema.

Esta mensagem tem como objetivo comunicar ao usuário a resposta a duas perguntas fundamentais: (1) Qual a interpretação do *designer* sobre o(s) problema(s) do usuário?, e (2) Como o usuário pode interagir com a aplicação para resolver este(s) problema(s)? O usuário concebe a resposta

a estas perguntas à medida que interage com a aplicação. Assim, esta mensagem é unilateral, uma vez que o usuário recebe a mensagem concluída e não pode dar continuidade ao processo de comunicação [de Souza, 1993] naquele mesmo contexto de interação. Além disso, como esta mensagem (a interface) é ela mesma capaz de trocar mensagens com o usuário, ela é um artefato de comunicação sobre comunicação, ou meta-comunicação. (DE SOUZA et al., 1999 p.443),

Com a engenharia semiótica, a interação entre o usuário-sistema deixa de ser o foco principal, dando lugar para a expressão do *designer* e ao processo de *design*. Dessa forma, para a engenharia semiótica, o que se deve atentar é a mensagem do *designer*, enviada de forma unidirecional ao usuário do *software*.

## 2.2 - Um olhar da metalinguística<sup>51</sup> sobre *softwares* e interfaces

Em relação à linguagem, reconhecemos que tanto a engenharia cognitiva quanto a engenharia semiótica não respondem satisfatoriamente as diversas perguntas que podem ser colocadas, em relação ao nosso objeto de estudo, no âmbito da Ciência Linguística<sup>52</sup>. Nesse sentido, buscamos, na teoria dialógica bakhtiniana, subsídios para um estudo das relações de alteridade que se estabelecem nas relações dialógicas inerentes a qualquer tipo de linguagem<sup>53</sup>, incluindo a relação linguagem-homem-máquina (SOUZA, 2009).

Assim, observando os *softwares* e as interfaces, iremos identificar nessas classes, entre todas as formas como se apresentam, que esses organismos também

---

<sup>51</sup> A metalinguística, postulada por Bakhtin (1981), é uma espécie de síntese entre a filologia e a filosofia, abarcando também a Linguística, a Antropologia e a Teoria da Literatura. Segundo Faraco (2003, p.91), Bakhtin no livro Problemas da Poética de Dostoiévski, fala explicitamente da necessidade de duas disciplinas distintas para os estudos da linguagem verbal: a linguística (para o estudo gramatical propriamente dito) e a metalinguística para o estudo das práticas socioverbais, concentrando seu foco particularmente em sua dinâmica e significação e tratando, entre outros aspectos, de relações dialógicas. Embora considerando ambas as disciplinas distintas, Bakhtin (1981), afirma que, na prática, os limites entre ambas são frequentemente violados. De todas as disciplinas linguísticas contemporâneas, como observa Faraco (2003, p.103), é a análise do discurso aquela que mais diretamente se aproximaria do projeto de uma metalinguística.

<sup>52</sup> Estudar os *softwares* ou os gêneros digitais tendo como ponto de apoio os postulados da engenharia semiótica ou a engenharia cognitiva nos levaria a olhá-los por um prisma enviesado, no qual as relações de alteridade inexistem, bem como os efeitos de sentido produzidos em tais relações. Assim, perguntas como: qual o papel do outro na constituição de um artefato informatizado? Como a presença desse outro pode interferir, por exemplo, na questão do estilo? Como o *software* colabora para o sucesso das interações entre dois sujeitos situados via, por exemplo, um comunicador instantâneo? Tais perguntas não encontrariam muitas respostas.

<sup>53</sup> Referimo-nos unicamente à linguagem natural, a linguagem humana. Não ao mecanismo de fabricação do *software* que define sua parte mecânica: linguagem de programação.

possuem uma ancoragem na metalinguística, disciplina proposta por Bakhtin (1981). O que entendemos por isso é que, tanto nos *softwares* quanto nas interfaces desses *softwares*, existem formas marcadas e não marcadas de presença do outro, deixadas por desenvolvedores<sup>54</sup>, que perpassam desde o processo de fabricação, até o produto em uso por outros sujeitos historicamente situados.

Tomando por base esse princípio, Peres (2007)<sup>55</sup> propôs uma nova forma de olhar os estudos sobre a IHC, partindo do arcabouço conceitual desenvolvido pelo Círculo de Bakhtin, envolvendo conceitos como autor, autoria, e dialogismo. Assim, enquanto a engenharia semiótica acredita em uma ‘mensagem concluída e unilateral’ enviada do *designer* para o usuário do *software*, olhando por um prisma dialógico, a autora (PERES, 2007) argumenta que essas mensagens são vozes que se materializam em interfaces. Desta feita, quem interage com um *software* – por exemplo, o *software* cliente de e-mail, que serve de receptáculo para o gênero digital e-mail –, além de interagir com outros sujeitos situados de forma diacrônica, encontra também responsividade de outros sujeitos situados, uma vez que as vozes que se fazem representar nas enunciações que aparecem nas interfaces dos *softwares*, – guiam-lhes as ações em um ambiente digital.

Vistas pelo enfoque do princípio dialógico inerente à linguagem, as interfaces dos *softwares* possuem, em si, marcas linguísticas das formas de presença do outro, podendo revelar aspectos de interesse, ao nosso objeto de estudo, tais como: as relações de alteridade, os signos, etc. De certa forma, essas formas de presença do outro são materializadas por vozes – no sentido bakhtiniano do termo, ou seja, a marca ideológica de um eu que se vincula a um lugar social –, encontra aí a sua maior ressonância. As relações dialógicas que perpassam os artefatos computacionais, são, pois, relações de alteridade entre um eu e um *outro*, situado em um lugar social. Daí decorre o olhar do analista para a interface, não como uma mensagem já concluída – conforme apregoa a engenharia semiótica –, mas como vozes, formas de presença do eu e do *outro* as quais guiam ações de uso, que conectam desenvolvedores e usuários, situados em pólos extremos.

---

<sup>54</sup> Os profissionais envolvidos na criação de um *software*: engenheiros, programadores, designers etc.

<sup>55</sup> Até onde temos ciência, essa tese defendida na pós-graduação em psicologia na UFPE, foi a primeira a tratar da questão.

É mais especificamente no trabalho da equipe que irá desenvolver a interface do *software* – Graphical User Interface (GUI) – que as questões relacionadas a subjetividade tornam-se mais expressivas, considerando que esses profissionais de *design* têm, por missão, comunicar ao usuário do *software*, o que ele poderá fazer com um *software*. Assim, a pressuposição de um outro, a quem o *software* irá servir, pode interferir 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<sup>56</sup> enquanto estão desenvolvendo as interfaces dos *softwares*.

A ideia de comunicar algo a alguém e fazer com que esse outro, em potencial, assumam uma posição axiológica frente a um chamado que se manifesta em um artefato computacional (ver na figura 9 as enunciações dos desenvolvedores dos *softwares* para que o usuário tome uma atitude responsiva), constitui um dos trabalhos mais sujeitos a interferência de terceiros dentro de uma fábrica de *software*.

---

<sup>56</sup> Os profissionais envolvidos na fabricação de *software*: *designers*, engenheiros, programadores, gerentes de produto etc



**Figura 9 o software de e-mail acoplado no software de navegação Mozilla**

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 subjetividade 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>57</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 sógnica 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>58</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

---

<sup>57</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>58</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*.

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.

Vejamos a figura 10: nela temos um *software* aplicativo de código fechado, chamado de Internet Explorer, esse *software* de navegação na Internet, ao ser aberto, envia para o sujeito usuário a pergunta ‘what do you want to do next?’. Essa enunciação que não é propriamente uma enunciação do *software*, mas uma enunciação deixada pelos profissionais que o fabricaram, já está pré-formatada, ou seja, a forma como se apresenta, a tipologia, as cores, a organização das opções a serem escolhidas tudo isso que em conjunto marcam o que chamamos de estilo, é pré-dado.

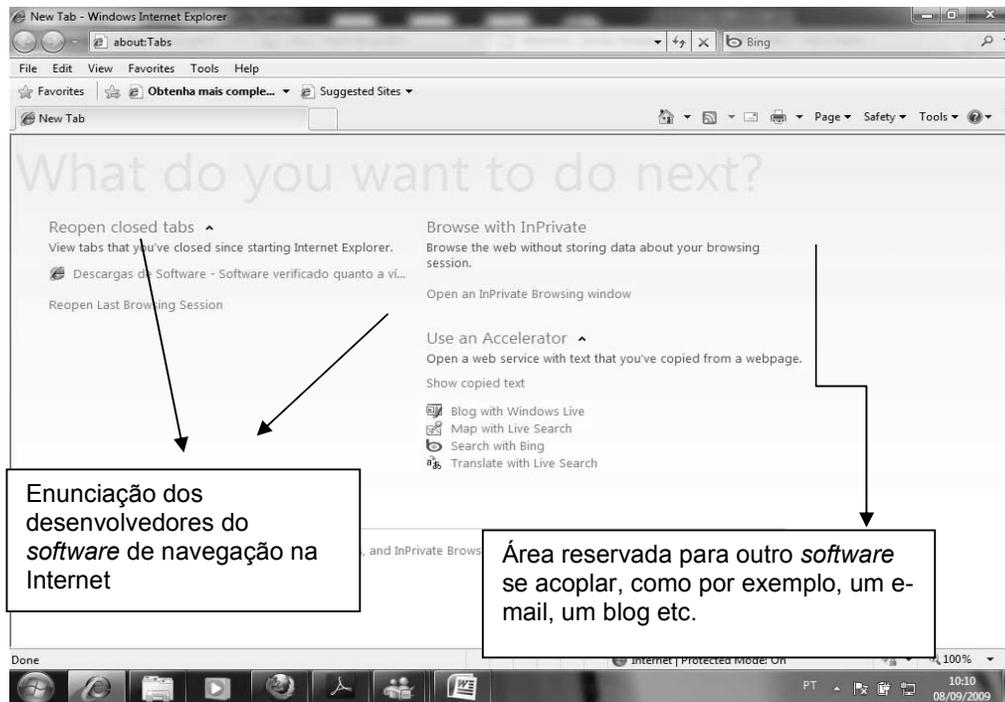


Figura 10 - *software* Internet Explorer

No trabalho prescrito, na produção de interfaces, a marca pessoal, dos protagonistas da atividade, aparecerá na escolha das cores, dos botões ou mesmo na escolha de uma tecnologia que permita certo acabamento dos enunciados: se serão sublinhados, se estarão em negrito, qual o tamanho da fonte etc. Assim, gêneros como o bate-papo, o curriculum *on-line*, possuem, em si, formas estilísticas já pré-concebidas pelos protagonistas da atividade. Essa determinação estilística acaba possibilitando, e ao mesmo tempo limitando, a margem de ação de outros protagonistas da atividade e dos usuários dos *softwares*. A estes últimos cabe interagir com essas formas e aplicar – quando possível –, suas próprias marcas estilísticas. Em geral essas marcas são vistas em formas típicas de enunciados, em gêneros digitais: as abreviações dos chats, os *emoticons*, etc, são exemplos disso.

## 2.3 Como os *softwares* são fabricados<sup>59</sup>: um olhar ergolinguístico

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>60</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>61</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.

---

<sup>59</sup> Vale ressaltar mais uma vez que, para as observações feitas sobre a fábrica de *software*, valemos de nossas memórias enquanto trabalhadores (*designer* de artefatos digitais) de uma empresa de desenvolvimento de *software* em Recife. Cargo que ocupamos de janeiro de 2004 até meados de 2008. Desse modo, para as observâncias aqui feitas não fizemos coleta de dados *in loco* para análise

<sup>60</sup> Ergolinguística é a palavra utilizada para se referir aos estudos sobre Linguagem e Trabalho.

<sup>61</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

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>62</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>63</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.

---

<sup>62</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>63</sup> A explanação aqui iniciada diz respeito ao desenvolvimento de *softwares* de código fechado

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>64</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*.

---

<sup>64</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.

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>65</sup>, os atos de uns não permanecem estrangeiros aos outros. Ao contrário, os atos de uns complementam a atividade dos outros.

---

<sup>65</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.

### 2.3.1 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.

### **2.3.2 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.

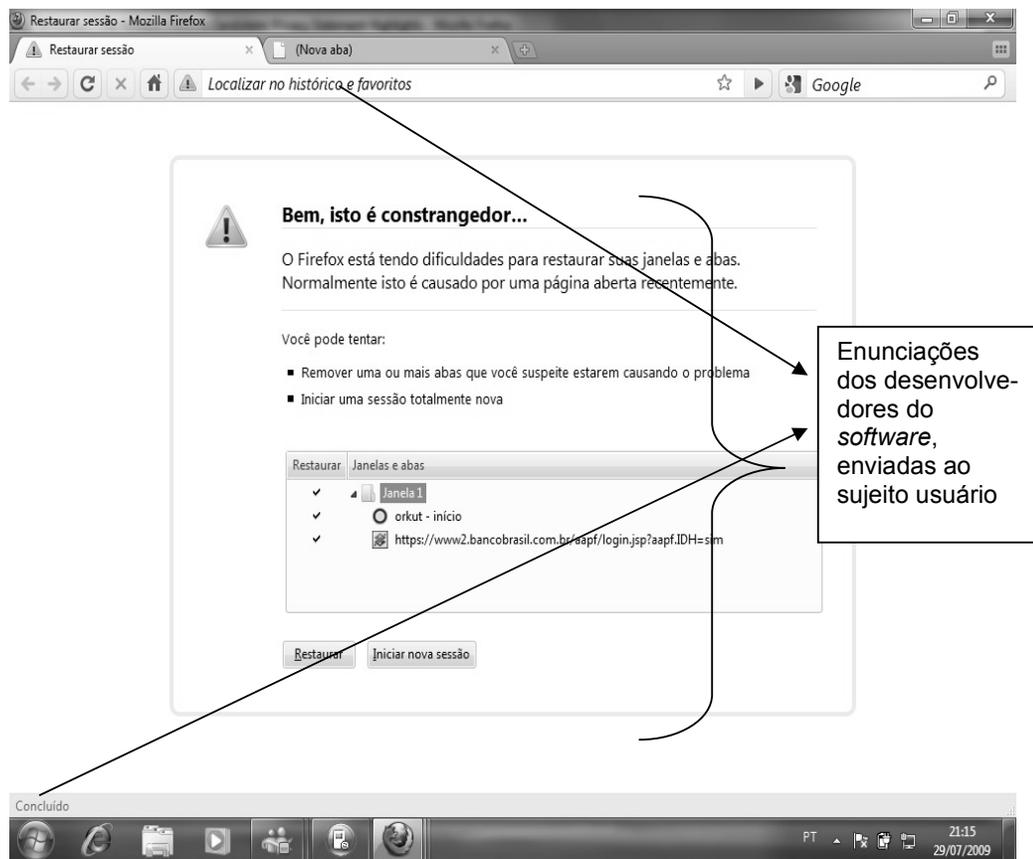
É mais especificamente no trabalho da equipe que irá desenvolver a interface do *software* – Graphical User Interface (GUI) –, que as questões relacionadas ao estilo, a intersubjetividades, a posições valorativas, a alteridade, se tornam mais expressivas. Esses profissionais de *design* têm por missão 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, assuma 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 11, 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 11 - 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, como são tomadas as interfaces, têm por objetivo mobilizar o usuário do *software*, de orientar suas ações. Essa orientação primária – em primeiro momento facilitar 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.

## 2.4 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>66</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>67</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*, são de alguma forma 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>68</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*

---

<sup>66</sup> Os *designers* são os responsáveis por desenvolver a forma arquitetônica do *software*

<sup>67</sup> É nesse momento que a forma mecânica do *software* é desenvolvida, retomaremos esse ponto nas conclusões desse trabalho.

<sup>68</sup> Conforme já apontou Peres (2007)

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.

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>69</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.

---

<sup>69</sup> Clot, Y. *La fonction psychologique du travail*. Paris: PUF. 1999

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.

Após verificarmos que o *software* é um produto produzido por sujeitos historicamente situados e que nas interfaces desses *softwares* existem formas da presença de *outrem*, chegamos ao Capítulo 3 deste estudo. Nele veremos que essas formas de representação são vistas não só em formas de enunciados, mas são também percebidas nos signos que compõem a interface. Entretanto, antes de sistematizarmos tal argumento, teremos de fazer um breve recuo, é assim que iniciamos o Capítulo 3, retomando a origem da rede Internet e do primeiro *software* que serviu para navegação na rede. Julgamos relevante fazer essa retomada uma vez que são tais *softwares* de navegação que permitem manter contato com a gama de gêneros presentes na Internet. Com isso, estamos respondendo o terceiro objetivo deste trabalho, estaremos atentando para os aspectos mecânicos e arquitetônicos do *software* e verificando como eles colaboram para o aparecimento dos hipertextos, dos gêneros.

## **CAPÍTULO 3**

Breves considerações sobre redes de computadores e a noção de *software* como objeto sógnico; a construção de sentidos nos *softwares* e os pontos de alteridade

### 3 Implicações para o estudo dos gêneros e dos hipertextos

A existência de hipertextos e gêneros digitais deve-se, nitidamente, ao desenvolvimento do primeiro *software* baseado em janelas para navegação na rede Internet (o primeiro *software* com interface gráfica), o qual permitiu o acesso a outros computadores via WWW. Dessa forma, as observâncias sobre signos, neste tópico, dirão respeito unicamente à presença destes nos ditos *softwares*, os quais permitem interações mediadas. Assim, de início, vemos que é na rede Internet que os gêneros digitais e os hipertextos<sup>70</sup> ganham mais visibilidade<sup>71</sup>.

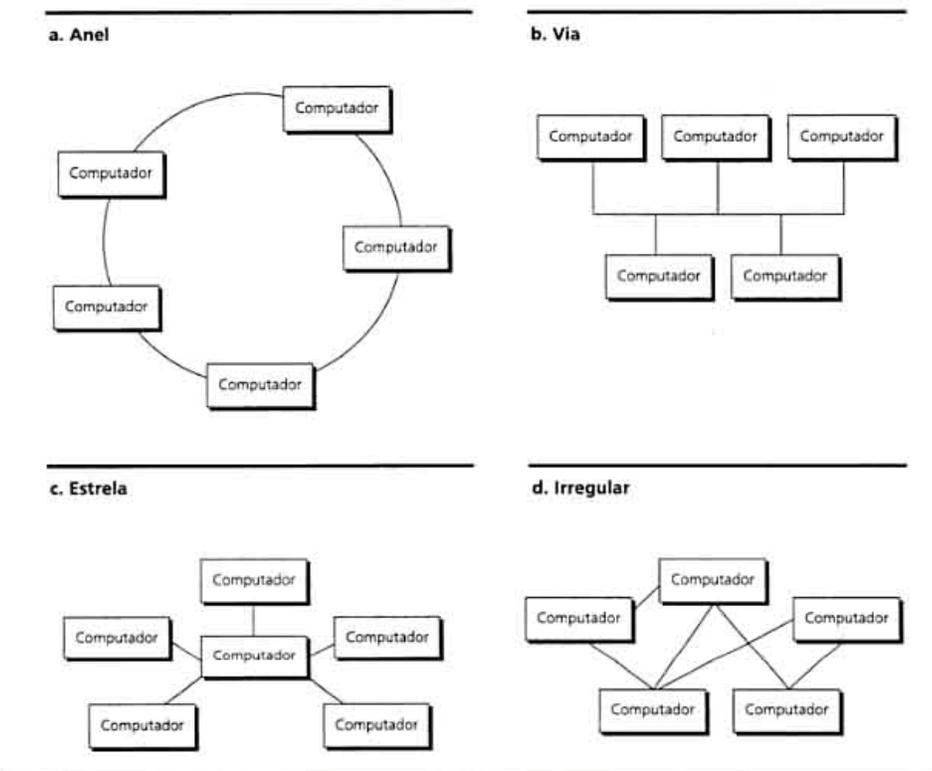
A Internet é uma rede de redes. Redes de computadores são formadas por máquinas independentes que efetuam transferência de arquivos através de conexões de redes utilizando-se dos *softwares* de navegação, chamados também de *softwares* aplicativos. Em relação ao estudo das redes de computadores, Brookshear (2001) nos mostra que existem duas modalidades classificatórias: de um lado, as redes locais (também conhecidas como redes fechadas), e, de outro, as redes abertas. Como exemplo de redes locais, temos as redes de computadores domésticas, que conectam dois ou três computadores; e como exemplo de rede aberta, rede de longa distância, a Internet.

Outro modo de classificar as redes é levar em conta suas topologias, assim temos redes em anel, em estrela, em via ou irregular. Brookshear (2001) indica que a conexão entre computadores não precisa ser necessariamente física e que as redes sem fios já estão cada vez mais comuns. É o caso, por exemplo, das redes 3G ou das redes de transmissão por ondas de rádio, nas quais o sinal se propaga em várias direções e pode ser utilizado para criar uma rede de vias (vide figura 12 b).

---

<sup>70</sup> De acordo com Lima-Neto (2008 p.20) “no Brasil atualmente, existem duas correntes no tocante ao estudo do hipertexto: de um lado, aqueles que defendem que o hipertexto em si não tem nada de novo, a não ser a diferença no suporte e na forma e rapidez de acessamento (KOCH, 2003; POSSENTI, 2002; COSCARELLI, 2003, 2005, 2005a; RIBEIRO, 2005, 2008); de outro, os que defendem que o hipertexto possibilita um novo modo de enunciação: o digital (XAVIER, 2002, 2007; ARAÚJO, 2003, 2006; KOMESU, 2005a, 2005b; BRAGA, 2005)”.

<sup>71</sup> Assumimos a existência de hipertextos off line como o sistema de ajuda de alguns *softwares*

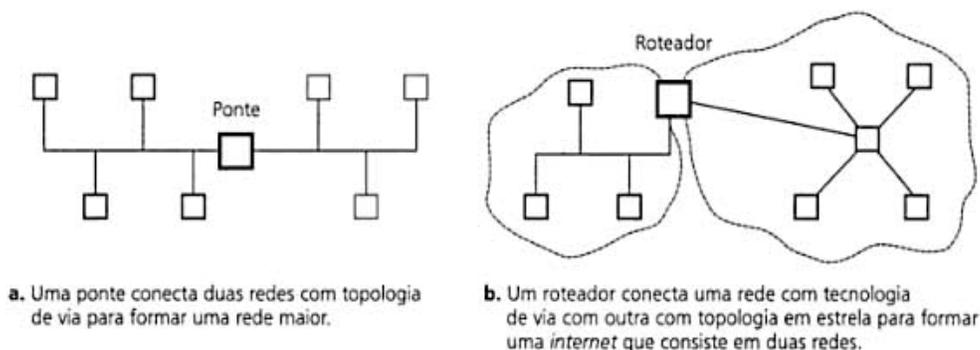


**Figura 12 topologia das redes**

Fonte: Brookshear (2001, p.128)

Quando existe uma compatibilidade entre redes existentes, a ligação entre elas é feita por meio de um dispositivo chamado de ponte. Entretanto quando é necessário unir duas redes de características opostas, por exemplo, uma rede em formato de estrela (figura 12 c) que é incompatível com a rede em formato de via (figura 12 b), utiliza-se para isso um roteador o qual passa a construir uma rede de redes permitindo que as redes originais continuem a existir e operar de forma independente, porém ligadas por um *router* (um computador pertencente às duas redes) o qual faz as conversões de idiosincrasias das redes originais (ver Figura 13 b) possibilitando assim que os dados, as mensagens, possam circular entre ambas. A esse tipo de conexão dá-se o nome de Internet. Ver figura 13.

A rede Internet nasceu nas últimas décadas do século XX, pelas mãos do departamento de defesa dos Estados Unidos. Inicialmente esse departamento montou um projeto intitulado de ARPA (Advanced Research Project Agency) o qual tinha por objetivos encontrar pesquisadores no mundo universitário, na área de tecnologia, para alcançar superioridade tecnológica em relação à União Soviética.



A distinção entre uma ponte e um roteador.

**Figura 13** diferença entre ponte e roteador

Fonte: Brookshear (2001, p.129)

Dentro do projeto Arpa, foi criado um grupo de pesquisa denominado de Arpanet cujo objetivo era desenvolver o que na época chamavam de computação interativa, o que poderia ser traduzido como a capacidade de várias instituições trabalharem *on-line*.

Para montar essa rede de computadores, como assinala Castells (2003), a Arpanet valeu-se de uma tecnologia inovadora, desenvolvida de forma independente por Paul Baran, em um centro de pesquisa da Califórnia. Essa tecnologia chamada de comutação por pacotes permitia a comunicação entre computadores, por redes, de forma descentralizada. As primeiras redes que foram interligadas estavam localizadas em universidades dos Estados Unidos. O passo seguinte, prossegue Castells (2003, p.14), foi fazer com que a Arpanet pudesse ser interligada com

outras redes de computadores. Isso só foi possível a partir da criação de protocolos de comunicação chamados de TCP/IP.

A tecnologia TCP/IP foi comercializada para empresas de computadores pessoais, as quais a incluíram em seus *softwares* sistemas operacionais. Em 1983 preocupados com a segurança das redes, que até então servia para a comunicação entre universidades e agências militares, o Departamento de Defesa Americano desmembra a Arpanet e cria a MILNET, essa serviria apenas para fins militares. A Arpanet passa então a se chamar Arpa-Internet.

Em 1990 a Arpanet era praticamente obsoleta sendo retirada das mãos dos militares e entregue à administração do National Science Foundation. Entretanto, essa permuta administrativa durou pouco, uma vez que a tecnologia de ligação em rede de computadores já estava presente em todos os computadores pessoais comercializados, ou seja, os alicerces da interconexão de computadores já estavam praticamente formados. O sucesso dessas interconexões se deve também ao desenvolvimento do Modem, que permitiu a comunicação e a transferência de arquivos entre computadores pessoais. Com o fim da ArpaInternet, em 1995, a Internet passou a ser comercializada por empresas privadas, assim surgiram os provedores de acesso<sup>72</sup>.

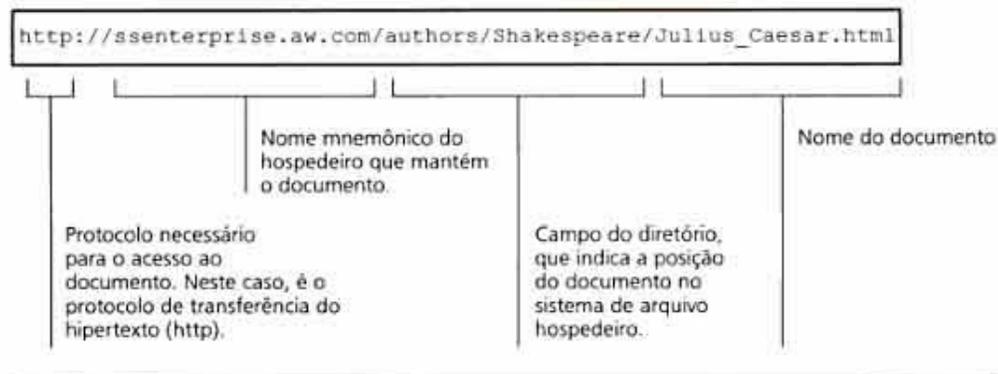
O que permitiu a Internet alcançar o mundo todo foi o desenvolvimento da WWW. “A WWW é uma aplicação de compartilhamento de informação desenvolvida em 1990 por um programador inglês, Tim Berners-Lee, que trabalhava no CERN, o laboratório Europeu para a Física de partículas baseado em Genebra” (CASTELLS, 2003, p.17). Berners-Lee desenvolveu um *software* que permitia obter e acrescentar informações de e para qualquer computador conectado com a rede Internet.

Depois desse *software* de navegação na rede vieram outros fabricados por empresas como a Netscape Communications, que criou um *software* de navegação chamado de Netscape Navigator, em 1994, e a Microsoft, em 1995, com o seu Internet Explorer.

---

<sup>72</sup> Empresas que permitiam a comunicação entre o computador pessoal e outros computadores espalhados pelo globo. Funcionando como um intermediador do processo.

Hoje a Internet é uma combinação de redes locais, com redes de longa distância operando via protocolo de hipertexto. É no *software* cliente (Internet Explorer ou Firefox), que reside no computador do usuário, que a WWW é acessada. O *software* cliente ou *browser* é o responsável por permitir, através de protocolos de redes, que o usuário obtenha acesso à rede Internet, aos hipertextos e aos *softwares* que permitem comunicação entre dois sujeitos situados. Ver figura 14



**Figura 14 protocolos de redes**

Fonte: Brookshear (2001, p.133)

Com o desenvolvimento do primeiro programa que permitia a navegação em rede por computadores, nascia a ferramenta que daria origem a uma infinidade de gêneros que *vemos através da tela de um computador* (Ver figura 15). Uma das funções desses programas de navegação na rede é transformar a linguagem de produção de *softwares*, ou sistemas *on-line*, em objetos perceptíveis por humanos.



**Figura 15 primeiro software de navegação na rede Internet: Mosaic**

Fonte: [http://img.zdnet.com/techDirectory/\\_MOSAIC.GIF](http://img.zdnet.com/techDirectory/_MOSAIC.GIF)

Esses *softwares* – *browsers* – manipulam diversos documentos, decodificando e organizando o modo como se apresentam. Assim, por exemplo, o código que dá origem ao *software* cliente de webmail 'http://webmail.souza.pro.br' antes de ser interpretado pelo programa de navegação (o Internet Explorer, o Firefox, o Chrome entre muitos) é da seguinte forma<sup>73</sup>:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pt-br"> function
adEvento(tO, tE, tF)
{ return tEvs.add(tO, tE, tF); }
/*  */</pre>
</div>
<div data-bbox="167 882 876 914" data-label="Footnote">
<p><sup>73</sup> O código exposto serve só como amostra e está editado, não representando o <i>software</i> on line por completo.</p>
</div>
```

```

function createCookie(name,value,days) {
{ var date = new Date();
  date.setTime(date.getTime()+(days*24*60*60*1000));
  var expires = "; expires="+date.toGMTString();
<head>
  <title>ULTRAmail</title>
  <link rel="shortcut icon" href="http://webmail.souza.pro.br/favicon.ico" />
  <meta http="content-type" content="" />
  createCookie(days) {
{ var date = new Date();
  date.setTime(date.getTime()+(days*24*60*60*1000));
body { background: #00474B; text-align: center; }
.cb { font-size: 1px; height: 2px; margin-top: -2px; clear: both; }
/* # IE */
body { position: relative; width: 100%; height: 100%; font: 10px sans-serif;
color: #BBCCBB; }
#loginbase { width: 100%; position: relative; top: 50%; left: 0; }
#loginbase .middlealign { width: 100%; position: relative; top: -50%; left: 0; }
/* # Compliant */
body { display: table; }
#loginbase[id] { display: table-cell; vertical-align: middle; position: static; }
#loginbase[id] .middlealign { position: static; }
  var expires = "; expires="+date.toGMTString();<h2>Login</h2> </div> <div
id="copyright">
  ULTRAmail - A sua solução de e-mails <br/>
  Copyright © 2001-2009. Todos os direitos reservados. </div>
</div></div></body> </html>

```

Uma vez decodificado pelo *browser* de navegação o e-mail ganha contornos operáveis por humanos, passando a assumir a forma representada na figura 16



Figura 16 webmail

É comum que esses *softwares* aplicativos de navegação na rede Internet possuam a capacidade de manipular, além de textos e hipertextos, outras formas, tais como: sons, vídeos e imagens. Tais *browsers* deram origem a gêneros e aos hipertextos.

Se nos *softwares* mantemos relação direta com a escrita, enunciando, com outros sujeitos situados etc., são também nos *softwares* que podemos interagir e construir textos em forma de hipertextos. Tudo o que está em um *software* é ou pode vir a ser hipertextual, ou, às vezes, é puro hipertexto. Nossa ideia sobre hipertexto é de certa forma condizente com a opinião que o identifica como um modo de enunciação digital<sup>74</sup>. De fato, o hipertexto é um modo de enunciação digital que se materializa, ou melhor, que só se realiza em um *software* (SOUZA, 2009). Assim, entendemos como hipertexto todo texto escrito em um *software* e ligado a outros textos ou vídeos/imagens por hiperlinks. Esse *software* pode ser desde um processador de textos até um webmail.

<sup>74</sup> 'Um modo de enunciação digital' é como Xavier (2002) define hipertexto.

### 3.1 Os signos e os *softwares*

Quando a Xerox desenvolveu a primeira GUI (graphical user interface) na década de 70, do século passado, ela abriu o mundo da informática para um campo ainda pouco explorado: o da manipulação direta através de elementos semióticos<sup>75</sup>. Assim, quando a primeira interface gráfica baseada em janelas foi desenvolvida o mundo passou a interagir com um artefato informatizado cuja base são os objetos sígnicos. O estudo dos signos é bastante antigo e por demais complexo, por isso mesmo, a breve retomada que iremos começar aqui só serve para situar nossas observações em relação a esses objetos em ambiente digital. Desse modo, o panorama que iremos traçar não tem a pretensão, de forma alguma, de apresentar, em detalhes, todas as escolas e doutrinas, que tratam da questão do signo<sup>76</sup> bem como não temos pretensão aqui de fazer um levantamento histórico de tais estudos.

O signo, antes mesmo de ser objeto de estudo dos linguístas, psicólogos, psicanalistas, lógicos entre outros, já era objeto de investigação de Platão e Aristóteles, dos estóicos etc. Durante anos houve grande divergência quanto a denominação que seria dada aos estudos dos signos, assim, para os linguístas, representados por Ferdinand de Saussure (1997)<sup>77</sup>, tal estudo recebeu o nome de semiologia e ocupava-se do signo no texto e das estruturas sígnicas verbais (semas, lexemas etc.) O enraizamento linguístico e o caráter diádico são comuns à essa orientação. Enquanto que Charles Sanders Peirce ([1839-1914] 2003) adotou o nome semiótica, a qual tem como ponto de apoio o signo platônico e o aristotélico – esquema triádico, filiando-se diretamente a uma corrente filosófica.

O termo semiologia, conforme nos aponta Nöth (2008), durante muito tempo foi o preferido nos países românicos, enquanto que os autores anglófonos preferiam utilizar o termo semiótica. Por conta desta disputa lexical, alguns semioticistas

---

<sup>75</sup> Diz-se que uma interface é de manipulação direta quando nela o sujeito usuário puder interagir sem a necessidade de conhecer alguma linguagem de programação ou de enviar algum comando por mensagem de texto, na manipulação direta os comandos ao sistema são dados através de objetos semióticos.

<sup>76</sup> Utilizaremos indistintamente no decorrer deste trabalho os termos: elementos semióticos, elementos sígnicos e semiótica com o sinônimo de signo.

<sup>77</sup> Utilizamos aqui a versão traduzida de 1916

começaram a elaborar distinções entre os termos semiótica e semiologia. Assim, semiologia passou a referir-se a uma teoria dos signos humanos, signos textuais, enquanto que o termo semiótica passava a ser utilizado para se referir a uma ciência geral dos signos. Essa disputa na nomenclatura dos estudos dos signos só chegou ao fim “com a interferência formal da Associação Internacional de Semiótica, no ano de 1969” (NÖTH, 2008, p.24). A partir de então o termo semiótica passou a ser o termo geral para as investigações que englobavam o estudo dos signos. Ainda assim, assegura Nöth (2006) *on-line*:

Como assuntos de terminologia, são raramente resolvidos por completo, em conferências internacionais, não é de se estranhar que sobraram uns resíduos de opiniões sobre diferenças entre os conceitos de semiótica e de semiologia, às vezes bem fundadas em sistemas complexos de teorias semióticas, às vezes também em concepções históricas hoje ultrapassadas. Um breve resumo de várias opiniões sobre o assunto é o seguinte:

- (1) Quem fala de semiótica se enquadra na tradição da teoria geral dos signos, especialmente de Charles Sanders Peirce, ao passo que os que preferem o conceito de semiologia se vêem na tradição semio-lingüística de Ferdinand de Saussure.
- (2) Enquanto a semiótica é a ciência geral dos signos, que inclui o estudo dos signos da natureza não humana, a semiologia é uma ciência humana que vai além da lingüística, estudando fenômenos trans-lingüísticos (textuais) e códigos culturais.
- (3) Em Hjelmslev, encontra-se a concepção de que a semiologia é uma metasemiótica que contém uma teoria dos mais diferentes sistemas de signos. Estes, por sua vez, são definidos como “semióticas”.
- (4) Semiótica e semiologia são sinônimos. Uma certa preferência do termo semiologia nada mais indica senão a proveniência do autor de um país de fala românica. Um argumento de purismo lingüístico, que se ouviu na França nos anos de 1970, era que o conceito de semiologia é uma melhor tradução do termo inglês *semiotics* para as línguas romanas e, por isso, é preferível ao termo semiótica, por um motivo puramente estilístico.

Como apontamos no capítulo 2, a semiótica de orientação peirceana e a teoria cognitiva são as bases para os estudos sobre interfaces no campo da computação. Na orientação peirceana, a relação entre o signo e o objeto corresponde ao interpretante, sem o qual não existiria signo. O significado do signo também não reside nele, mas na relação entre os signos. Em Peirce ([1839-1914] 2003), o significado é o interpretante<sup>78</sup> e este, por sua vez, ao ser um signo, irá sempre remeter a outro signo numa cadeia ininterrupta e sempre aberta a novos remetentes.

---

<sup>78</sup> Santaella (2003) argumenta que o interpretante não se refere ao intérprete do signo, mas a um processo relacional que se cria na mente do intérprete.

Os remetentes fazem com que um signo jamais tenha o mesmo significado cada vez que for interpretado, uma vez que o signo interpretado se converte em outro. Essa semiose<sup>79</sup> tem repercussões, conforme Ponzio (2008), sobre a concepção de sujeito, uma vez que o sujeito em Peirce ([1839-1914] 2003) é ele mesmo um signo. Um sujeito que se encontra constantemente deslocado, convertido em outro, um sujeito que é sempre signo interpretante de um signo anterior, desse modo o autor (PONZIO, 2008, p.162) argumenta que "o caráter dialógico do sujeito é, portanto, inevitável". Peirce parte do pressuposto de que as cognições, as ideias e até mesmo o homem são essencialmente entidades semióticas e "o fato de que toda ideia é um signo junto ao fato de que a vida é uma série de ideias prova que o homem é um signo", defende Nöth (2003, p.61).

Assim, enquanto o estudo do signo, para Peirce ([1839-1914] 2003), se constitui em três níveis: o sintático, o semântico e o pragmático, para o Círculo bakhtiniano, o signo está ligado à ideologia e "tudo que é ideológico possui um significado e remete a algo situado fora de si mesmo. Em outros termos, tudo que é ideológico é um signo" (BAKHTIN/VOLOCHÍNOV, 1996 p.31). Assim, tanto em Bakhtin quanto em Peirce, as estruturas dialógica e dialética do signo pressupõem necessariamente o diálogo e se configuram como dialógica. Uma vez que para algo ser signo tem que ser idêntico e diferente, ao mesmo tempo, de si mesmo (PONZIO, 2008).

O ponto de ruptura entre nosso trabalho e os trabalhos sobre Interação Humano-Computador produzidos no campo da computação, se dá basicamente na escolha dos postulados semióticos que compõem o objeto de estudo. Sendo assim, em detrimento de uma semiótica de cunho pierceano (a qual nos levaria a estudar o signo nas interfaces e nos gêneros como coisas que geram efeitos na mente dos intérpretes), optamos por estudar os signos nas interfaces dos *softwares* como marcas ideológicas de um eu e de um outro que perpassam por relações de alteridade.

---

<sup>79</sup> Refere-se ao processo de apreensão de um signo o qual gerará outro signo, poderia ser comparado em parte, ao processo dialógico.

De tal modo, ao estudarmos os signos pelo viés dos postulados bakhtinianos, estamos atentando para o fato de que os processos semióticos só refletem o mundo, refrata-o em signos. A refração é o modo como se inscreve nos signos a diversidade de contradições dos grupos humanos. A enunciação de um signo tem efeitos de sentido e por isso mesmo o signo é pluriacentuado<sup>80</sup> e plurívoco<sup>81</sup>, podendo ser sempre outro. A refração do signo é comparável aos milhares de fios ideológicos e é a ideologia uma condição necessária do signo. Segundo Faraco (2003), um signo não pode significar sem refratar, eles são espaços de encontro de diferentes índices sociais de valor, isto é, espaço da alternância entre um *eu* e um *outro*.

Por ser espaço de encontro de diferentes índices sociais de valor, um signo só existe na ideologia, assim afirmam Bakhtin/Volochínov (1996, p.30):

tudo que é ideológico possui um significado e remete a algo situado fora de si mesmo. Em outros termos, tudo que é ideológico é um signo. Sem signos não existe ideologia. Um corpo físico vale por si próprio: não significa nada e coincide inteiramente com sua própria natureza. Neste caso, não se trata de ideologia.

No entanto, todo corpo físico pode ser percebido como símbolo: é o caso, por exemplo, da simbolização do princípio de inércia e de necessidade na natureza (determinismo) por um determinado objeto único. E toda imagem artístico-simbólica ocasionada por um objeto físico particular já é um produto ideológico. Converte-se, assim, em signo o objeto físico, o qual, sem deixar de fazer parte da realidade material, passa a refletir e a refratar, numa certa medida, uma outra realidade.

Ideologia é, portanto, uma palavra que está diretamente ligada a outras significações (na análise do discurso; na sociologia etc). No Círculo, essa palavra é em geral utilizada para o universo das artes, da ciência, da filosofia, ou seja, é utilizada para fazer referência a uma área da atividade intelectual humana que expressa uma posição avaliativa<sup>82</sup>. Bakhtin admite que, (cf. KONDER, 2002), nem todos os signos podem ser substituídos por palavras. Porém, para ele, a palavra está inexoravelmente presente em qualquer ato de compreensão e interpretação de signos. O signo, no Círculo, é material, no sentido em que é um produto histórico-social; é essa materialidade que nos interessa (não importando se sejam signos

---

<sup>80</sup> Constituído por diversos valores sociais

<sup>81</sup> Constituído por diversas vozes sociais

<sup>82</sup> Para um exame detalhado das questões de ideologia em Bakhtin ver Faraco (2003, p.45)

verbais (a palavra) ou não-verbais (os ícones, símbolos etc) quando do estudo do signo nas interfaces dos *softwares*.

Bakhtin (1996, p.32) ressalta que, além de refletir uma realidade, “todo fenômeno que funciona como signo ideológico tem uma encarnação material, seja como som, como massa física, como cor, como movimento do corpo ou como outra coisa qualquer”. Assim, prossegue o teórico: “a realidade do signo é totalmente objetiva e, portanto, passível de um estudo metodologicamente unitário e objetivo”.

### 3.2 Signos na interface do *software*

Os signos nas interfaces dos *softwares* são objetos semânticos que se adaptam e se moldam em situações de interação. Eles evocam sempre do sujeito usuário, uma atitude responsiva ativa uma vez que são pontos de manifestação do *eu* e do *outro*. Dessa forma, ao estudarmos os signos, nas interfaces dos *softwares*, estamos atentando para questões dialógicas que atravessam esses objetos, o que necessariamente implica o estudo das relações de alteridade. A alteridade estabelecida não só entre um usuário e outro usuário, mas também a alteridade mediada pelos elementos sígnicos presentes nas interfaces dos *softwares*, quando do sujeito utilizador em interação.

Assim, quando nos propomos a estudar<sup>83</sup> os aspectos sígnicos<sup>84</sup> em uma interface, estamos atentando para algo mais que se estabelece nesse corpo, estamos atentando para os aspectos semânticos e ideológicos desses artefatos. E esses dois aspectos, por opção, serão tratados a luz dos escritos bakhtinianos.

Existe, por certo, a ideia de que o computador é usado através de signos, cada aspecto deste, da mais simples linguagem a uma interface de última geração,

---

<sup>83</sup> A *guisa* de registro: acreditamos que uma análise a luz da teoria dos signos proposta por Peirce, não invalidaria o exame desse par, mas, nos obrigaria a caminhar pelo caminho da cognição, uma vez que a semiótica de Peirce é uma semiótica cognitiva; se não o fazemos é mais por uma questão de escolha que de inviabilidade.

<sup>84</sup> Como signos consideramos os botões, os ícones, os menus e os símbolos presentes em uma interface.

são signos. Estudar os signos em *softwares*, entender como eles funcionam, é entender o ambiente (físico) das enunciações em meio digital. O estudo do signo, portanto, irá guiar o estudo, do estilo, das relações de alteridade que perpassam desde a fabricação ao uso de um *software*<sup>85</sup>. Partimos da premissa que o signo é um artefato meta-comunicativo<sup>86</sup> um objeto que pertence a uma ideologia só alcançada em contextos vivos de uso.

Assim os signos produzidos para serem usados nas interfaces dos *softwares*, divergem de outros criados em outra natureza pela função que desempenham, ocupando sempre uma função em torno de um objeto. Essa função pode ser hipertextual ou hipermodal (permitir a ligação de um texto ou de um vídeo ou imagem através de outra imagem ou vídeo) a uma função contextualizadora ou de sinalização (como é o caso do botão fechar de um *software* de navegação na Internet). Enquanto expressão e conteúdo possibilitam e transmutam um significado.

Se nos voltássemos para a função de um signo em uma interface de *software*, estaríamos nos voltando para o significado transportado e apreendido por um discurso<sup>87</sup>. Uma vez que, ao acionarmos um signo, este sempre gerará uma significação, para, a partir daí, gerar uma função. É a isso que chamamos de ‘diálogo de interface’ sobre o qual o sujeito usuário tomará uma atitude responsiva. Dito de outra forma: quando o sujeito usuário está interagindo com um *software*, seja navegando na rede Internet, seja digitando um texto no editor de textos, ele sempre estará em responsividade com os outros sujeitos situados (os desenvolvedores dos *softwares*) e estes, por sua vez, estabelecerão com o usuário, uma espécie de ‘diálogo’ mediado pelos signos da interface.

Por exemplo, na Figura 17 vemos um *software* reconhecido pelos usuários como ‘e-mail’. Ele está acoplado a um navegador web chamado de Firefox. Quando o usuário apertar um signo (um botão), representado por um X no canto direito do *software* de navegação na web, este enviará, de imediato, ao sujeito usuário, uma

---

<sup>85</sup> Veja para essa questão o tópico 2.3 deste trabalho, intitulado de “Um olhar da metalinguística sobre *softwares* e interfaces”

<sup>86</sup> Possuidor de uma dimensão sócio cognitiva razão pela qual é compartilhado por agrupamentos humanos.

<sup>87</sup> Idem nota 38

enunciação através de uma ‘caixa de diálogo’<sup>88</sup>, cabendo, ao sujeito usuário, a partir desta enunciação, tomar uma decisão: fechar, cancelar, salvar etc. Esses signos *a priori* nunca aparecerão sozinhos nas interfaces: basta o usuário apontar a seta do mouse para ele que uma enunciação irá emergir. Essa enunciação auxilia no processo de decodificação do signo e resolve os possíveis problemas de interpretação que o signo possa ocasionar ao sujeito usuário. O conhecimento do signo ou a função que ele exerce em um ambiente é também mediada por essas micro-enunciações.

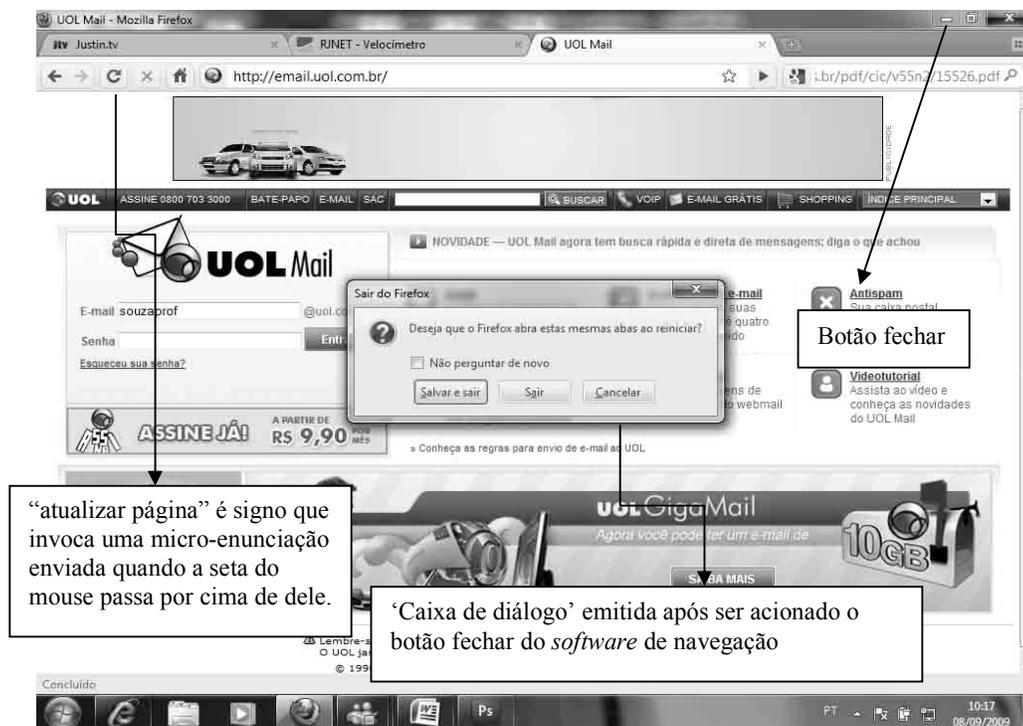


Figura 17 – signos e função

A função de um signo, em uma interface, é gerar significação. Por essa razão, estudá-lo é abrir caminho para o estudo do significado, do sentido. Entretanto, estudar o signo, de forma isolada, sem levar em conta sua arquitetura, não nos daria muitas respostas, uma vez que a significação, em si, não pertence a um elemento

<sup>88</sup> Na verdade essas enunciações foram alocadas nos *softwares* pelos protagonistas da atividade na hora de fabricar o produto

isolado. Ou seja, só quando passamos a considerar o todo arquitetônico (as funcionalidades dos *softwares* e as interfaces) é que um signo passa a adquirir significação plena. A significação, em si, pertence ao todo da enunciação. É por isso que um signo utilizado fora de um contexto mais amplo perde parte de seu potencial.

Na verdade, o signo pertence a um universo só compreensível dentro de um contexto, basta verificar o signo utilizado em uma sala de bate papo web (*emoticons*), e ver o quanto ele diverge em forma, significado e função dos criados para serem utilizados em outros ambientes. As interfaces dos *softwares* são carregadas de elementos semióticos. Esses elementos, repletos de significação, são fabricados para serem funcionais; fabricados para executarem uma ação, e.g., fechar, salvar, enviar confirmar, etc. São também os responsáveis pela comunicação entre o usuário e a parte mecânica do *software*.

### 3.3 As funcionalidades dos signos na interface do *software*

Ao estudarmos o signo em interfaces de *softwares*, como objeto detentor de um significado, estamos levando em conta o local em que esse signo está adentrado. Assim, um signo utilizado numa sala de bate-papo aberta, e outro utilizado numa sala de bate-papo que utilize outra tecnologia, por exemplo, IM (instante *messenger*) – *Messenger*, *Google talk* –, divergem, gradativamente, em forma, significado e função. O signo, utilizado em uma interface de *software*, diverge de outro criado para outro ambiente na função<sup>89</sup>. Contudo, a relação que o sujeito estabelece com o signo, não irá variar. A função do signo está ligada ao seu significado. Podemos considerar que há signos mais funcionais e signos menos funcionais. E essa funcionalidade é pré-determinada pelo sujeito desenvolvedor do *software* quando da construção do algoritmo que dará origem a esse *software*.

A interface do *software* é uma coleção de signos que são interpretados por sujeitos em interação com o sistema. Vejamos, por exemplo, a figura 18: temos a ‘tela’ (chamamos de tela a representação gráfica do *software*). Nela observamos

---

<sup>89</sup> Quando falamos em função do signo estamos falando da ação que esse signo pode desencadear, como por exemplo, fechar um programa, minimizar salvar etc.

vários signos que desempenham papel fundamental na construção do sentido e na organização da interface. Há signos visíveis no sistema (*software* fabricado para dar funcionalidades a outros). Nos signos do *software*, chamado de ‘e-mail’, vemos os ícones do canto esquerdo, abaixo, do nome ‘entrada’. Neles, além do próprio signo, há também um enunciado informando a função que ali poderá ser executada, a partir do momento que o usuário acione o signo. Esses signos também possuem uma funcionalidade. No caso em questão, a funcionalidade é hipermodal, ou seja: permitir o acionamento de um comando via ligação por links. A organização estrutural desses signos na interface, em geral, não varia. Eles possuem quase sempre uma mesma arquitetura, gerando o que, em *design* de ambientes digitais, chama-se de ‘imagem do sistema’.

Se olharmos mais para abaixo, perceberemos outros ícones pertencentes ao sistema operacional, como por exemplo, o símbolo PT, indicando que ali se trata de um botão para mudar o idioma do teclado, o computador com uma tomada – indicando uma conexão, o botão com a logomarca da empresa fabricante do *software*, indicando que ali se encontra o botão iniciar, etc. Cada signo, além de permitir que o usuário mantenha relação com outros sujeitos e com o próprio *software*, possui uma função.

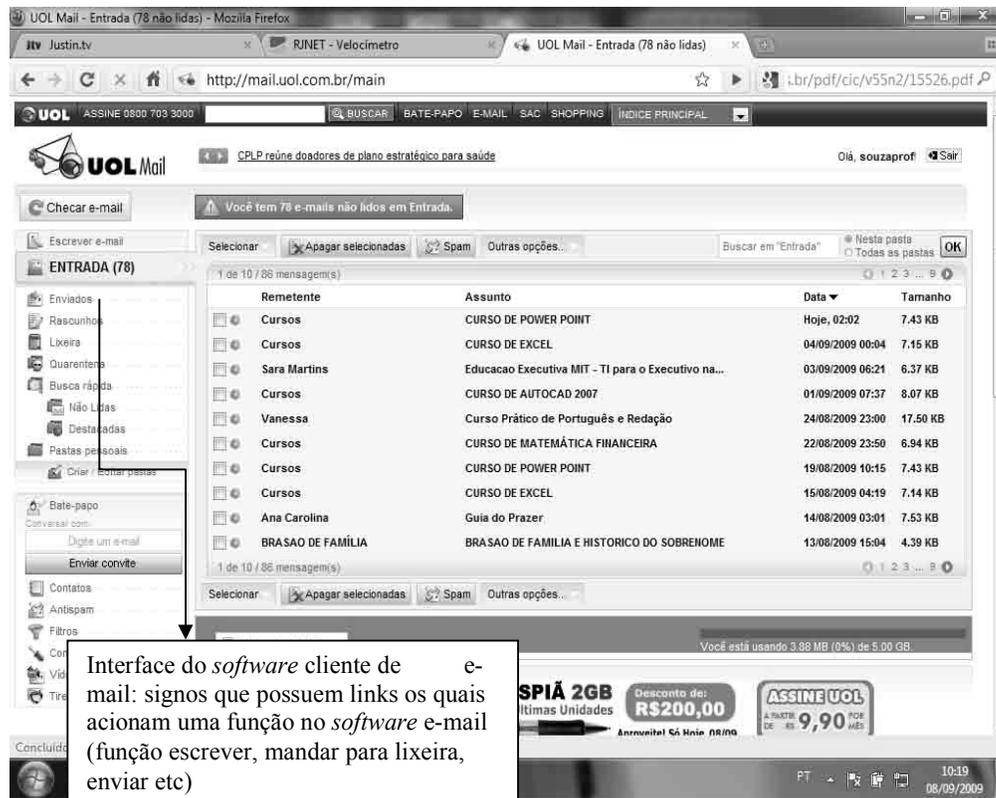


Figura 18 - 'tela do webmail'

A forma arquitetônica (interface) e o conteúdo do *software* são definidos pelos protagonistas da atividade (os produtores do *software*) com vistas à melhoria da comunicação. Os botões, as palavras, os menus ou as cores, são signos em potencial, e consistem em ser uma expressão que conduza o usuário a ativar interpretantes que o faça utilizar o sistema computacional ou um *software* aplicativo, no caso em questão, o webmail, de forma intuitiva.

Entendemos que o *software* é um produto no qual o protagonista da atividade envia aos usuários certas coordenadas (materializadas em forma de enunciações). Essas coordenadas são por assim dizer, pistas 'contextuais', materializadas através de signos verbais e não-verbais, as quais permitem ao sujeito utilizador interagir com o ambiente digital, estabelecendo desse modo uma relação dialógica (relação de sentido, relação de alteridade, responsividade etc). Dessa forma, essas 'pistas'

seriam um conjunto de proposições sistematicamente relevantes para o sujeito estabelecer relação com o todo, poder guiar-se.

Uma das funções mais importantes do signo, em especial na interface do *software*, é a de permitir que o sujeito estabeleça uma macrocompreensão – dada através de metáforas computacionais, p.ex.: botões, pastas, arquivos, salvar etc. Ao estabelecer essa macrocompreensão, o usuário (sujeito) estabelece, antes, associações de outra natureza: do contrário seria quase impossível guiar-se em ambiente digital. Essas informações perceptuais são, em parte, condicionadas pelo *designer*. As observações feitas sobre esse ciclo de ação que se desencadeia desde o momento de desenvolvimento de um *software* até o uso por sujeitos situados são aqui conduzidas pelo viés do dialogismo constitutivo do Círculo de M. Bakhtin (1997). Conforme aponta Brait (2005), dialogismo é tema central em Bakhtin, ele é um elemento constitutivo da linguagem e o princípio que rege a produção e a compreensão dos sentidos. É também o princípio que marca a fronteira entre um *eu/outro* que se interpenetram, sem se fundirem ou se confundirem.

Discutindo o dialogismo em IHC – Interação Humano Computador–, Peres; Meira (2006) retomam os postulados propostos por Bakhtin (1997). Para os autores, quem enuncia, emprega signos na interação. Tais recursos (*signos*) permitem gerenciar as inevitáveis incertezas e ambiguidades nas ações com vista à produção de sentido. O signo (cf. PONZIO, 2008 p. 120) caracteriza-se por se adaptar a contextos situacionais, sempre novos e diferentes bem como pela sua pluralidade e indeterminação semântica. O signo, em Bakhtin (1996),<sup>90</sup> não está reduzido a apenas um processo de estímulo-resposta, mas é um processo de interação entre duas consciências. Todas as manifestações, nos enunciados produzidos *no* ou *através de softwares*, estão, ou são, manifestações ideológicas constituídas por signos verbais (a palavra) e não-verbais. As relações que se estabelecem entre um signo verbal (palavra) e um não-verbal (*emoticons*, ícones, imagens, símbolos, menus etc.) fundamentam-se numa relação de complemento e são responsáveis pela contextualização, apreensão da realidade.

---

<sup>90</sup> Não fazemos a distinção que Bakhtin/ Voloshinov em *Marxismo e filosofia da linguagem* faz entre signo verbal e signo não verbal, por acreditarmos que no *software* cada signo não verbal se apóia em um signo verbal tornando-se por semiose um objeto plurisemiótico.

Os signos em geral, no *software*, operam uma função. Essa função pode ser de ordem operatória (como o botão 'sair', como o botão enviar) pode ser de ordem contextual, como a totalidade sígnica do *software*, pode ser de ordem discursiva (como os *emoticons*), enfim, o signo exerce funções variadas em um mesmo ambiente.

Essas funções possuem duas características: a primeira delas diz respeito à sua constituição, ao algoritmo que lhe deu origem, ou seja, liga-se ao processo de fabricação de *software*, *ao seu todo mecânico*; a segunda diz respeito à ordenação simbólica do signo. Essa ordenação simbólica só ganha contornos operáveis quando do desenvolvimento da interface. A interface (forma visível de um *software*) é uma expressão em potencial, e não de fato constituída através de signos – abstração do que pode vir a ser. Esses signos estão dispostos espacial e temporalmente e são também responsáveis pela comunicação usuário-sistema (sujeito/meio). O conjunto desses signos cria o que chamamos de arquitetônica enunciativa e são eles que permitem, ao usuário, diferenciar a gama de *softwares* que vemos, por exemplo, na grande rede. Esta representação semiótica de sentido só pode acontecer na medida em que tal objeto possa ser interpretado pelos interactantes. As variações dos signos estão ligadas ao meio: *softwares* que fundam o uso e podem variar de acordo com o mesmo. A rigidez mecânica dos *softwares* e suas características funcionais determinam o uso de um signo.

É de especial importância, para nós, o estudo dos signos, no *software* de navegação, na grande rede por entendermos que dele procederá todo o progresso em torno das noções de referência, contextualização, estilo, etc. A unicidade, o formato, a constituição – também a constituição dos enunciados –, o gênero, o contato imediato, as mediações feitas mutuamente, os complementos, a mudança de relação e a relação que se estabelece entre sistemas (linguísticos ou não) são dependentes e complementam-se numa relação dialética com o signo. A relação do signo com o *software* poderia ser resumida a uma relação funcional-contextual. A fórmula para essa relação precede o campo do linguístico, ampara-se no campo da significação e apreensão da realidade. A mudança de um signo, no *software*, mudará a relação entre sujeito e sistema computacional, entre sujeito e o gênero, entre sujeitos e sujeitos.

### 3.4 Hipermodalidade

Aspecto teórico e terminológico relevante que precisamos trazer aqui, antes de iniciarmos esse tópico, é a distinção necessária entre duas noções ainda pouco discutidas e pouco analisadas na literatura pertinente, e que se ligam diretamente ao problema do estudo dos signos: a multimodalidade e a hipermodalidade. Não vamos aqui retomar a literatura que foi ou está sendo desenvolvida a respeito, o que fugiria muito dos propósitos desta seção. Ainda assim, para uma primeira distinção, utilizaremos o termo *hipermodal* para os *diversos modos de representação semiótica apreendidos e materializados em um software*, cuja característica principal é o fato desses corpos poderem interagir entre si e com os sujeitos utilizadores, de forma automática ou através de links<sup>91</sup>. Dito isso, passamos a considerar os objetos que se aportam em um *software* não como multimodal já que eles não são constituídos por imagens estáticas, mas como *hipermodal*. Uma vez que tais corpos (hipermodais) exigem do sujeito que com ele interage, participação ativa. Seja através de acionamentos de links seja através de *feedbacks* recebidos através da interface dos *softwares*<sup>92</sup>. Participação ativa, então, é parte constituinte de qualquer sistema hipermodal<sup>93</sup> e é também o que o diferencia de um multimodal. A hipermodalidade poderia ser resumida como uma particularidade de um *objeto* que se aporta em um organismo de hipermídia, de um *software*.

Já os organismos de hipermídia são aqueles que estão materializados em um *software* cuja interface gráfica permite a junção de textos, de sons, vídeos imagens etc que se interligam de forma dinâmica através de sistemas de hiperlinks possibilitando com isso que usuários estabeleçam macrocompreensão entre enunciados e entre camadas de informações. Assim, ao enveredar pelos caminhos

---

<sup>91</sup> Segundo Gomes (2007 *apud* Lobo-Sousa 2009 p.140) o termo “hipermodalidade” refere-se à fusão de multimodalidade com hipertextualidade. Trata-se de uma maneira de nomear as novas interações entre os significados das palavras, imagens e sons da hipermídia, o que permite diferenciá-lo da multimodalidade presente nos textos, como nos jornais ou revistas impressos

<sup>92</sup> Assumindo que gêneros digitais são *softwares* que possuem algum propósito comunicativo

<sup>93</sup> Por hipermídia entendemos a combinação de sons, textos, vídeos e imagens em um organismo. A hipermídia, por conseguinte, une os conceitos de hipertexto e multimídia, gerando, assim, organismos hipermodais. Muitos dão ao termo hipertexto o mesmo significado que hipermídia. Entretanto, concordamos com Ted Nelson, inventor dos dois termos, quando sugere que hipermídia é o hipertexto acrescido de outras mídias e não puramente o texto escrito, como é o caso dos hipertextos. Neste trabalho utilizaremos as expressões hipermodal e hipermidiático como sinônimos

da hipermodalidade somos obrigados a considerar a estrutura que aporta os textos, os vídeos, as imagens e os sons em ambiente digital, ou seja, somos obrigados a considerar o *software*.

Não obstante, ainda assim, ao enveredar pelos caminhos da hipermodalidade, e não ao da multimodalidade, nos afastamos de análises centradas, por exemplo, nos conceitos inerentes da semiótica social e que têm, como seus fieis difusores, Kress e van Leeuwen (1996). Análises essas que partem de uma concepção de semiótica que difere gradualmente da que adotamos (para nós o signo é dialético). Os postulados semióticos, adotados por nós, ao longo de nossas investigações, filiam-se diretamente ao postulado no Círculo bakhtiniano (1996). Desse modo, ao afirmarmos que um *software* é ou pode vir a possuir elementos hipermodais, estamos levando em conta não um *minimum*, mas uma série de elos que se interligam sobre uma interface. Esses elos são regidos por forças tecnológicas as quais lhe dão certo acabamento, limitando e definindo o seu uso. Essas forças diferem, por exemplo, daquelas dos gêneros impressos.

Os processos hipermodais percebidos em um *software* são, por natureza, sem centro e são também partes de um processo que, em um menor grau, define-se como linear. Esses processos são, por assim dizer, as várias correntes e as várias forças que concorrem para sua criação. Daí o abismo que separa os gêneros multimodais dos gêneros hipermodais. Assim, ao trabalharmos com signos em ambientes digitais, não estamos só preocupados com os *emoticons* (que são marcas contextualizadoras de uma interação mediada) ou só com as imagens em movimento ou estáticas que se aportam em *softwares*. Estamos também levando em consideração a interface do *software* e seus signos. Consideramos signos, por exemplo, os domínios de navegação, os atalhos como a barra de ferramenta de um navegador web, os botões de navegação presentes em uma interface.

Ainda no tocante aos estudos sobre hipermodalidade, temos ciência da existência de trabalhos cujo escopo é o estudo da hipermídia. Alguns deles são produzidos no âmbito da linguística, como é o caso de Braga (2005), de Lemke

(2002); outros são desenvolvidos pelo viés da Ciência da Computação, como é o caso de Lee, K., Lee, Y.K. y Berra, P.B. (1997), entre outros.

Os *softwares*, pela própria constituição mecânica, podem acoplar outros elementos e se associar as mais diferentes mídias. Essas mídias são, por natureza, hipertextuais, ou seja, são não-lineares. Elas são multilineares e incorporam, em si, dois modos de ser: o modo autor (onde são criados os sistemas de nós e âncoras) e o modo usuário (onde ocorre a navegação).

É ao utilizar um sistema, um *software*, que o sujeito passa de um nível para outro numa progressão contínua. As hipermídias são partes dessa progressão: elas são partes de um todo não-linear, são organismos interativos, organismos que interagem com sujeitos em interação através da interface do *software*. Desse modo, os menus, os botões e os ícones da interface são objetos que colaboram para a construção de êxitos e das ações. A interface é completamente dependente do sistema ao qual está vinculada.

Assim, quando tratamos da hipermídia, não estamos atentando para uma parte do sistema semiótico, estamos levando em consideração a arquitetônica em que essa hipermídia está abrigada. Estamos levando em consideração o processo de descodificação dos signos que se agrupam em um *software*. Estudar a hipermídia como algo desvinculado dessa arquitetônica é tratá-la não como um corpo repleto de significação, mas como um processo de identificação, como um sinal inerte, isolado do todo arquitetônico.

Nenhum signo em ambiente digital permanece isolado. Por menor que seja sua forma ou por menor que pareça ser sua função, ele torna-se parte de uma unidade maior, de um todo arquitetônico, através da estrutura mecânica do *software*. É essa estrutura que permite que os *softwares* possam acoplar em si outras formas. O estado mecânico do *software* regula os usos que podemos fazer de um artefato informatizado, regula a própria dinâmica do sistema computacional e tudo mais que se junta a esse sistema computacional. O signo no *software* não existe apenas como parte de uma realidade acabada, mas como parte de uma realidade ligada a uma função mecânica. Isso não significa, obviamente, que o estado mecânico do *software* possa suplantam a significação do signo. É claro que, o valor, a representatividade do signo, e a sua estruturação semiótica, o levam sempre para

um estado de primazia sobre o estado mecânico do *software*. Preliminarmente, poderíamos dizer que a existência do signo nada mais é do que a materialização do estado mecânico do *software*. É essa sua real natureza. Essa forma mecânica do *software*, forma que permite a junção de outras semioses no corpo do *software*, permite que o *software* possa acoplar em si outros elementos além daqueles que estão em sua interface, os da hipermídia<sup>94</sup>.

---

<sup>94</sup> Sons, vídeos, imagens em movimento etc

## **CAPÍTULO 4**

Considerações sobre a interação com *softwares*. Aplicação da base teórica desenvolvida ao longo desse estudo.

## 4 Aplicação da base teórica

Para aplicação da base teórica escolhemos um *software* aplicativo, que funciona através de redes, chamado de e-mail<sup>95</sup>. Inicialmente iremos descrever sua parte mecânica, ou seja, a parte que define as funcionalidades do *software*. Em seguida, iremos nos deter na parte arquitetônica do *software*, na interface, e verificar como ao interagir com *softwares*, mantemos uma relação de sentido. De imediato, Forouzan (2006) nos mostra que o e-mail é caracterizado por ser um programa usado para enviar mensagens eletrônicas, através de vários tipos de conexões de redes, usando também os mais variados protocolos. Dentre eles, o protocolo de rede chamado de SMTP (*Simple Mail Transfer Protocol*).

Tais mensagens eletrônicas ficam alocadas em uma caixa de correio (*mailbox*), a qual consiste em uma área de armazenamento passivo. Assim, quando alguém envia, por exemplo, uma mala direta, ela é depositada nessa caixa de armazenamento. O correio eletrônico possui uma estrutura fixa, nela podemos encontrar: (a) um envelope onde fica quase sempre o endereço do remetente e do destinatário; (b) uma mensagem que é composta pelo corpo e pelo cabeçalho, a qual define o remetente, o destinatário, a data e o assunto.

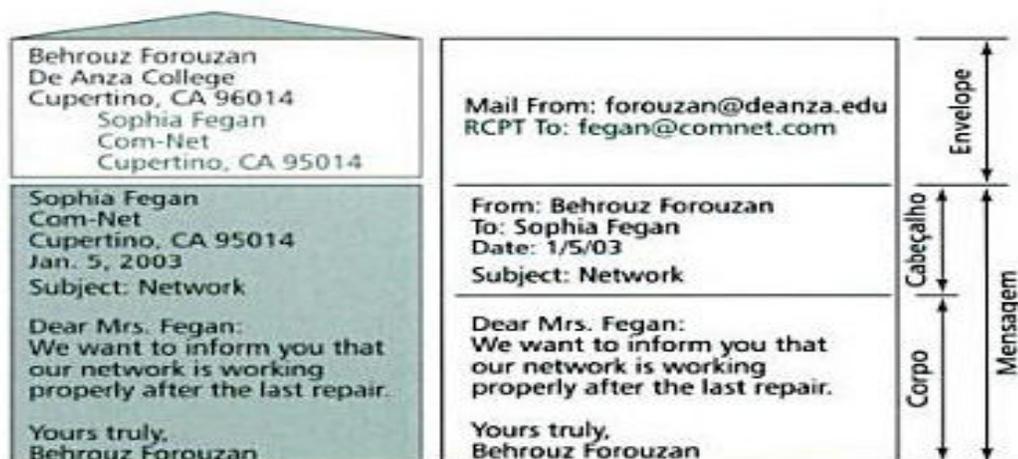


Figura 19 – formato de um e-mail

Fonte: Forouzan (2006, p. 619)

<sup>95</sup> O e-mail em questão está acoplado ao *software* aplicativo Firefox. Por está acoplado em um navegador web ele também poderia ser chamado de webmail.

Além dessa estrutura fixa no *software* de e-mail, podemos encontrar uma estrutura mecânica, a qual verifica, de forma automática e periódica, a caixa de correio. Se o usuário receber um e-mail novo, essa estrutura mecânica do *software* o destaca dos demais, usando algum símbolo de advertência. Esse e-mail novo fica junto de outros já lidos (Figura 20 a), cada um ocupando uma linha da caixa de correio (20 b; 20 c; 20 d). Geralmente, essas linhas contêm um sumário que inclui dentre outros itens, o endereço de e-mail do remetente, o assunto, bem como a hora em que o e-mail foi enviado ou recebido.

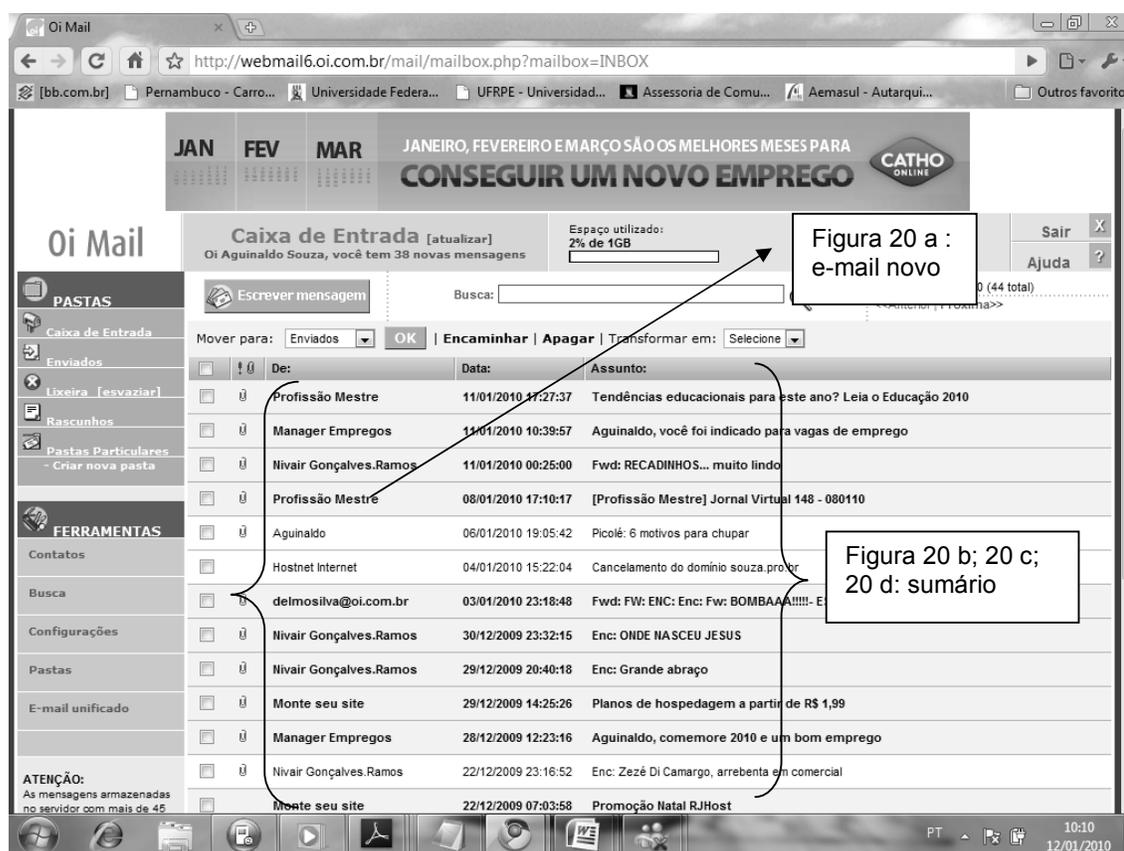
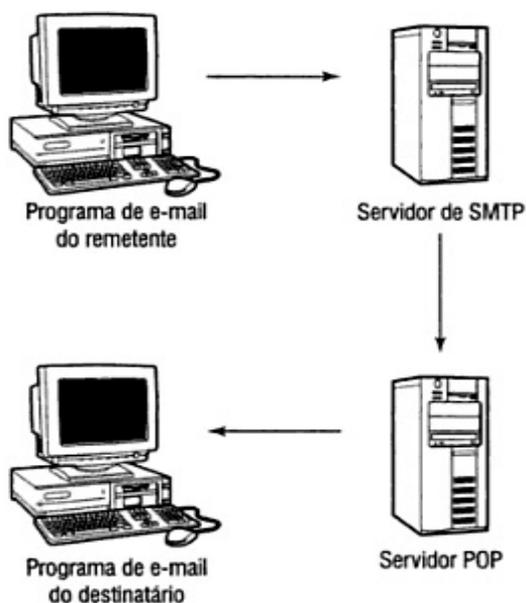


Figura 20 – estruturação de um e-mail

Para entregar um e-mail o *software* utiliza um esquema de endereçamento usado pelo SMTP, esse sistema de endereçamento é composto de duas partes: a primeira é chamada de parte local e a segunda de nome de domínio. No processo de envio de e-mail, são requeridos alguns componentes na mecânica do *software*,

tais como: um *Mail User Agent* para criar e enviar as mensagens eletrônicas. Um *Transfer Agent* que funciona como uma unidade de correio, transferindo mensagens recebidas do *User agent* para o destino apropriado. Um *Mail Delivery agent* que é um módulo que entrega mensagens à caixa de correio do usuário. Além disso, para enviar e receber e-mail o *software* cliente de e-mail necessita de um servidor SMTP e um servidor POP (*Post Office Protocol*). Conforme vemos na figura 21



**Figura 21** processo de envio de e-mail

Fonte: Scrimger et al (2002, p.369)

Entretanto, hoje já é possível acessar e-mails tanto em aplicativos instalados no computador do usuário quanto na web. Esse último é acessado através dos *softwares* aplicativos que permitem a navegação na WWW (o Firefox, por exemplo). Para que isso seja possível, é necessário além de um servidor SMTP, um protocolo chamado de IMAP (*Internet Mail Acces Protocol*). Esse protocolo permite acessar mensagens armazenadas em um servidor remoto como se estivessem armazenadas em um servidor local. (SCRIMGER et al., 2002) . A parte local do endereço de e-mail define o nome da caixa de correio, onde os e-mails dos usuários são depositados para serem recuperados pelo *user agent*.



Figura 22 User agent

Fonte: Forouzan (2006, p. 621)

Os *user agents* oferecem uma área de texto na janela do *software* onde o usuário preenche como deseja. É nessa área que vemos os mais diferentes gêneros que são recebidos no *software* de e-mail. Ver figura 23. O *user agent* geralmente cria duas caixas no *software* de e-mail: uma de entrada (figura 23 a) e outra de saída (figura 23 b). Basicamente há dois tipos de *user agents*: um orientado a comandos e outro baseado em interface GUI (*Graphical User Interface*). Esse segundo permite que o usuário interaja com o *software* através de elementos semióticos, tais como: os ícones, as barras de menus e as janelas (FOROUZAN, 2006).

Essa estrutura, a parte mecânica, é que permite o funcionamento do *software*. E além de interagir entre si, entre os módulos do *software*, essa mecânica também interage com humanos. Essa outra dimensão da interação do *software* é mais percebida quando guiamos nossa atenção para a forma arquitetônica dele, é nela que vemos mais claramente a presença das (inter)subjetividades. Tomemos como exemplo, nesse caso, a interface do *software* cliente de webmail, nela, cada signo (figura 23 c) não-verbal se apóia em um signo verbal (palavra). Essa constatação implica que, em termos de interação, a relação que o sujeito estabelece com o signo deve ser também analisada em um nível maior, mais global. Isto significa que o tipo de contato que se estabelece entre o sujeito utilizador e a interface do *software*, bem

como a relação que se dá em ambientes digitais, são subordinados a leis sígnicas (o discurso e sua constituição, o enunciado e suas formas de apresentação, o estilo<sup>96</sup>).

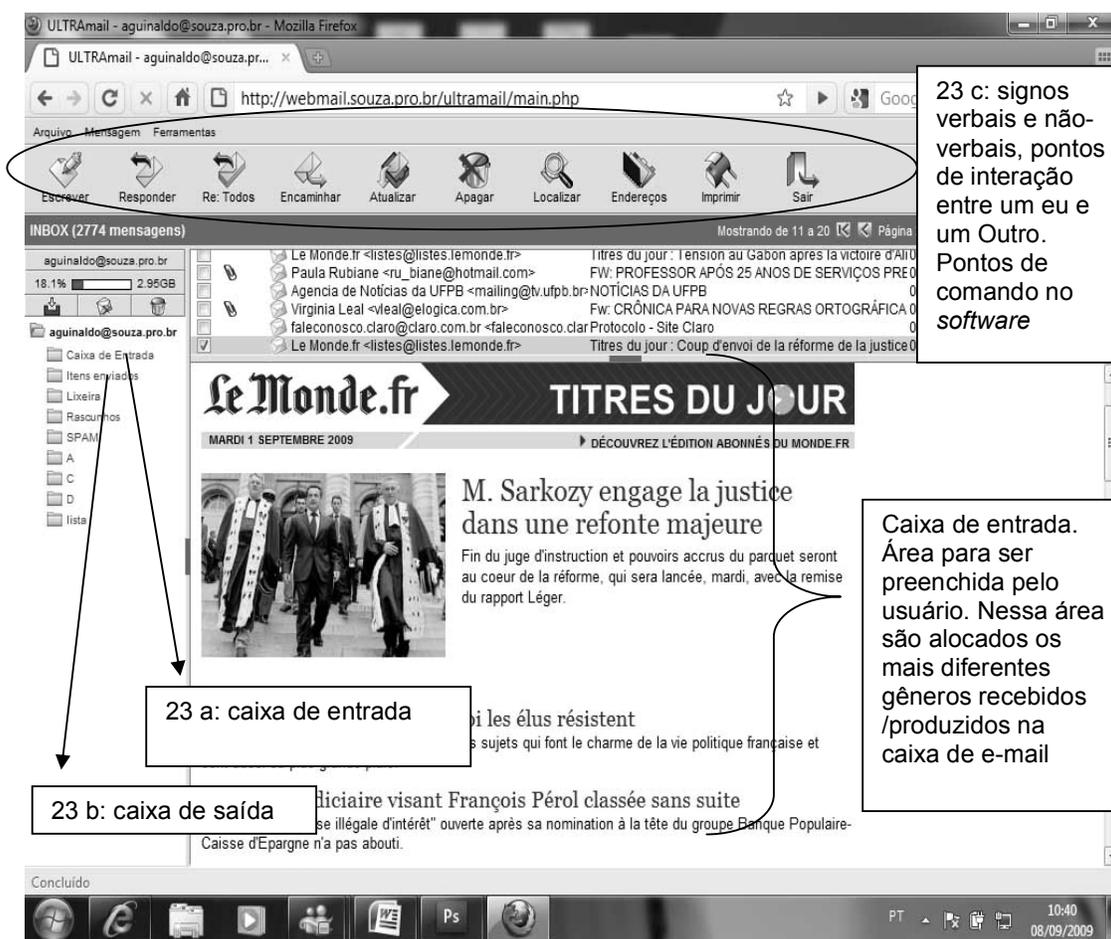


Figura 23 – área de texto

Interagimos com esses signos de forma contínua, estabelecendo, desse modo, uma relação de responsividade, tanto em relação ao produtor do *software* quanto ao parceiro enunciativo mais próximo. Uma relação que poderíamos chamar de relação de sentido, relação dialógica<sup>97</sup>. Isso se verifica, por exemplo, quando do uso de um *software* cliente de e-mail, como o que vemos na figura 24. Nele, além do

<sup>96</sup> Para essa questão ver tópico '2.6- Das marcas da subjetividade no desenvolvimento de *softwares*'

<sup>97</sup> Embora seja pré-estabelecida pelo software, essa relação gera efeitos de sentido.



signos na interface do *software* que origina esse gênero<sup>99</sup> (figura 25 (b,c,d)). Essa linha demarcará o estilo nos enunciados ali produzidos. Enquanto que o segundo (figura 25 letra b,c,d) está ligado ao produtor do *software*, o primeiro (figura 25 letra a) está diretamente relacionado com as marcas contextualizadoras deixadas ao longo de uma interação mediada

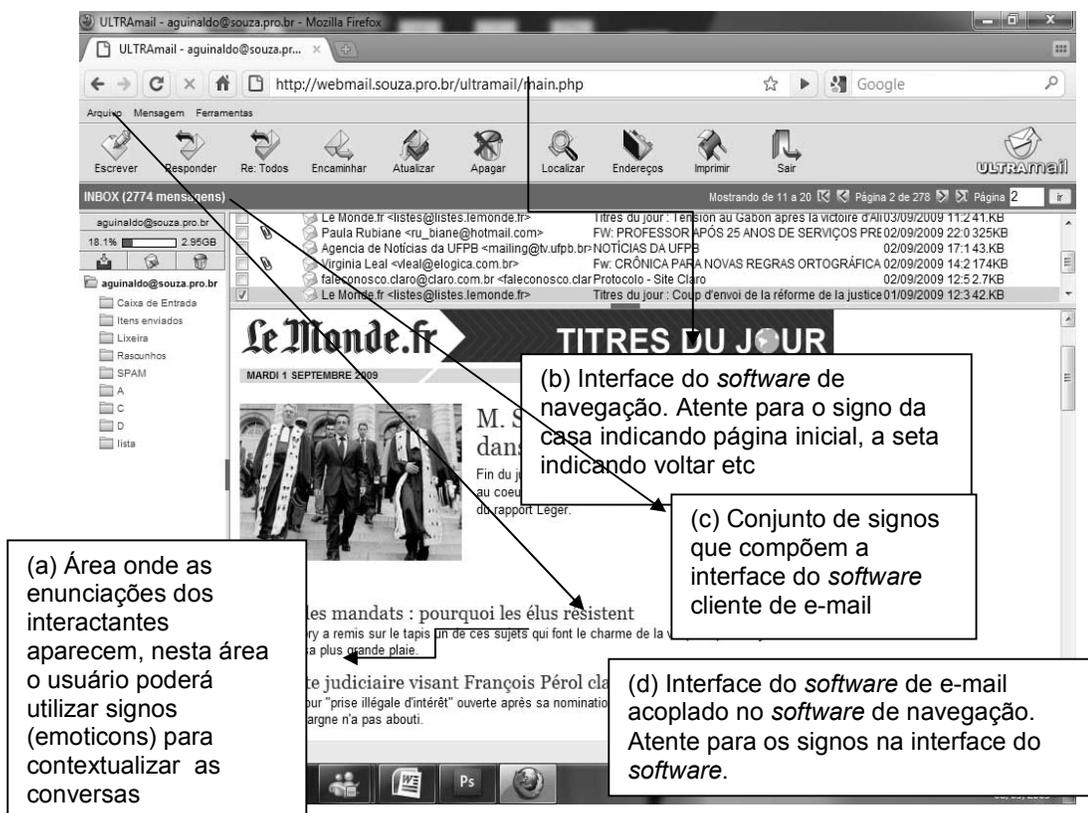


Figura 25 - *software* reconhecido como 'e-mail' acoplado ao Firefox

De igual modo, o enunciado e suas marcas estilísticas, produzidos na área do *software* reservada para que sujeitos possam enunciar, é também condicionado pelos signos – de interface. Veja nesse caso a figura 24: observe que no webmail, ali mostrado, existe uma área na qual o usuário poderá ler as enunciações de outros sujeitos situados e outra em que ele poderá escrever e enviar mensagens, etc. Todas essas enunciações, que se apresentam ou que são produzidas, são, de certa forma, condicionadas pelo produto *software* em sua forma mecânica. Forma que

<sup>99</sup> A totalidade arquitetônica

restringe o número de caracteres que poderá ser digitado e o formato e o tamanho da letra, bem como a cor da letra ou os elementos contextualizadores (*emoticons*) que poderão ser utilizados.

A parte mecânica do *software*, as linhas do código determinam tais modelos, determinam as características que um *software* pode adquirir que um texto pode adquirir. Essa parte mecânica do *software*, por sua vez, contém atributos que descrevem suas características e restrições. Nesse processo de nascituro ainda não estão presentes às vozes dos outros sociais que fabricaram o *software*. Estas são materializadas no desenvolvimento da interface.

#### **4.1 Do usuário e do *software***

A estrutura gráfica da interface do programa, que é reconhecida como webmail, levará o interactante a mobilizar diversas competências – enciclopédica, genérica etc –, com vistas à construção do sentido. O sentido não é construído só com base nos enunciados, mas pela combinação sígnica do *software*. Esses padrões sígnicos constituem, em última análise, a ‘realidade’ perceptual guiada pela práxis.

O sujeito, para se guiar por esses ambientes de interação digital, desenvolve e estabelece mecanismos de identificação/diferenciação que o permite mover-se pela gama de gêneros presentes na Internet. Apreendendo as cores, formas, funções, mensagens, enfim, signos. São justamente esses estereótipos semânticos que criam, no sujeito, ao deparar-se com uma interface, padrões perceptivos. É assim que ele consegue distinguir um programa de bate-papo de um e-mail, de um blog, etc<sup>100</sup>. Pode-se dizer que ao interagir com sistemas computacionais, a todo o instante o usuário está participando de uma ação linguística. Ao apertar um botão, ao acionar um link, ao abrir um programa de e-mail, ao digitar uma lista de compras em um processador de textos, enfim, em todos esses eventos o que o usuário faz é lidar com operações, tomar decisões, ler e interpretar num verdadeiro embate que se estabelece com o *software*.

---

<sup>100</sup> Sobre esse assunto favor retornar o tópico 1.2 deste trabalho

Essa relação que ocorre num fluxo contínuo e que passa despercebida pelo sujeito em interação, é mediada pela interface do *software*. O conceito de interface corporifica então a noção dialógica<sup>101</sup> e nos remete, de imediato, a outras vozes, às vozes de quem desenvolveu os *softwares*, que em todo momento estão em responsividade com outras vozes, conforme já apontou Peres (2007) em sua tese doctoral.

Assim, ao interagir com sistemas computacionais, o usuário está participando ativamente de uma ação linguística. Antes mesmo de interagir com outros sujeitos através de um *software*, antes mesmo de utilizar esse *software*, o usuário está imerso em uma ação mediada. Essa ação mediada ocorre a partir do momento em que o usuário entra em contato físico-visual com a interface do *software* sistema operacional e se propaga por todo o tempo em que com o sistema estiver interagindo. Desse modo, antes mesmo de utilizar um *software* aplicativo, o usuário está imerso em uma relação dialógica que se estabelece através da interface do *software* sistema operacional.

Essa interação usuário-*software*, mediada pela interface, permitirá que os interactantes desenvolvam tarefas, sejam guiados por camadas de informações, estabeleçam macrocompreensão entre estruturas. Cada uma dessas ações guia outras ações num fluxo contínuo que toma forma através de modelos mentais. Os modelos mentais são invocados através dos signos da interface. As translações dos signos em outros signos são partes de uma progressão de sentidos e se desenvolvem durante o processo de interação.

Os signos, para nós, englobam também as formas escritas da língua, os sinais gráficos, os caracteres. Na interface do *software* eles têm uma forma convencional e são utilizados como 'bússolas' para os usuários (sujeitos) poderem

---

<sup>101</sup> C.f Silveira (2002) a Engenharia Semiótica considera a interface de uma aplicação como uma mensagem do designer para o usuário representando a maneira como o designer projetou a interface e para que e por que ela foi construída. É nesse ponto que nos afastamos em parte de alguns postulados da Engenharia Semiótica e passamos a considerar que há nessas relações – entre a mensagem de quem produz o *software* e a interface - um fluxo dialógico desencadeado entre o ser que produz o *software*, entre a interface do *software* e o usuário da língua, e entre o usuário da língua, a interface do *software* e o sujeito que produz o *software*. Essas inter-relações entendemos como dialógica no sentido bakhtiniano.

se guiar por vários níveis de informações. Para evitar o excesso de estresse cognitivo o produtor do *software* procura adequar esses signos a uma forma comum compartilhada – metáforas. Com efeito, essa estratégia ajuda a construção do sentido (acarretada com a ajuda do signo). Há, portanto, toda uma organização concreta e estrutural na interface do *software*, que permite a recepção de um ambiente, gerando, assim, uma recepção sistêmica sobre os enunciados. O signo, no *software*, funciona como referente nos mecanismos de produção do significado: um referente não-verbal que condiciona o evento semântico. A nossa noção de referenciação é fabricada pela prática (interação) que mantemos com o signo no *software*. Estando este, relacionado com as coordenadas de ações que são socialmente apreendidos em situação. Para nós existe uma ligação limítrofe entre signo, sujeito e sentido.

Todo envolvimento entre o usuário e um *software* pode ser considerado como dialógico no sentido de que, ao manter com ele algum tipo de relação, está fazendo a retomada de outros processos, está em responsividade a chamados que a toda hora se manifestam na interface do *software*, seja através de signos de interface ou através de enunciações que se materializam na interface. Todos os elementos semióticos presentes na interface do *software* são conectáveis uns com outros, ou seja, se ligam de forma automática através de uma relação hipertextual. Esses ícones, símbolos e menus presentes nas interfaces dos *softwares*, são por assim dizer, pontos de interação entre um *eu* e um *outro* e poderiam ser considerados como o primeiro contato entre o usuário e o sistema computacional.

Esse evento de acionamento desencadeia a sequência de interação entre o sujeito e o *software* e se desenvolve de forma sequencial, perpassando para outras interações – não só as interações com os *softwares* aplicativos, mas também as interações com outros usuários mediadas pelos *softwares* aplicativos. As marcas de um diálogo entre o *software* e o usuário, são de certa forma, co-fabricadas por essa relação. Essa relação entre o usuário e o sistema computacional obedece a estilos de interação. Conforme Preece et al., (1994) e Shneiderman (1998), podemos resumir esses estilos de interação em: linguagem natural, linguagens de comando e menus. Além dos estilos de interação os autores destacam ainda o paradigma de interação que determina como o usuário interage com o sistema.

Os estilos e paradigmas de interação determinam o modo como o usuário se relaciona com um *software*, com uma interface ou vice-versa. As interações mediadas pela interface ajudam o usuário na hora de interagir com um sujeito situado (parceiro enunciativo mais próximo) através de *softwares* que permitem a comunicação entre dois sujeitos situados. Por exemplo, o *software* cliente de e-mail, através do *feedback* que o sistema fornece. *Feedback* visual, como mensagens na interface do *software*, auditivos como um bip ou uma volta a interface inicial de algum *software*. Tudo isso por meio de signos cuja função vai variar de acordo com o ambiente (*software*/interface) em que este estiver ligado. A representação do signo na interface do *software* (seja *software* sistema operacional ou *software* aplicativo) é de ordem metafórica.

O diálogo (homem/máquina) observado em sistemas computacionais é diferenciado, de certa maneira, pelo estilo. Assim, temos de considerar dois tipos de diálogos quando o usuário estiver interagindo com um *software*: o diálogo de interface, que utiliza de elementos coesivos para se manifestar, ou seja, utilizam a estrutura do *software* e a interface do *software* através de comandos-respostas dadas na interface do sistema, como as caixas de diálogo que são enviadas para a interface. A noção de diálogo que adotamos está muito perto da abordagem dialógica já tão bem discutida pelo filósofo russo M. Bakhtin (1997). É através dessas relações dialógicas desencadeadas em interação que o sujeito, o interactante, consegue interagir com um *software* aplicativo como, por exemplo, um comunicador instantâneo. A palavra diálogo, em Bakhtin, conforme mostra Marchezan (2005), é entendida como reação do *eu* ao *outro*, como "reação da palavra à palavra de outrem", como ponto de tensão entre o *eu* e o *outro*.

E esse ponto de tensão, quando do interagir com um *software* aplicativo, como, por exemplo, o *Messenger*, acontece através de trocas simbólicas entre o usuário e o desenvolvedor do sistema computacional, entre o usuário e outro usuário (ver figura 26). Esse ponto de tensão fundamenta e instrui a linguagem em ato. Esses diálogos, que acontecem através de enunciações presentes na interface dos *softwares*, embutidas nelas (nas interfaces) por outros sociais (*designers*, programadores etc) são o ponto de apoio sobre o qual o sujeito usuário estabelecerá relações, por exemplo, de alteridade. Ver figura 26

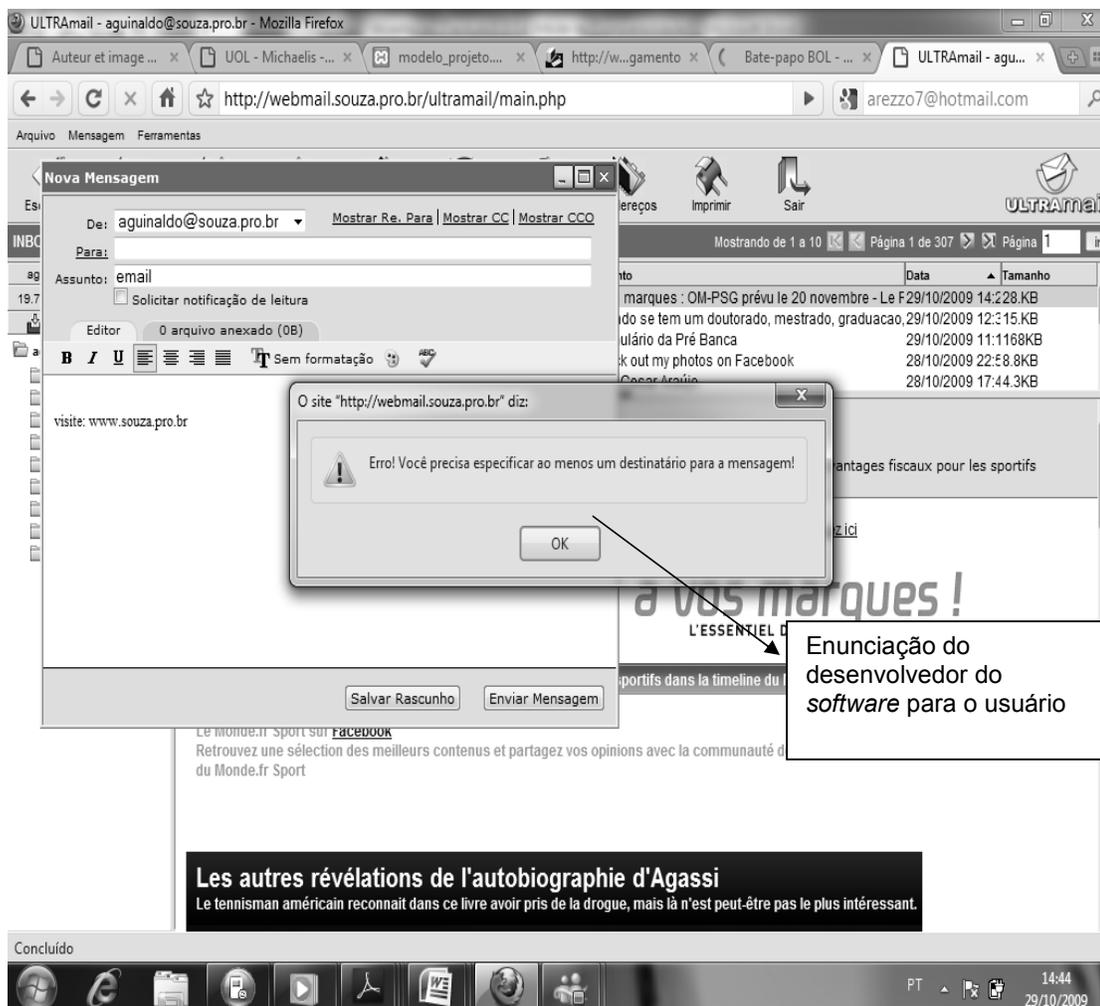


Figura 26 – troca simbólica entre desenvolvedor e usuário

## 4.2 Relações de sentido

Os processos sequenciais que engendram<sup>102</sup> um *software*, colaboram para produção, percepção e recepção para com o ato de compreensão, por exemplo, de um gênero digital, por parte do usuário. Não é a toa que, ao interagir com um *software*, o sujeito está a todo o momento produzindo sentido. As relações que se

<sup>102</sup> As funcionalidades do *software*, sua parte mecânica, os signos, as enunciações presentes em uma interface, a base hipertextual, a base hipermediática ou seja: sua parte mecânica e arquitetônica.

estabelecem entre o usuário e o *software* são relações que se desencadeiam na direção do sujeito desenvolvedor e do sujeito usuário (interação).

Dessa feita, ao interagir com um objeto, com um *software*, que é ao mesmo tempo o objeto no qual a escrita se inscreve, o objeto que dá forma aos gêneros digitais, que permite a aparição de textos que se interligam de forma não sequencial e não hierárquica (hipertexto) etc, devemos levar em consideração não só os enunciados que se materializam e são reconhecidos como gêneros<sup>103</sup>, mas também o fato de que, ao interagir com esses organismos computacionais, estamos em responsividade a outros sujeitos situados<sup>104</sup>. Essas questões que se vinculam diretamente ao conceito de autor/autoria, de atitude responsiva, criam limites precisos sobre as enunciações dos enunciadores, modificando-as, propiciando o tom expressivo, definindo o estilo.

Quando o usuário está imerso em uma relação mediada por um artefato computacional, quando ele está interagindo com um *software* que possui uma interface gráfica baseada em janelas, ele está a todo o momento respondendo a outros sujeitos, em responsividade não só com o parceiro enunciativo mais visível, mais imediato, mas também em responsividade com os desenvolvedores dos *softwares* (SOUZA, 2009; 2009a; PERES, 2007).

Essa atitude, essa compreensão responsiva ativa – nos termos que Bakhtin/Voloshinov (1996) apontam – é recorrente durante todo o tempo em que o usuário estiver interagindo com o *software*. Assim, por exemplo, quando o sujeito usuário se depara com uma interface, ele estará em relação dialógica com enunciações de outros sujeitos situados (os desenvolvedores), assumindo perante elas uma atitude responsiva ativa.

A interface do *software* é o elo mediador entre o humano e a máquina. Ela é também a responsável pela noção de realidade que o usuário estabelece com o sistema computacional. Sendo assim, é a interface que dá forma (embora seja o *software* propriamente quem define a função) ao gênero digital. Todo *software* é, por natureza, hipertextual, sendo sempre identificado pela interface.

---

<sup>103</sup> Como, por exemplo, os e-mails

<sup>104</sup> Veja, por exemplo, os enunciados que aparecem nas interfaces dos *softwares*, enunciados que guiam o uso, que indicam como interagir com o *software*

A observação das interfaces permite, pois, ao pesquisador, analisar aspectos da subjetividade dos sujeitos que produziram a interface, envolvendo questões que remetem a categorias conceituais como autor, autoria, compreensão responsiva. Basicamente, quando passamos a considerar o *software* como um organismo que aporta em si hipertextos e gêneros digitais, como o lugar da inscrição da escrita em ambiente digital, não se pode dissociar as relações de sentido que são desencadeadas quando um sujeito situado projeta o *software* e outro sujeito situado começa a utilizá-lo. Essas relações dialógicas constituem, pois, o material sobre o qual o processo de uso de um *software* repousa.

E é considerando tais relações que tomamos as interfaces dos *softwares* não como uma metacomunicação (cf. diz a engenharia semiótica), mas como enunciados de um autor/criador do *software* (SOUZA, 2009) para, a partir daí, estabelecer, com esses enunciados, com esses signos materializados nas interfaces, relações que produzem uma compreensão responsiva ativa por parte dos sujeitos envolvidos.

Essas relações dialógicas, que são relações de sentido, são possíveis a partir do momento em que o usuário desses *softwares* passa a enxergar os signos, visualizados em um *software*, como uma marca enunciativa de um Outro sujeito situado (o desenvolvedor do *software*). É importante lembrar que os signos aos quais estamos nos referindo aqui não guardam nenhuma relação com os signos utilizados por sujeitos sociais em interação mediada por um *software*, tais como os *emoticons* que os interactantes utilizam na comunicação por *software* de e-mail.

Cada sujeito social ao interagir com um *software* - esteja esse *software* codificado em uma máquina fotográfica digital, em um computador ou até mesmo em um telefone celular - está em constante diálogo com esse *software*. Essa relação entre o *software* e o usuário existe no nível da interação e faz parte de uma progressão de construção de sentidos. Essa relação guia as outras relações que o sujeito interacional realiza com o sistema, guia as relações entre sujeitos através de interações mediadas, como por exemplo, uma conversa através de um comunicador instantâneo como o *msn messenger* ou *yahoo messenger*. Em todas essas ações, a língua é o elo motor.

É através dessas inter-relações que toda a comunicação em ambiente *on-line* se desenvolve. Assim, é impossível compreender, de fato, como os enunciados em ambientes digitais são construídos ou recebidos por interactantes em interação, sem

se atentar para o ambiente em que esses enunciados nascem. Ambientes cuja própria natureza faz parte de um todo social. Os *softwares* (aplicativos e os sistemas operacionais), os enunciados produzidos através de artefatos computacionais são partes constituintes de um continuum que nasce no momento em que um sujeito projeta um *software*, e fazem parte de um todo social cuja natureza é não-linear.

As pessoas, ao interagirem com um *software*, participam de uma semiose linguística guiada por signos linguísticos (a palavra) e semióticos (os índices, os emoticons, os ícones). Essas ações mediadas, por uma interface, interferem diretamente na recepção e no uso que os sujeitos interactantes fazem, por exemplo, de um gênero introdutório como a home page<sup>105</sup>. Esses sistemas, esses *softwares*, são fabricados para produzirem ações sociais, seja interligando pessoas, seja para produzir e dar funcionalidade a um programa em que um indivíduo comunica-se com outro.

Assim, quando interagimos com um *software*, estamos interagindo com um organismo capaz de promover ações sócio-discursivas. Quando estamos sentados em frente a um computador, ou quando estamos operando um telefone celular, ou um computador de mão, enfim, quando estamos interagindo com máquinas que possuem e processam um *software*, estamos lidando com textos e elementos sógnicos, com enunciados materializados em uma interface. Desse modo, por exemplo, ao preencher um formulário eletrônico ou ao escrever um e-mail, além de utilizar a língua escrita, o usuário está também participando de uma manifestação linguística na medida em que os enunciados e signos presentes nas interfaces dos *softwares* lhes guiam as ações. A ideia de interface invoca sempre duas entidades que não se separam de forma alguma: o usuário e a máquina.

O que um usuário poderá fazer ao enviar um e-mail ou mesmo quando está em embate com outros sociais em dispositivos de mensagens instantâneas ou poderá fazer em um blog, é de certa forma, pré-dado a ele. Ou seja, o modelo conceitual e de uso de um *software* e de uma interface é pré-fabricado. Cada elemento desse sistema informatizado se liga a diferentes elementos sendo mediado pela interface do *software*.

---

<sup>105</sup> Comungamos com Bezerra (2007) a ideia de que a home page é um gênero introdutório

O processo de mediação ocorre não só quando o sujeito interacional está diante de um *software*, por exemplo, de um webmail, está realizando alguma ação com aquele *software*. O processo de mediação ocorre desde o momento em que um usuário acessa um sistema operacional e se propaga até a WWW. Essa mediação acontece através da interface do *software*, através de diálogos presenciados na interface do *software*. Quando o usuário adentra a WWW, esse processo de mediação que ocorre através da interface do *software*, passa a ser somado a outro processo de mediação, chamado de hipermodalidade. Isso ocorre porque os *softwares*, pela própria constituição, por causa da sua mecânica, podem acoplar outros elementos e se associar as mais diferentes mídias. Essas mídias são, por natureza, hipertextuais, ou seja, são não-lineares.

## **5 CONCLUSÕES**

## 5.1 Conclusões e apontamentos futuros

Dadas as complexidades do produto que recebe os hipertextos e gêneros digitais, restringimos nosso enfoque ao que chamamos de microunidades de análise do *software*. Isso significa que tratamos apenas da estrutura que permite o aparecimento dos hipertextos e gêneros digitais, e que ignoramos o material linguístico produzido por sujeitos sociais em interação através de um *software*, ou seja, não fizemos uma análise detalhada sobre os gêneros do discurso, bem como sua temática, expressividade etc.

Entendemos por microunidades de análise no *software* os signos e enunciações que compõem sua interface, bem como as funcionalidades que são dadas pela mecânica do *software*. A análise de tais elementos foi aqui conduzida à luz dos postulados teóricos da Teoria Dialógica da Linguagem (BAKHTIN, 1997) e da Ergolinguística (SCHWARTZ, 2002 a;b e FAÏTA, 2002); tais postulados levou-nos a considerar que na interface do *software* estão materializadas formas marcadas e não marcadas da presença de outrem. As microunidades, por outro lado, quando observadas como unidades ímpares, levou-nos a perceber que o sentido construtivo de cada elemento no *software*, só pode ser estabelecido em conexão com o todo. Assim é, decompomos o *software* em microunidades para descrever o todo arquitetônico dele, disso extraímos o significado dos signos e dos enunciados que são alocados nas interfaces dos *softwares* por outrem. O significado de um signo, que é alocado na interface do *software*, não pode ser inferido de um conhecimento não-linguístico, também não pode ser tratado como algo amorfo, mas sim como algo que se constitui em uma dialética, uma dialógica.

Quando passamos a considerar o *software* um 'suporte' atípico, vimos que é necessário atentar para o lugar material<sup>106</sup> onde os enunciados se manifestam<sup>107</sup>.

---

<sup>106</sup> Utilizamos o termo material aqui para nos referirmos a qualquer tipo de substância que ocupa uma extensão, seja de comprimento, largura ou profundidade, no espaço. Adquirindo, portanto, uma determinação geométrica e quantitativa, suscetível de medida e de enunciação matemática. Conforme definição encontrada no dicionário Houaiss 2.0 de língua portuguesa.

<sup>107</sup> O enunciado, conforme Maingueneau (2002, p.80), ocupa um certo espaço material. No *software* esse espaço material reservado ao texto, ao enunciado, é limitado pelas janelas. As janelas limitam o espaço de atuação do sujeito no *software*, limitam a quantidade de caracteres que podem ser

Essa dimensão material dos enunciados foi, durante muito tempo, relegada para segundo plano. Nas palavras de Maingueneau (2002), não estávamos habituados a considerar o lugar de realização dos enunciados, dos textos. Entretanto, o suporte condiciona a própria constituição do texto, modela o gênero do discurso. Ao tratarmos da questão do *software* percebemos que ele não é apenas um portador de textos, o lugar onde os hipertextos e gêneros se aportam. Mas ele, especificamente a interface dele, imprime certo aspecto aos conteúdos (gêneros) e comanda o uso que deles poderemos fazer.

Ao tratarmos de um *software*, não é suficiente levar em conta apenas sua parte mais visível, sua interface. É necessário atentar que o *software* precisa ser visto como um todo arquitetônico, como um produto. Ao olhar o *software* como um produto, nos deparamos de imediato como uma problemática que se relaciona diretamente para com a totalidade arquitetônica dele<sup>108</sup>. Essa forma do *software* é também uma forma material, isto é, uma totalidade acabada; materializa e permite o uso de escolhas de formas linguísticas que resultam em posições axiológicas as quais, por sua vez, são capazes de gerar múltiplas interrelações responsivas. Essa materialidade do *software* permite ao usuário estabelecer relação com os hipertextos e gêneros. E é nessa forma material que nasce o aspecto dinâmico, hipertextual dos enunciados produzidos sobre um *software*.

A forma arquitetônica<sup>109</sup> poderia ser entendida como ato aristotélico<sup>110</sup> e a área em que sujeitos historicamente situados interagem, poderia ser entendida como potência que corresponde ao ato. Assim, o todo arquitetônico do *software* (a relação entre ato e potência) é o que permite a formação, o aparecimento e o reconhecimento de um *software*. Essa articulação que constitui o *software* é definida para que o produto *software* possa ser visto não como uma entidade desvinculada

---

utilizadas por um indivíduo. É nessa espacialidade dada pelas janelas que podemos interagir com os *softwares* e com suas interfaces.

<sup>108</sup> Diz respeito às funcionalidades do *software*, a interface e ao próprio *software*. Arquitetônica e mecânica são termos abduzidos por nós de Bakhtin, entretanto não guardam relação direta com a definição dada pelo autor, e será usado ao longo desse trabalho com significado disforme do por ele empregado.

<sup>109</sup> Diz respeito a interface do *software*, incluindo os signos que nelas estão, os enunciados dos sujeitos desenvolvedores, os enunciados que guiam ação etc.

<sup>110</sup> Aristóteles chamou 'ato' àquilo que uma substância é, e 'potência' àquilo que uma substância pode vir a ser. Para mais informações consultar Kenny 2008 [on-line]

cujas partes funcionam automaticamente, mas como um todo integrado que forma unidade coerente. Dessa forma, o *software* e todo seu estado mecânico<sup>111</sup>, em conjunto com a totalidade arquitetônica, estão ligados aos usos que podemos fazer de um *software* e estes são visto na relação com a forma do *software*. A forma arquitetônica, além de servir para construção do sentido, organiza o espaço onde as interações mediadas ocorrem.

A totalidade mecânica do *software* é responsável pela função que o *software* executará e é através da junção de uma função e uma forma arquitetônica que um *software* que permite interações com outros sujeitos (como por exemplo, o comunicador instantâneo *Messenger*) é tomado enquanto gênero. A arquitetura é predeterminada na sua funcionalidade. A forma mecânica do *software* permite o uso racional deste, e é também a forma mecânica do *software* que permite o acoplamento de vários objetos (sons, vídeos, imagens) em um mesmo organismo. Cada uma das funções e cada uma das relações entre as funções, que acontecem quando do uso de um *software*, são projetadas pelos protagonistas da atividade. E foi direcionando nosso olhar para o modo como um *software* é fabricado que pudemos perceber que, em suas interfaces, existem enunciados concretos deixados por desenvolvedores.

É através da interface que o sujeito utilizador identificará o *software* aplicativo que estará em uso, identificará o sistema operacional; é através dela que podemos estabelecer relação com o sistema computacional e é na interface que as marcas do outro (desenvolvedores) se mostram mais visíveis, como por exemplo, os enunciados que guiam ação, os signos que compõe a interface etc.

É certo que nosso direcionamento para a interface do *software* esteve mais calcado numa relação de alteridade que se estabelece através desses ambientes. Assim é que, ao olharmos para as interfaces dos *softwares*, direcionamos nossa atenção não para uma mensagem unilateral enviada por algum protagonista da atividade ou para uma imagem do sistema projetada por alguém para outro alguém. Ao olharmos para as interfaces dos *softwares* miramos as microenunciações que a compõem, os signos que a formam, as ligações hipertextuais, ou seja, a

---

<sup>111</sup> Diz respeito ao código que permite ao *software* desempenhar determinada função

arquitetônica do *software*. É a partir desse direcionamento de olhar que passamos a ver as interfaces como ambientes contextualizadores, ambientes em que todo embate comunicativo ocorre.

Acreditamos, pois, que a interface contextualiza o ambiente para enunciação, isto é, quando de uma interação mediada, a exemplo de uma conversação no MSN *Messenger* ou de um *chat* aberto. Entretanto, faz-se necessário ainda sistematizar essa afirmação, verificar até que ponto os comandos mecânicos, os signos, as interações que o *software* estabelece com o usuário durante embate comunicativo, colabora para os êxitos das conversas mediadas entre dois sujeitos situados.

Sistematizar tais estudos é tarefa para ser feita em várias direções. É necessário atentar para as mensagens sobre os estados dos signos, sobre as funções de aplicação, sobre as interações básicas e, por fim, sobre a estrutura semântica da interface. Por sua importância aos estudos das interações mediadas por computador, essa investigação precisa ser feita de maneira sistemática, cabendo aqui o registro para futuras pesquisas: a relação governada pela interface, como aponta Johnson (1997), é uma relação semântica, caracterizada por significado e expressão. 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.

Nessa retomada do processo de fabricação de um artefato tecnológico, interessou-nos também verificar as rupturas na comunicação materializadas na interface de um *software*, os avisos de erro, as mensagens enviadas do sistema

para o sujeito usuário, para que este tome uma atitude responsiva ativa. Tudo isso surge como ações de uso que limitam o uso de um *software*.

*Software*, interface de *software*, signos de interface e funcionalidades do *software*. Essas são questões que ainda são pouco visitadas no âmbito das ciências da linguagem. Disso decorre o caminho árduo que trilhamos e as possíveis lacunas que poderão ser encontradas nesse trabalho. Entretanto foi ao decompor o *software* em microunidades (interface, funcionalidades, signos), que passamos a olhar o *software* não como um conjunto de algoritmos matemáticos, mas como um produto. Ideia discutida no primeiro capítulo desse trabalho.

Acreditamos que as reflexões que iniciamos até aqui são suficientes como um esboço daquilo que consideramos o lugar da inscrição da escrita digital. O que buscamos, em cada classe, na mecânica e nas interfaces dos *softwares*, foi apenas a precisão adequada ao assunto língua/linguagem. Assim, como “um carpinteiro e um geômetra estudam o ângulo reto de maneiras diferentes; o primeiro o faz até o ponto em que o ângulo reto é útil ao seu trabalho, enquanto o segundo indaga o que é o ângulo e como ele é<sup>112</sup>”, cumpriu-nos também olhar para os *softwares* para as interfaces dos *softwares* a fim de encontrar nelas as repostas não das perguntas que um profissional da Interação Humano-Computador ou da Ciência da Computação faz, mas sim encontrar a resposta sobre como os enunciados, hipertextos, como os gêneros digitais são formados. Para isso fomos obrigados a travar diálogo com outras ciências, estranhas a ciência linguística, por esse motivo, muitas frestas foram deixadas ao longo desse trabalho, apesar disso, acreditamos que essas frestas longe de invalidarem a teorização aqui desenvolvida só colaboram para com a formação de outros olhares sobre o objeto aqui analisado, sobre o *software*.

---

<sup>112</sup> Aristóteles (1996, p. 127)

## REFERÊNCIAS

ARAÚJO, J.C. *Chat na Web: um estudo de gênero hipertextual*. Dissertação de mestrado. Universidade Federal do Ceará, UFC, Brasil. 2003

\_\_\_\_\_ *A conversa do cotidiano na Web: o estudo da transmutação em um gênero textual*. In: Mônica Magalhães Cavalcante; Mariza Angélica Paiva Brito. (Org.) *Gêneros textuais & Referenciação*. 1 ed. Fortaleza: PROTEXTO - UFC, 2004.

\_\_\_\_\_ *Os chats: uma constelação de gêneros na Internet*. Tese de Doutorado. Universidade Federal do Ceará, UFC, Brasil. 2006

\_\_\_\_\_ *"Pra tã c a galera vc tem q abreviar muito": o internetês e as novas relações com a escrita*. In: DIEB, M. (Org). *Relações e saberes na escola: os sentidos do aprender e do ensinar*. Belo Horizonte: Autêntica, p. 119-134. 2008

ARISTÓTELES. *Ética a Nicômaco*. Trad. Leonel Vallandro e Gerd Bornheim. In: *Coleção Os Pensadores: vol. IV*. São Paulo, Abril Cultural, 1996.

AUTHIER-REVUZ, Jacqueline. *Palavras incertas: as não-coincidências do dizer*. Trad. de Pfeiffer, C.R. e outros. Campinas, SP: Ed. da UNICAMP, 1998.

BAKHTIN, M. *Problemas da poética de Dostoiévski*. Forense Universitaria. Rio de Janeiro, 1981.

\_\_\_\_\_ *Marxismo e filosofia da linguagem – Problemas fundamentais do método sociológico na ciência da linguagem*. Trad. Michel Lahud e Yara Frateschi Vieira. 8 ed. São Paulo: Hucitec, 1996

\_\_\_\_\_ *Estética da criação verbal*. São Paulo: Martins Fontes, 1997.

\_\_\_\_\_ *O discurso no romance*. In: *Questões de Literatura e de Estética (A Teoria do Romance)*, SP: Hucitec, 1998.

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). Mahwah, NJ : Lawrence Erlbaum, 2003.

BRAGA, Denise Bértoli. *Hipertexto: questões de produção e de leitura*. Estudos Lingüísticos XXXIV, p. 756-761, 2005

BRAIT, Beth. *Bakhtin: conceitos-chaves*. 2 ed. São Paulo: Contexto, 2005

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

BEZERRA, B. G. *Gêneros introdutórios em livros acadêmicos*. Universidade Federal de Pernambuco, UFPE, Brasil. 2006

\_\_\_\_\_ *Do manuscrito ao texto impresso: investigando o suporte*. In: Mônica Magalhães Cavalcante; Maria Helenice Araújo Costa; Vicência Maria Freitas Jaguaribe; Valdinar Custódio Filho. (Org.). *Texto e discurso sob múltiplos olhares: gêneros e seqüências textuais*. Rio de Janeiro: Lucerna, 2007

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

CASTELLS, Manuel. *A galáxia da Internet: reflexões sobre a Internet, os negócios e a sociedade*. Trad. Maria Luiza X. de A. Borges. Rio de Janeiro: Jorge Zahar, 2003

CHARTIER, R. *A ordem dos livros: leitores, autores e bibliotecas na Europa entre os séculos XIV e XVIII*. Brasília, DF : UnB, p. 95-111. 1994.

\_\_\_\_\_ *Do código ao monitor: a trajetória do escrito*. *Estud. av.* [online]. 1994, v. 8, n. 21, pp. 185-199. ISSN 0103-4014. doi: 10.1590/S0103-40141994000200012. 1994a

\_\_\_\_\_ *Os Desafios da Escrita*. São Paulo: Editora UNESP, p. 23. 2002

\_\_\_\_\_ *O leitor, o livro e a leitura*. Salto para o Futuro, 25 de junho de 2004. Entrevista. Disponível em <[http://www.tvebrasil.com.br/salto/entrevistas/roger\\_chartier.htm](http://www.tvebrasil.com.br/salto/entrevistas/roger_chartier.htm)> acessado em 12 de dezembro de 2008

COSCARELLI, C. V. *Da leitura de hipertexto: um diálogo com Rouet et alii*. In: ARAÚJO, J. C. & BIASI-RODRIGUES, B. (Orgs.). *Interação na Internet: novas formas de usar a linguagem*. Rio de Janeiro: Lucerna, 2005a.

\_\_\_\_\_. *Entrevista: Hipertexto*. *Letra Magna: Revista Eletrônica de Divulgação Científica em Língua Portuguesa, Lingüística e Literatura*. Ano 02, n.03, 2005. Disponível em: < [http://www.letramagna.com/Magna\\_entrevista.htm](http://www.letramagna.com/Magna_entrevista.htm) > acessado em 20 de outubro de 2009

\_\_\_\_\_ *Hipertexto e subversão: um diálogo com Andrea Ramal*. FALE/UFMG. Belo Horizonte: 2003.

Disponível em: <http://bbs.metalink.com.br/~lcoscarelli/GEhptxramal.htm> . Março, 2003. <acesso em 29 de outubro de 2009>

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

DE SOUZA, C.S. “*The Semiotic Engineering of User Interface Languages*”. *International Journal of Man-Machine Studies* 39. Academic Press. pp. 753-773. 1993

\_\_\_\_\_ *The Semiotic Engineering of Human-Computer Interaction*. Cambridge, Mass. The MIT Press. 307 p. 2005

DE SOUZA, C. S., Leite, J. C., Prates, R. O., Barbosa, S. D. J. *Interação Humano-Computador: Perspectivas Cognitivas e Semióticas* In: *Anais das Jornadas de Atualização em Informática I* ed. Rio de Janeiro : Edições Entre Lugar, 1999

FAÏTA, D. *Análise das práticas languageiras 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

FARACO, C. A. *Linguagem e diálogo, as idéias lingüísticas do círculo de Bakhtin.* Ed. Criar, 2003.

FRANÇA, Maristela. *No princípio dialógico da linguagem, o reencontro do Homo loquens com o ser humano industrial.* In: FIGUEIREDO, M.; ATHAYDE, M; BRITO, J. et al (orgs.). *Labirintos do trabalho: interrogações e olhares sobre o trabalho vivo.* Rio de Janeiro: DP&A. 2004

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

FERNANDES, Jorge Henrique Cabral. *Qual a Prática do Desenvolvimento de Software?.* Ciência e Cultura, Brasil, v. 55, n. 2, p. 29-33, 2003.

FOROUZAN, Behrouz A. *Comunicação de dados e redes de computadores.* 3. ed. Porto Alegre: Bookman, 2006.

FLYNN, Ida M. e Mchoes, Ann Mciver. *Introdução aos Sistemas Operacionais.* Tradução de Marcelo Alves Mendes, Revisor técnico Flávio Soares Corrêa da Silva. Editora: Thomson Pioneira, São Paulo 2002

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

GOMES, F. G. *Hipertextos multimodais: o percurso de apropriação de uma modalidade com fins pedagógicos.* Tese (Doutorado em Linguística Aplicada). Campinas: IEL/UNICAMP, 2007.

HOLQUIST, Michael *Dialogism: Bakhtin and his world.* London and New York, Routledge. 1990

JONSSON, Ewa. *Electronic Discourse. On Speech and Writing on the Internet*. Luleå University of Technology. Department of Communication and Languages. 1997. Disponível em: <http://www.ludd.luth.se/users/jonsson/D-essay/ElectronicDiscourse.html> . Acesso em 14 de dezembro de 2008

JOHNSON, Steven. *Cultura Da Interface: como o computador transforma nossa maneira de criar e comunicar*. Jorge Zahar Editor. Rio de Janeiro, 1ª Edição - 2001

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

LEMKE, JAY L. *Travels in hypermodality*. *Visual Communication*, Vol. 1, No. 3, 299-325. 2002

LEE, K., Lee, Y.K. y Berra, P.B. *Management of Multi-structured Hypermedia Documents: A Data Model, Query Language, and Indexing Scheme*. *Multimedia Tools and Applications* 4, pp. 199-223. 1997

LEWIS, Michael. *A nova novidade: uma história do Vale do Silício*. Tradução de Ricardo Rangel. São Paulo: Companhia das Letras. 2000

LIVINGSTON, Jessica. *Startup, fundadores da Apple, do Yahoo!, Hotmail, Firefox e Lycos contam como nasceram suas empresas milionárias*. Tradução de Marilena Reginato de Moraes Souza, Rio de Janeiro: Agir, 2009

LIMA-NETO , Vicente. *A natureza intergenérica do scrap*. In. Anais do I colóquio sobre Hipertexto. Fortaleza, UFC 2008

LIMA-NETO, V.; ARAÚJO, J. *Sobre a genericidade do scrap*. Anais do V Simpósio Internacional de Estudos de Gêneros Textuais, Caxias do Sul - RS, 2009.

LOBO-SOUSA, Ana Cristina. *Hipertextualidade: uma abordagem enunciativa de hipertextos*. Dissertação de mestrado em Linguística UFC. 2009

LOBO-SOUSA, A.C. ; ARAÚJO, J.C. ; PINHEIRO, R.C. . *Letramentos que emergem da hipertextualidade*. In: ARAÚJO, JC; DIEB, M.. (Org.). *Letramentos na web: gêneros, interação e ensino*. Fortaleza: Edições UFC, 2009, p. 111-122.

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

MARCUSCHI, Luiz Antônio. *Gêneros textuais: definição e funcionalidade*. Simpósio: gêneros textuais, mídia e ensino de língua. InPLA instituto de pesquisa aplicada São Paulo - PUC-SP. 2002

\_\_\_\_\_. *Gêneros textuais emergentes no contexto da tecnologia digital*. In: MARCUSCHI, L.A.; XAVIER , A.C. *Hipertexto e gêneros digitais: novas*

*formas de construção do sentido*. Rio de Janeiro: Lucerna, p. 13-67. 2004

\_\_\_\_\_. *Da fala para a escrita - atividades de retextualização* - 5ª edição Editora: Cortez, 2005

MARCHEZAN, Renata Coelho. *Diálogo*. In.: Bakhtin outros conceitos-chave. Beth Brait Org. São Paulo: Contexto. 2005

MAINGUENEAU, Dominique. *Análise de textos em comunicação*. 2ª ed. São Paulo: Cortez, 2002.

MEIRA, Luciano; Peres, Flavia (2004): *A dialogue-based approach for evaluating educational software*. In *Interacting with Computers*, 16 (4) pp. 615-633

MURPHY, Niall D. *Front Panel. Designing software for embedded user interfaces* R&D Books, USA, 1998

NELSON, Theodor Holm: *the Hypertext*, in: Proc. World Documentation Federation Conf. 1965

NIELSEN, J. *Usability Engineering*. San Francisco, CA: Morgan Kaufmann, 1993.

NOUROUDINE, A. 2002. *A linguagem: dispositivo revelador da complexidade do trabalho*. In: SOUZAESILVA, M. C.; FAÍTA, D. (orgs.). *Linguagem e trabalho: construção de objetos de análise no Brasil e na França*. Trad. Ines Polegatto e Décio Rocha. São Paulo: Cortez.

NORMAN, D. "*Cognitive Artifacts*". In Carroll (ed.) *Designing Interaction: Psychology at the Human-Computer Interface*, 1991.

NÖTH, Winfried. *Panorama da semiótica: de Platão a Peirce*. 3ª edição. São Paulo, Annablume, 2003

\_\_\_\_\_ *Semiótica e semiologia: os conceitos e as tradições*. *Com Ciência* 74, 2006. Disponível em:

[www.comciencia.br/comciencial?section=8&edicao=11&id=82](http://www.comciencia.br/comciencial?section=8&edicao=11&id=82). acessado em 20 de setembro de 2009

PAIVA, V.L.M.O. *E-mail: um novo gênero textual*. In: MARCUSCHI, L.A. & XAVIER, A.C. (Orgs.) *Hipertextos e gêneros digitais*. Rio de Janeiro: Lucerna, p.68-90 2004. Disponível para consulta em <http://www.veramenezes.com/emailgenero.htm> Acesso em 25/12/2009

PRANGE, A. P. L. *Da literatura aos blogs: um passeio pelo território da escrita de si*. Dissertação (mestrado em Psicologia clínica) Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro. 2002.

PEIRCE, Charles Sanders. *Semiótica*. Trad. José Teixeira Coelho Neto. 3. ed. São Paulo: Perspectiva, 2003. Tradução de: *The Collected Papers of Charles Sanders Peirce*. Charles Hartsforne e Paul Weiss, Org.1939-1914

PERES, Flávia. *Diálogo e autoria: do desenvolvimento ao uso de sistemas de informação*. 204 f, Tese de Doutorado, Curso de Pós-graduação em Psicologia Cognitiva, Universidade Federal de Pernambuco. 2007

PERES, Flávia e Meira, Luciano. *O diálogo como metáfora dos processos de desenvolvimento de software*. In.: *Anais do VII Simpósio sobre fatores humanos em sistemas computacionais*: Natal, RN. 2006

PREECE, J.; Rogers, Y.; Sharp, E.; Benyon, D.; Holland, S.; Carey, T. *Human-Computer Interaction*. Addison-Wesley. 1994

POSSENTI, S. *Notas um pouco céticas sobre hipertexto e construção de sentido*. In: *Os limites do sentido*. Curitiba: Criar. 2002

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.

PONZIO, Augusto *A revolução bakhtiniana: o pensamento de Bakhtin e a ideologia contemporânea*. Coordenação de tradução Valdemir Miotello. São Paulo, Contexto . 2008

KENNY, Anthony *As ciências teóricas de Aristóteles*. 2008 [on-line]  
Disponível em: <http://www.cfh.ufsc.br/~wfil/aristociencia.htm> acesso em 26/09/2009

KRESS, G.; Van leeuwen, T. *Reading images: the grammar of the visual design*. 2 ed. London: Routledge, 2006.

KOCH, Ingedore G. Villaça. *Desvendando os segredos do texto*. 2ª ed. São Paulo: Cortez, 2003.

KOMESU, Fabiana Cristina. *Entre o público e o privado: um jogo enunciativo na constituição do escrevente de blogs da internet / Tese de doutorado*. Campinas, SP, 2005.

KONDER, Leandro. *A questão da ideologia*. São Paulo: C.P. 2002

REZENDE, Denis Alcides. *Engenharia de software e sistemas de informação* 3.ed. Rio de Janeiro: Brasport, 2005

RIBEIRO, Ana Elisa. *Ler na tela. Novos suportes para velhas tecnologias*. 2003. 144 p. (Dissertação de mestrado em Estudos Lingüísticos) - Faculdade de Letras, Universidade Federal de Minas Gerais, Belo Horizonte, 2003.

\_\_\_\_\_ *Ler na Tela- Letramento e novos suportes de Leitura*. In: COSCARELLI, Carla Viana & RIBEIRO, Ana Elisa. *Letramento Digital Aspectos Sociais e Possibilidades pedagógicas*. BH: Autêntica Editora, .p.125-150 2005

\_\_\_\_\_ *Navegar lendo, ler navegando: Aspectos do letramento digital e da leitura de jornais*. Tese (Doutorado em Lingüística). Belo Horizonte: Programa de Pós-Graduação em Lingüística (POSLIN). Universidade Federal de Minas Gerais (UFMG), 2008

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. 2000a

\_\_\_\_\_ *Le paradigme ergologique ou um metier de philosophe*. Paris, Octares 2000b.

SCHITTINE, Denise. *Blog: comunicação e escrita íntima na internet*. Rio de Janeiro: Civilização Brasileira, 2004

SAMPAIO, M.C.H. *A Ergologia e os Estudos da Linguagem e das Práticas Linguageiras em Situações de trabalho*. Recife, n. 8, p.149-167. 2003

SAMPAIO, M. C. H. ; BARRETO, K. M. L. ; ZAPPAROLI ; AGUIAR, K. R. ; CABRAL,

SAMPAIO, M. C. H. ; BARRETO, K. M. L. ; ZAPPAROLI ; AGUIAR, K. R. ; CABRAL, I. F. F. S. ; SANTOS, M. C. V. ; Silva, Ana Paula Ribeiro da ; VALE NETO, J. P. *Linguagem e Envelhecimento: Diálogo entre Lingüistas e Gerontólogos e Aplicações em Ações Integradas de Pesquisa e Intervenção em Saúde*(TRABALHO COMPLETO). In: XIV CONGRESSO BRASILEIRO EM ERGONOMIA, 2006, CURITIBA. CONGRESSO BRASILEIRO EM ERGONOMIA, 2006a. disponível em: [http://ergocentervix.com.br/site/artigos/artigos\\_3/linguagem\\_e\\_envelhecimento.pdf](http://ergocentervix.com.br/site/artigos/artigos_3/linguagem_e_envelhecimento.pdf) <acesso em 05-03-2010>

SANTOS, M. C. V. ; Silva, Ana Paula Ribeiro da . *O método dialógico-discursivo: aplicações em estudos da memória-trabalho*(COMPLETO). In: Simpósio internacional de Métodos Qualitativos nas Ciências Sociais e na Prática Social, 2006, Recife. Anais do Simpósio internacional de Métodos Qualitativos nas Ciências Sociais e na Prática Social, 2006.

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>

SANTAELLA, Lucia. *O que é semiótica*. Editora brasiliense. São Paulo 2ª Edição - 2003

SAUSURRE. Ferdinand D. *Curso de Lingüística Geral*, Cultrix, 20a Edição, Janeiro de [1916] 1997

SIBILIA, Paula. *“Os diários íntimos na Internet e a crise da interioridade psicológica”*. 2003. Disponível em: < [http://www.antroposmoderno.com/antro-version-imprimir.php?id\\_articulo=1147](http://www.antroposmoderno.com/antro-version-imprimir.php?id_articulo=1147)>. Acesso em 29 de outubro de 2009

SILVEIRA, M.S. *Metacomunicação Designer-Usuário na Interação Humano-Computador: Design do Sistema de Ajuda*. Tese de Doutorado, Departamento de Informática. Pontifícia Universidade Católica do Rio de Janeiro. 2002

SOARES Magda *Novas práticas de leitura e escrita: letramento na cibercultura*. Educação e Sociedade: Revista de Ciência e Educação, Campinas, v.23, p. 143-160, dez. 2002.

SOBRAL, A. U. *Filosofias (e filosofia) em Bakhtin*. In: BETH BRAIT. (Org.). Bakhtin: conceitos-chave. 4 ed. São Paulo: editora contexto, v. 1, p. 123-150. 2008

SOUZA, A. G.; Carvalho, E. P. M. *Uma noção de Suporte Virtual*. In.: Hipertextus Revista Digital - Volume 1. NEHTE, UFPE Recife PE. 2006. Disponível em: <http://www.ufpe.br/nehte/revista.htm>

SOUZA, A.G.; *Gêneros Virtuais – algumas observações*. In: Letra Magna - Revista Eletrônica de Divulgação Científica em Língua Portuguesa, Lingüística e Literatura Ano 04- n.07 - 2º Semestre de 2007. Disponível em: <http://www.souza.pro.br/generos.pdf>

\_\_\_\_\_ *O signo: no gênero e no suporte virtual*. In: Letra Magna - Revista Eletrônica de Divulgação Científica em Língua Portuguesa, Lingüística e Literatura Ano 04- n.07 - 2º Semestre de 2007. Disponível em: [http://www.souza.pro.br/signos\\_suporte\\_genero.pdf](http://www.souza.pro.br/signos_suporte_genero.pdf)

\_\_\_\_\_ *Os gêneros digitais e a questão do software como suporte*. In.: Anais do II Encontro Nacional sobre Hipertexto 2007 - Universidade Federal do Ceará - Fortaleza, Brasil. 2007b

\_\_\_\_\_ *O software, lugar de inscrição da escrita em ambiente digital*. In. Anais do VI Congresso internacional da ABRALIN. UFPB João Pessoa, Paraíba. 2009

\_\_\_\_\_ *Software, hipermídia, hipertexto e gêneros digitais: observações preliminares*. Encontro internacional de texto e cultura – anais – página 42-55. CD-ROM ISSN: 2176-199X. Protexito - UFC, Fortaleza, Ceará 2009a.

SCRIMGER, Rob; Lasalle, Paul; Parihar, Mridula; Gupta, Meeta. TCP/IP a Bíblia - Tradução de Edson Furmankiewicz. Rio de Janeiro RJ, Ed. Campus, 2002

SHNEIDERMAN, B. *Designing the User Interface*, 3rd Edition. Reading, MA: Addison Wesley. 1998

TÁVORA, A.D.F. *Construção de um conceito de suporte: a matéria, a forma e a função interativa na atualização de gêneros textuais*. Tese (Doutorado em Linguística). Fortaleza: PPGL/UFC, 2008.

TAYLOR, Frederick Winslow. *Princípios de Administração Científica*. Tradução de Arlindo Vieira Ramos. Editora Atlas, São Paulo. 1976

XAVIER, A. C. *O Hipertexto na sociedade da informação: a constituição do modo de enunciação digital*. Tese de Doutorado. Campinas: Instituto de Estudos da Linguagem da Unicamp, 2002.

\_\_\_\_\_. *A dança das linguagens na Web: critérios para a definição de hipertexto*. In: SILVA, T. C.; MELLO, H. Conferências do V Congresso Internacional da Associação Brasileira de Linguística. Belo Horizonte: FALE/UFMG, p. 199-210. 2007.

# Livros Grátis

( <http://www.livrosgratis.com.br> )

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)  
[Baixar livros de Literatura de Cordel](#)  
[Baixar livros de Literatura Infantil](#)  
[Baixar livros de Matemática](#)  
[Baixar livros de Medicina](#)  
[Baixar livros de Medicina Veterinária](#)  
[Baixar livros de Meio Ambiente](#)  
[Baixar livros de Meteorologia](#)  
[Baixar Monografias e TCC](#)  
[Baixar livros Multidisciplinar](#)  
[Baixar livros de Música](#)  
[Baixar livros de Psicologia](#)  
[Baixar livros de Química](#)  
[Baixar livros de Saúde Coletiva](#)  
[Baixar livros de Serviço Social](#)  
[Baixar livros de Sociologia](#)  
[Baixar livros de Teologia](#)  
[Baixar livros de Trabalho](#)  
[Baixar livros de Turismo](#)