

# Regularidade e Generalização em Interfaces Gráficas

CLARISSE SIECKENIUS DE SOUZA  
ICAD-TeCGraf  
Departamento de Informática — PUC-Rio  
email: clarisse@icad.puc-rio.br

**Abstract** Regularities in the interface language support important cognitive processes performed by users during human-computer interaction. In particular, generalization processes play a major rôle when it comes to anticipating or recalling how to activate existing application functionalities. Textual interface languages incorporate regularities in their underlying grammar. Graphic interfaces, however, usually offer direct manipulation capabilities and do not have a grammar to describe the structure of interaction. As a result, regularities may be difficult to characterize and explore. This paper shows that regularities may be formally treated in graphic interfaces, with considerable advantages in terms of the users understanding of systems behavior.

**Keywords:** Human-Computer Interaction, Interface Design, Graphic Interfaces, Direct Manipulation, Computer Semiotics

## 1. Introdução

A gramática de uma língua é o mecanismo através do qual um conjunto infinito de sentenças pode ser gerado a partir de um número finito de regras. Associadas a ela, operações de natureza cognitiva como captura de regularidades e formulação de generalizações norteiam a comunicação e a inteligibilidade mútua entre seres humanos. É por exemplo devido à captura de uma regularidade e a uma generalização que as crianças ao aprenderem português produzem palavras como "fazi" ao invés de "fiz". É também devido a este fenômeno que um usuário comum do sistema operacional DOS<sup>®</sup> acostumado ao significado do parâmetro "/" (inclusão de subdiretórios) em comandos como BACKUP, RESTORE, REPLACE e XCOPY pode presumir, erradamente, que sua ocorrência no comando PRINT ou FORMAT tenha a ver com subdiretórios. O reconhecimento de padrões regulares e sua subsequente utilização na interpretação ou codificação de fatos novos permeia todos os processos cognitivos e constitui um fator de suma importância para determinar o sucesso ou insucesso de uma linguagem de interface.

As linguagens de comando das interfaces textuais para a comunicação entre usuários e sistemas lançam mão de uma série de regularidades — lexicais, sintáticas e semânticas — da linguagem humana, normalmente o inglês. Argumenta-se com frequência que as interfaces não-textuais gráficas contornam as especificidades lingüísticas das interfaces textuais, já que os signos gráficos apresentam menor variação de um país para outro do que as palavras e estruturas ideais de uma linguagem de comando ou programação. Contudo, há que se perguntar

como se comportam as imagens em relação a regularidades e generalizações.

Este trabalho reporta uma experiência de gramaticalização de itens de uma interface gráfica, a qual resultou na indicação de uma inconsistência entre sintaxe e semântica de imagens. Por causa dela, verifica-se uma grave contradição entre o que favorecem as generalizações feitas a partir de regularidades percebidas e aquilo que efetivamente se passa com o uso da aplicação. A análise foi realizada sobre o Desktop de um Macintosh IIsi, Sistema 7.1. Foram comparados os padrões de manipulação direta que norteiam as operações de cópia e deslocamento de objetos (folders e documentos) [Apple Computer, Inc. (1987)] à luz de dois recursos: uma adaptação do paradigma GOMS (Goals, Operators, Methods & Selection Rules) [Card, S.K.; Moran, T.P.; Newell, A. (1983)], para a modelagem de cada uma das tarefas, e as regras das TAG's (Task-Action Grammars) [Payne, S.J. e Green, T.R.G. (1986)], para a análise estrutural das ações de interface. Com base nos dados de análise, foi feito um re-projeto do Desktop para eliminar-se a inconsistência. O ambiente reprojetoado foi simulado primeiramente em Hypercard [Apple Computer, Inc. (1990)] e posteriormente em Toolbook [Asymetrix (1991)]. As simulações permitiram a realização de testes sobre compreensão e previsão de comportamento de objetos da interface.

O objetivo da investigação que aqui se apresenta não é o de re-projetar a interface de um sistema operacional. As metas visadas são essencialmente duas: (a) ilustrar o papel de regularidades e generalizações em interfaces gráficas e (b) demonstrar o potencial do ferramental analítico escolhido

para a detecção e eliminação de inconsistências de estrutura e significado neste tipo de interface. Assim, apresentam-se a seguir: na seção 2, as análises de tarefa e padrões de ação na interface, conforme as versões adaptadas dos modelos GOMS e TAG; na seção 3, um reprojeto do

Desktop, no qual a inconsistência verificada para cópia e deslocamento de objetos no sistema 7 é eliminada; na seção 4, discutem-se os fenômenos cognitivos ensejados ou bloqueados em interfaces similares; e finalmente na seção 5 traça-se a conclusão do trabalho.

Espaço de Origem	Espaço de Destino	Efeito
Disco Rígido	Disco Flexível	Cópia
Disco Flexível	Disco Rígido	Cópia
Disco Rígido	Desktop	Deslocamento
Disco Flexível	Desktop	Deslocamento
Desktop	Disco Rígido	Deslocamento ou Cópia
Desktop	Disco Flexível	Deslocamento ou Cópia

Tabela 1: Sumário dos Efeitos do Arrasto de Ícones entre Espaços Selecionados

**2. Deslocamento e Cópia de Objetos numa Interface Gráfica**

Originária do padrão Xerox STAR [Bewley, W. L.; Roberts, T. L.; Schroit, D.; Verplank, W.L. (1987)], a interface do Desktop de computadores Macintosh, Sistema 7.1,

possibilita a cópia e o deslocamento de objetos como folders e documentos através de ações diretas do mouse sobre a sua representação gráfica. Desta forma, verifica-se que, considerados três espaços em que se podem encontrar tais objetos, o arrasto (drag) dos ícones a eles correspondentes entre um e outro espaço resulta nas operações identificadas na Tabela 1.

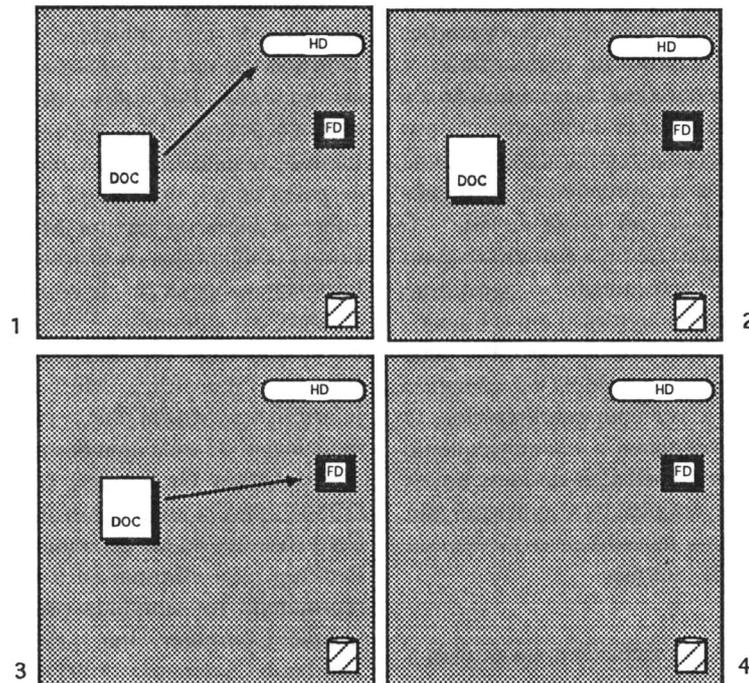


Figura 1: Marcação de Contexto para o Arrasto de Objetos no Desktop

Uma observação cuidadosa da Tabela 1 revela que existe uma ambigüidade no tocante ao que se passa quando os objetos são arrastados do Desktop para os discos. Por vezes o arrasto resulta em deslocamento, por vezes resulta em cópia. Obviamente estes efeitos não são descontrolados: estão associados ao contexto ativo no

momento em que se dá o arrasto. Quando o contexto ativo é o disco flexível e se faz um arrasto do Desktop para o disco rígido, o efeito é o de cópia. Se o contexto ativo é o disco flexível e se faz o mesmo arrasto, o efeito é deslocamento.

Embora pareça sem maior importância, o fato de a interface ser de natureza gráfica e não textual coloca na marcação visual de contexto ativo uma responsabilidade considerável: a de levar o usuário a efetuar o arrasto correto para o efeito desejado. Considerando que o efeito visual da marcação de contexto é o que se encontra esquematizado na Figura 1, percebe-se que as chances de o usuário fazer previsões equivocadas sobre o efeito do arrasto do Objeto DOC nas direções exemplificadas são significativas.

Na Figura 1, o Desktop está representado (como no Macintosh) pela janela global onde figuram certos objetos. Neste exemplo, há 4 objetos sobre o Desktop: um documento (DOC), um disco rígido (HD no canto superior

direito dos quadros), um disco flexível (FD imediatamente abaixo do disco rígido) e a lata de lixo (forma retangular rajada no canto inferior direito de cada quadro). O contexto ativo está marcado pela cor preta: trata-se do disco flexível, em todos os quadros. Vê-se que no quadro superior esquerdo da Figura 1, quando se arrasta o documento para o disco rígido, a sua imagem permanece sobre o Desktop (não está representada no quadro seguinte, ao lado, a mensagem enviada ao usuário, dando conta de que está sendo realizada uma cópia). Já no quadro inferior esquerdo, quando se arrasta o documento para o disco flexível, sua imagem desaparece de cima do Desktop (sem mensagens), o que é interpretado, corretamente, como um deslocamento.

CR 1
GOAL: copiar OBJECT: documento METHOD: arrastar do LOCAL ATUAL para LOCAL DESTINO SELECTION RULE: CONTEXTO ATIVO: disco rígido LOCAL ATUAL: disco rígido LOCAL DESTINO: disco flexível  CONTEXTO ATIVO: disco flexível LOCAL ATUAL: disco flexível LOCAL DESTINO: disco rígido  CONTEXTO ATIVO: disco rígido LOCAL ATUAL: desktop LOCAL DESTINO: disco flexível  CONTEXTO ATIVO: disco flexível LOCAL ATUAL: desktop LOCAL DESTINO: disco rígido
GOAL: deslocar OBJECT: documento METHOD: arrastar do LOCAL ATUAL para LOCAL DESTINO SELECTION RULE: CONTEXTO ATIVO: disco rígido LOCAL ATUAL: disco rígido LOCAL DESTINO: desktop  CONTEXTO ATIVO: disco flexível LOCAL ATUAL: disco flexível LOCAL DESTINO: desktop  CONTEXTO ATIVO: disco rígido LOCAL ATUAL: desktop LOCAL DESTINO: disco rígido  CONTEXTO ATIVO: disco flexível LOCAL ATUAL: desktop LOCAL DESTINO: disco flexível

**CR2**

Dicionário de Ações:  
arrastar\_objeto(TRAJETO)

Esquema de Regras:

Task(Efeito=Copiar\_objeto,TRAJETO) := arrastar\_objeto(TRAJETO).  
TRAJETO := disco\_rígido,disco\_flexível.  
TRAJETO := disco\_flexível,disco\_rígido.  
TRAJETO := desktop,disco\_rígido.  
TRAJETO := desktop,disco\_flexível.

Esquema de Regras:

Task(Efeito=Deslocar\_objeto,TRAJETO) := arrastar\_objeto(TRAJETO).  
TRAJETO := disco\_rígido,desktop.  
TRAJETO := disco\_flexível,desktop.  
TRAJETO := desktop,disco\_rígido.  
TRAJETO := desktop,disco\_flexível.

Para a modelagem das tarefas de cópia e deslocamento de objetos sobre o Desktop pode-se usar uma variação do método Goals, Operators, Methods and Selection Rules (GOMS) [Card,S.K.; Moran,T.P.; Newell,A. (1983)]. Originalmente, o GOMS é uma ferramenta destinada a modelar o processamento de informação realizado pelo cérebro humano. A meta de seus autores foi a de instrumentar o processo de design de boas interfaces de usuário em sistemas computacionais. Na variação que propomos, embora se conservem inalterados os objetivos, ao invés de se modelarem operadores (atos motores ou de processamento de informação mínimos), modelam-se os objetos envolvidos na meta principal. A motivação para esta variação é ressaltar o papel central dos objetos, sobretudo devido ao padrão de manipulação da interface, delegando-se os operadores de interface à gramática formalizada pelas TAG's. Assim, GOMS passa a conotar GOALS, OBJECTS, METHODS e SELECTION

RULES, com o resultado que aparece no Conjunto de Regras 1 (CR1).

Este modelo oferece uma métrica cognitiva bastante simples para a análise do esforço de controle do usuário sobre ações efetuadas e efeitos atingidos. Trata-se do número de parâmetros associados a cada componente do modelo. Nesta situação tem-se: 1 parâmetro associado à meta (GOAL), 1 parâmetro associado ao objeto (OBJECT), 3 parâmetros associados ao método (METHOD) — arrasto (ação), LOCAL ATUAL e LOCAL DESTINO — e 4 combinações de 3 parâmetros (12, ao todo) associados às regras de seleção (SELECTION RULES). Percebe-se uma regularidade nos parâmetros das regras de seleção, no sentido de que o sucesso das metas é determinado pela distinção ou identidade entre o CONTEXTO ATIVO e o LOCAL DESTINO. Nos casos de cópia os parâmetros têm de ser distintos; no caso de deslocamento eles têm de ser idênticos.

**CR3**

CONTEXTO\_ATIVO[Task(Efeito=Copiar\_objeto,TRAJETO) := arrastar\_objeto(TRAJETO)]  
CONTEXTO\_ATIVO:= disco\_rígido  
TRAJETO := disco\_rígido,disco\_flexível.  
TRAJETO := desktop,disco\_flexível.

CONTEXTO\_ATIVO[Task(Efeito=Copiar\_objeto,TRAJETO) := arrastar\_objeto(TRAJETO)]  
CONTEXTO\_ATIVO:= disco\_flexível  
TRAJETO := disco\_flexível,disco\_rígido.  
TRAJETO := desktop,disco\_rígido.

Em termos da linguagem de interface concretizada pelos padrões de manipulação direta do Macintosh, as Task-Action Grammars (TAG's) [Payne,S.J. e Green,T.R.G. (1986)] possibilitam uma apreciação apurada de estruturas que refletem (ou não) a regularidade do modelo GOMS,

através da gramaticalização das ações básicas a serem realizadas pelo usuário. As TAG's são um formalismo de descrição muito próximo das regras de reescritura BNF, frequentemente empregadas para especificar linguagens de programação (e.g. a linguagem Pascal [Jensen,K. and N.

Wirth (1985)). As TAG's associam diretamente o que os usuários sabem e pretendem a elementos concretos da interface. Possuem um dicionário, composto de ações básicas de interface (equivalentes a "palavras"), e um conjunto de esquemas de regras de combinação (equivalentes às regras de formação de sentenças corretas de uma língua). Extraídos dos fatos representados na Tabela 1, os esquemas de regras TAG para cópia e deslocamento são os que aparecem no Conjunto de Regras 2 (CR2).

Os esquemas de regras acima deliberadamente retratam ações livres de contexto. Nem sempre os trajetos estipulados resultarão no efeito especificado (pois dependem de contexto). Entretanto, do ponto de vista estritamente associado à ação do usuário, que é o ponto focal das TAG's, estas regras são gramaticais em pelo menos 50% dos casos (na hipótese ideal de que cada disco tem 50% de chance de ser o contexto ativo). Para que elas dêem conta de 100% dos casos, torna-se necessário sensibilizá-las a contexto, através de esquemas do tipo mostrado no Conjunto de Regras 3 (CR3).

Já que o mesmo deveria acontecer para o efeito de Deslocar\_objeto, percebe-se que o número de regras cresceu da TAG livre de contexto (10 regras ao todo) para a sensível a contexto (16 regras ao todo). Assim como se utilizou esta métrica simples no modelo GOMS, pode-se reempregá-la neste caso para mostrar que a carga cognitiva de controle de contexto nas regras gramaticais que expressam as regularidades da interface é maior do que seria

se na linguagem gráfica aparecessem elementos estruturais representativos do contexto que desfizessem a ambigüidade da TAG livre de contexto.

Em particular, a TAG livre de contexto serve para mostrar que se poderia fazer uma generalização interessante sobre as regras de efeito Deslocar\_objeto: em todas elas está envolvido o Desktop, como origem ou destino do trajeto de arrasto. O mesmo não acontece nas regras de efeito Copiar\_objeto: a tendência seria generalizar que quando o trajeto envolve os discos rígido e flexível, dá-se a cópia. Entretanto, à superfície, esta generalização é contrariada pelos trajetos que envolvem o desktop. Ela seria válida levando-se em conta os contextos, como visto no modelo GOMS: entretanto, os elementos representativos do contexto não sofrem qualquer ação de manipulação direta nas regras da TAG, o que torna mesmo a TAG sensível a contexto algo um tanto estranho (por que fazem parte das regras elementos sobre os quais não se age?).

Para finalizar esta seção de análise, é útil mostrar-se o que de fato se passa na interface do Macintosh quando se manipulam documentos que estão ou que são postos sobre o Desktop. A Figura 2 mostra a situação totalmente indesejável, mas verificada no sistema 7.1, de se ter visualmente 2 representações gráficas idênticas, que correspondem a 2 entidades físicas distintas, é verdade, mas que do ponto de vista de seu conteúdo conceitual correspondem a 1 única entidade.

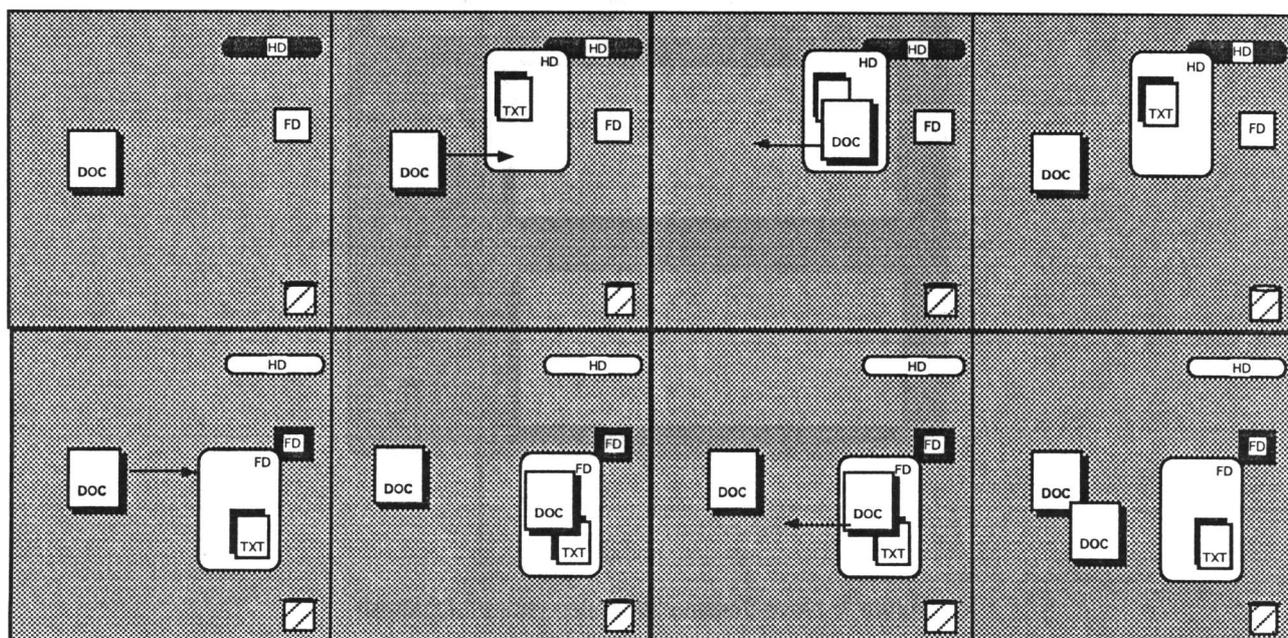


Figura 2: Setas indicadoras de deslocamentos de objetos sobre o Desktop mostram a aparente duplicação de objetos resultante da seqüência

Para exemplificar, aproveitando a seqüência da Figura 2, imagine-se que ao invés de haver no Desktop 2 ícones com o label "DOC", o label fosse "Meu CV"; ao se fazer a pergunta "O que é isto?", referindo-se a um deles, a resposta plausível seria "O meu Curriculum Vitae". Ao se fazer a mesma pergunta com referência ao outro ícone, a resposta plausível seria a mesma. Não é preciso aprofundar a demonstração dos efeitos nefastos de se ter dois objetos físicos associados a representações simbólicas idênticas: basta rememorar as dificuldades inerentes à passagem de parâmetros entre procedimentos de programa por valor, por resultado, por referência ou por nome em linguagens de programação conhecidas [Ghezzi, C. e Jazayeri, M. (1982)]. As possíveis confusões sobre que valor o parâmetro efetivamente tem sob a representação em que se vê nas linhas de programa são as mesmas do caso em estudo aqui.

### 3. Nova Representação Icônica das Operações de Deslocamento e Cópia de Objetos

A situação analisada na seção anterior motivou alterações no sistema semiótico — isto é, no sistema de símbolos utilizados para a comunicação entre agentes — do Desktop, a fim de que as regularidades existentes no modelo da tarefa fossem mais direta e inequivocamente representadas na interface. O ponto fundamental foi a marcação clara dos contextos ativos e as regularidades nas

operações em termos de sua representação em manipulação direta (a nível morfo-sintático de gramaticalização) e de seus efeitos sobre os objetos (a nível semântico de gramaticalização).

A opção adotada foi a de distinguir claramente o Desktop do disco rígido e o do disco flexível. De fato, como apontam os dados reportados anteriormente, cada disco tem seu espaço de trabalho, embora encontrem-se visualmente sobrepostos (veja-se o que ocorre na Figura 2). O sistema semiótico corrente na versão examinada desencaminha seriamente a interpretação do contexto ativo, uma vez que representações indistintas — por exemplo os dois objetos rotulados como DOC, na Figura 2 — correspondem de fato a instâncias totalmente distintas — um documento no Desktop do disco rígido e outro documento no Desktop do disco flexível. A figura 3 mostra como a subdivisão da tela em hemisférios distintos elimina esta ambigüidade.

Na reorganização do sistema semiótico do desktop, quando um disco flexível é inserido no drive, aparece associado a ele o seu espaço de trabalho — seu desktop. Este espaço poderá ser desativado e reativado através de ações da interface que não são o foco de discussão do presente trabalho. Contudo, é importante ressaltar que o momento de inserção sinaliza inequivocamente a existência de 2 possibilidades para o espaço de trabalho.

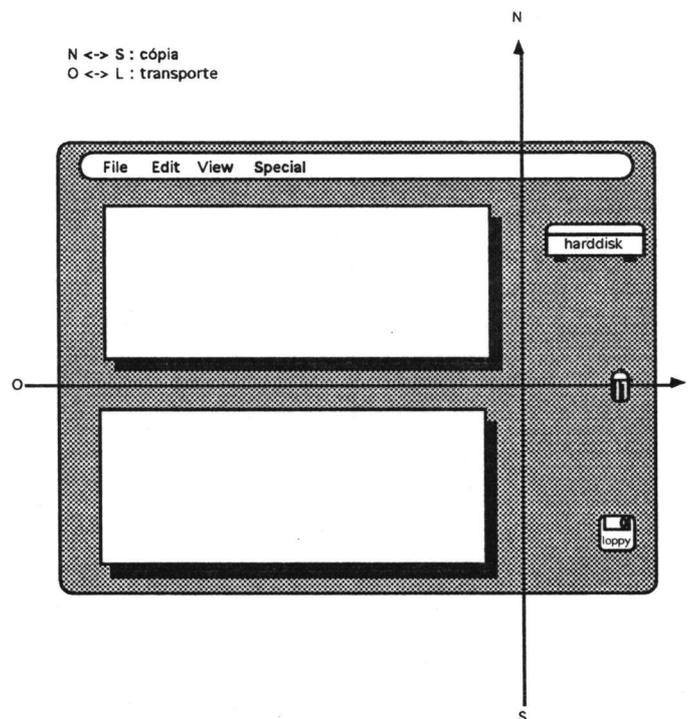


Figura 3: Novo Sistema Semiótico para o Macintosh

A tela apresenta-se dividida em hemisférios — metaforicamente o norte e o sul — oferecendo também noções longitudinais associadas ao lado direito da tela, onde estão os ícones dos discos e da lata de lixo. O posicionamento do cursor em um dos hemisférios marca o contexto ativo: se ao "norte", o contexto ativo é o do disco rígido, se ao "sul", o disco flexível. As determinações deste parâmetro sobre os efeitos do arrasto corresponderão ao fato de o local destino encontrar-se ou não no mesmo hemisfério do local atual do objeto. Uma heurística simples determinará que se um objeto é arrastado entre dois pontos do hemisfério sul (e.g. do disco flexível para o seu desktop), o efeito é o de deslocamento.

Se ele é arrastado do hemisfério sul para o norte (e.g. do disco flexível para o desktop do disco rígido), o efeito é de cópia.

Este sistema pode ser descrito por um outro conjunto de regras em uma TAG. É possível observar que, comparativamente às regras das TAG's propostas na seção 2, esta nova oferece um número bastante reduzido de regras (2 para cada operação), o que poderia na métrica cognitiva em ponto grande com que se vêm formulando as hipóteses deste trabalho corresponder a uma simplificação da linguagem de interface (ver CR4).

**CR4**

Dicionário de Ações:  
arrastar\_objeto(TRAJETO).

Esquema de Regras:  
Task(Efeito=Copiar\_objeto,TRAJETO) := arrastar\_objeto(TRAJETO).  
TRAJETO := hemisfério(i),hemisfério(j).

Esquema de Regras:  
Task(Efeito=Deslocar\_objeto,TRAJETO) := arrastar\_objeto(TRAJETO).  
TRAJETO := hemisfério(i), hemisfério(i).

Contudo, as regras acima introduzem um componente formal que não aparecera nas anteriores: a indexação de elementos. O fenômeno se deve a uma dependência de contexto herdada do modelo da tarefa (GOMS) e não parece introduzir complexidade cognitiva maior. Poder-se-ia argumentar que a indexação perturba a comparação entre as TAG's: potencialmente abstrai-se o fato de que fenômenos como este podem estar associados a processos cognitivos

complexos e que a quantidade de regras seria compensada pela ausência de indexação.

Para dar conta deste argumento, pode-se tentar eliminar a indexação, associando aos hemisférios elementos sintáticos distintos (embora conceitualmente redundantes). A situação encontrada seria a que se segue.

Espaço de Origem	Espaço de Destino	Efeito
hemisfério norte	hemisfério sul	Cópia
hemisfério sul	hemisfério norte	Cópia
hemisfério norte	hemisfério norte	Deslocamento
hemisfério sul	hemisfério sul	Deslocamento

Tabela 2: Novo Sumário dos Efeitos do Arrasto de Ícones entre Espaços Seleccionados

A comparação entre a Tabela 2 e a Tabela 1, na seção 2, já revela o desaparecimento de ambigüidades e um menor número de combinações entre as noções espaciais envolvidas. Ela encaminha a reescritura de uma TAG (ver CR5) em que a indexação não aparece, embora dela constem estes novos espaços.

A comparação entre esta última TAG e a primeira apresentada na seção 2 revela que mesmo com a eliminação da indexação (o que aumenta em 1 o número de

regras para a marcação de cada efeito), o novo sistema semiótico ainda se mostra mais simples pelo critério do número de regras necessárias para descrever as ações de interface.

#### 4. Discussão

Em trabalho anterior [Souza, C.S. (a ser publicado)], tivemos a oportunidade de apontar diretrizes de design para

**CR5**

Dicionário de Ações:arrastar\_objeto(TRAJETO)

Esquema de Regras:

Task(Efeito=Copiar\_objeto,TRAJETO) := arrastar\_objeto(TRAJETO).

TRAJETO := hemisfério norte, hemisfério sul.

TRAJETO := hemisfério sul, hemisfério norte.

Esquema de Regras:Task(Efeito=Deslocar\_objeto,TRAJETO) := arrastar\_objeto(TRAJETO).

TRAJETO := hemisfério norte, hemisfério norte.

TRAJETO := hemisfério sul, hemisfério sul.

linguagens de interface, incluídas as textuais e não-textuais. Com base na Teoria da Produção de Signos, de Umberto Eco [Eco,U. (1976)] , pudemos propor um espaço ideal para a elaboração de um sistema semiótico que vise a otimização dos processos cognitivos postos em ação por usuários de sistemas computacionais. Em particular, foi discutido que em interfaces não-textuais, como é o caso da que aqui se discute, a inexistência de um sistema lingüístico articulado que sirva de base à captura de regularidades e à formulação de generalizações é normalmente compensada pelas analogias.

A sistematicidade das noções de apagamento de objetos nas interfaces que seguem o padrão XEROX Star — expressas pela "colocação" do objeto na lata do lixo — é importada de uma analogia com a realidade do cotidiano de um escritório. A papelada inútil vai para o lixo. Halasz e Moran [Halasz,F. e Moran,T.P. (1982)] escreveram na década passada um importante trabalho acerca dos efeitos prejudiciais de certas analogias. Segundo os autores, as analogias com situações reais extra-computacionais podem levar os usuários a entender equivocadamente os conceitos e regularidades das operações de interface, na medida em que de fato a analogia se dá somente sob "certos aspectos" da ação, e não sobre todos.

Dentro do próprio sistema operacional do Macintosh, aqui exemplificado, a analogia da lata de lixo para o apagamento de objetos é observada. Contudo, a mesma ação também se aplica sobre os discos flexíveis quando se quer ejetá-los. Trata-se, claramente, de um caso em que a analogia tem limites arbitrários: se um documento ou folder é jogado no lixo, ele destina-se a ser apagado tão logo se esvazie a lata; já se um diskette é "jogado no lixo", ele é de fato ejetado do drive em que se encontra. Razões históricas levaram a linguagem de manipulação direta de imagens na tela a representar as duas operações como uma única ação de superfície [Erickson,T. (1990)].

Pode-se notar que o alcance das analogias compensa, por vezes, mas não resolve, na maioria delas, o problema das generalizações. Nas interfaces gráficas, como visto nas seções acima, a gramaticalização do sistema semiótico não é tão fácil de fazer como nas interfaces textuais (com linguagem de comando ou consulta). Entretanto, os efeitos desta gramaticalização são claramente sensíveis aos usuários.

Uma breve experiência empírica e informal com 2 sujeitos não-usuários de MacIntosh revelou que a interface com o novo sistema de desktops se desempenha melhor no que tange à manipulação direta de objetos para efeitos de cópia e deslocamento. Utilizando-se de simulações da versão original e da reprojeta, programadas em Hypercard [Apple Computer, Inc. (1990)] e em Toolbook [Asymetrix (1991)], a experiência consistiu de duas etapas: uma série de interações com o desktop reprojeta (onde aparecem os dois hemisférios) e outra série de interações com o desktop original. Em ambas as etapas, foi pedido aos sujeitos que dissessem quando um arrasto de objeto resulta em cópia e quando resulta em deslocamento.

Na primeira série, um dos sujeitos deu a resposta correta após duas interações e o outro após três interações. Na segunda série, com o modelo original do sistema 7.1, o sujeito que precisara de três interações para dar a boa resposta na primeira série precisou de cinco e o sujeito que precisara de duas desistiu de procurar a resposta. É importante acrescentar que este sujeito, até o ponto de desistir, estava interpretando a situação em que aparecem 2 objetos de mesmo nome sobre o desktop (v. Figura 2) como um efeito de cópia no arrasto do diskette para o desktop. Por um lado, esta interpretação equivocada está claramente associada à inexperiência com o MacIntosh; mas por outro, isto aponta para o fato de que com o outro design esta falta de prática não interferiu abolutamente no seu desempenho. Ao contrário, este sujeito precisou de uma interação a menos que o outro para inferir a regularidade da interface.

## 5. Conclusão

Os processos cognitivos associados à captura de regularidades e à formulação de generalizações por parte de usuários de interfaces gráficas devem constituir-se em objeto de tão cuidadosa análise quanto acontece em interfaces textuais. Contudo, a disponibilidade de um ferramental analítico abundante para sistemas lingüísticos articulados — naturais ou artificiais — não se verifica no caso dos sistemas semióticos onde os signos são essencialmente imagens e ações sobre elas.

Este trabalho partiu de uma análise teórica dos modelos de tarefa e ações de manipulação na interface de um sistema operacional do padrão Macintosh e mostrou que efeitos confusamente percebidos pelos usuários têm sua origem em uma inconsistência estrutural do sistema. A própria análise do problema, através de adaptações das ferramentas GOMS e TAG, apontou os locais em que um novo projeto de interface poderia atuar no sentido de desambiguar estruturas e tornar os processos cognitivos do usuário menos pesados.

A comparação estritamente formal das duas interfaces mostrou a qualidade comparativa de cada projeto, destacando-se a interface reprojetaada como superior à original. Dados experimentais colhidos em um breve experimento sugeriram que o resultado formal se confirma na realidade, pelos motivos expostos na seção 4, acima.

A conclusão final deste trabalho é a de que estudos teóricos acerca da sintaxe e semântica de sistemas de manipulação direta muito têm a contribuir para o design de interfaces gráficas consistentes e cognitivamente mais adequadas. Outros trabalhos por nós realizados sugerem que esta abordagem se estende também aos sistemas de visualização de processos ou estratégias de resolução de problemas [Souza, C.S. e Ortegoza da Cunha, M. (1993)].

## Agradecimentos:

Gostaria de agradecer a meus alunos dos cursos de Pós-Graduação do Departamento de Informática da PUC-Rio, nos anos de 1992 e 1993. Grande parte das idéias aqui apresentadas se devem à oportunidade que me ofereceram de aprender mais sobre este e outros assuntos.

## 6. Referências

- Apple Computer, Inc. (1987) *Human Interface Guidelines: The Apple Desktop Interface*. Reading, Ma. Addison-Wesley.
- Apple Computer, Inc. (1990) *HyperCard*
- Asymetrix (1991) *Using Toolbook*.
- Bewley, W. L.; Roberts, T. L.; Schroit, D.; Verplank, W.L. (1987) Human Factors Testing in the Xerox 8010 "Star" Office Workstation. Baecker e Buxton (eds.) *Human-Computer Interaction*. Los Altos. Morgan Kaufman. pp. 662-667
- Card, S.K.; Moran, T.P.; Newell, A. (1983) *The Psychology of Human-Computer Interaction*. Hillsdale. Lawrence Erlbaum & Associates.
- Eco, U. (1976) *A Theory of Semiotics*. Bloomington. Indiana University Press.
- Erickson, T. (1990) Creativity and Design — Introduction. in Laurel, B. (ed.) *The Art of Human-Computer Interface Design*. Menlo Park. Addison-Wesley Publishing Co. pp. 1-4
- Ghezzi, C. e Jazayeri, M. (1982) *Programming Language Concepts*. New York. John Wiley and Sons.
- Halasz, F. e Moran, T.P. (1982) Analogy Considered Harmful. in Moran (ed.) *Eight Short Papers in User Psychology*. Palo Alto. XEROX PARC. pp.33-36
- Jensen, K. and N. Wirth (1985) *Pascal User Manual and Report*. Springer-Verlag. Heidelberg.
- Payne, S.J. e Green, T.R.G. (1986) Task-Action Grammars: a Model of the Mental Representation of Task Languages. **Human-Computer Interaction** 1986, volume 2, pp. 93-133.
- Souza, C.S. (a ser publicado) The Semiotic Engineering of User Interface Languages. A ser publicado no **International Journal of Man-Machine Studies**. London. Academic Press.
- Souza, C.S. e Ortegoza da Cunha, M. (1993) *Animação de Imagens Fractais e Representação de Processos em Linguagens Visuais*. Manuscrito não-publicado.