Sobre o Ônus da Representação de Modelos em Interfaces Multimodais para Sistemas de Configuração Espacial

Raquel Oliveira Prates[1,3]
Clarisse Sieckenius de Souza[1]
Flávio Miguel Varejão[1,2]

[1]*Departamento de Informática*
Pontifícia Universidade Católica — PUC-Rio
email: {clarisse, fvarejao,raquel}@inf.puc-rio.br

[2]*Departamento de Informática*
Universidade Federal do Espírito Santo

[3]*TeCGraf - Grupo de Tecnologia em Computação Gráfica*
email: raquel@icad.puc-rio.br

PUC-RioInf.MCC26/95 June, 1995

Abstract
Layout configuration is the sort of problem that naturally requires direct manipulation interfaces, since spatial reasoning can be made easier by epistemic actions performed on visual model components of the problem in order to simplify mental computations. However, if the layout configuration PROCESS and FINAL STATE are to be stored for later recovery or re-design, strict direct manipulation seems to impose unnatural expressive constraints upon the interface. If different tasks involving static or dynamic computations in spatial reasoning are to be done, visual and textual representation seem to suit distinctively the expressive requirements of the former and the latter. Therefore, a MULTIMODAL INTERFACE stands out as the best alternative. Underneath each mode, there is a model representation. In this paper we present an exploratory investigation to find out whether a single model representation is able to meet the needs of both modes, visual and textual, of the MULTIMODAL INTERFACE.

Keywords
Interfaces, Multimodal Interfaces, Knowledge Representation, Layout Configuration

Resumo
Configuração de *layout* é o tipo de problema que normalmente requer interfaces de manipulação direta, uma vez que o raciocínio espacial é facilitado quando feito por ações epistêmicas executadas sobre componentes do modelo visual do problema, com o objetivo de simplificar as computações mentais. No entanto, se o PROCESSO e o ESTADO FINAL da configuração de *layout* devem ser armazenados para posterior recuperação ou re-configuração, o uso estrito de manipulação direta parece impor restrições de expressão pouco naturais à interface. Se diferentes tarefas de raciocínio espacial envolvendo computações dinâmicas e estáticas devem ser executadas, representações visual e textual parecem atender requerimentos de expressão distintos tanto da primeira quanto da segunda. Consequentemente, a INTERFACE MUTIMODAL desponta como a melhor alternativa. Sob cada modo, existe um modelo de representação. Neste artigo, apresentamos uma investigação exploratória para descobrir se um único modelo de representação é suficiente para atender as necessidades de ambos modos, visual e textual, de uma INTERFACE MULTIMODAL.

Palavras Chave
Interfaces, Interfaces Multimodais, Modelos de Representação, Configuração de *Layout*

1. Introdução

A configuração de *layout* é o tipo de problema que naturalmente requer interfaces de manipulação direta, visto que o raciocínio espacial pode ser facilitado por ações epistêmicas realizadas sobre componentes do modelo visual do problema, de maneira a simplificar as computações mentais [KM94]. Por ações epistêmicas entendem-se aquelas que não aproximam diretamente o estado atual da solução da tarefa de seu estado final desejado, mas aquelas que somente visam operar variações sobre o estado corrente de forma a revelar oportunidades de solução implícitas e dificilmente perceptíveis com a visualização corrente do problema (e.g. rotações 3D de objetos, mudança de ponto de vista do observador, etc.).

Contudo, se o PROCESSO de configuração do *layout* e o seu ESTADO FINAL devem ser armazenados para posterior recuperação ou *re-design*, o uso restrito da manipulação direta parece impor restrições não naturais à interface. Por exemplo, para mostrar as diversas etapas do PROCESSO de configuração, a interface somente ofereceria aos usuários uma animação das etapas sucessivas deste processo (*playback*).

Enquanto que a reprodução estática visual de um estado do *layout*, seja ele final ou não, parece atender às necessidades cognitivas do usuário numa variedade de situações, a mesma estratégia não parece ser adequada no caso de aspectos dinâmicos da tarefa. Por exemplo, uma seqüência de etapas, mostrando que um certo objeto tenha mudado de posições, mas que tenha eventualmente retornado ao seu local original, fornece os elementos necessários para que os usuários infiram que o projetista tenha considerado um conjunto de hipóteses antes de voltar a sua solução original. Contudo, a mesma seqüência de *layout*, se expressa num texto narrativo em linguagem natural (LN), poderia incluir palavras, tal como RETORNA, as quais se referem diretamente à uma abstração do padrão visual fielmente reproduzido na animação.

Em resumo, os recursos expressivos da representação visual, se comparados com a representação textual LN, parecem se adequar diferentemente a tarefas que envolvam computações estáticas ou dinâmicas no raciocínio espacial. Por esta razão, INTERFACES MULTIMODAIS têm sido propostas como a melhor alternativa disponível para os projetistas de sistemas [AHV93; Kra93].

Este artigo apresenta uma investigação exploratória realizada dentro do domínio de problemas de configuração de *layout* simples. Nosso objetivo é avaliar o impacto de INTERFACES MULTIMODAIS sobre a representação de modelos nas aplicações de *design* espacial.

Projetamos um programa simples para auxiliar os usuários a definirem o *layout* de uma sala típica de nossa universidade. As salas são normalmente compartilhadas por duas pessoas, no máximo, e seus móveis incluem mesas, computadores, estantes e armários. Fornecemos três tipos distintos de estilos de interface — manipulação direta, menus e linguagem de comandos — para que os usuários selecionassem aquele que preferissem. Fornecemos também dois estilos alternativos de saída geradas por computador — representações visuais e textuais do processo de *design* e do estado final do *layout*.

Como queríamos usar os dados com o propósito de recuperação, começamos o experimento construindo um modelo espacial do problema. A seguir, traduzimos o modelo para uma linguagem VISUAL e para uma linguagem TEXTUAL, enfatizando a habilidade de gerar texto narrativo para todo o processo de configuração. Por sua vez, o texto descritivo foi gerado a partir de um subconjunto desta gramática.

Projetamos, então, um número de testes espaciais de memória, realizados inicialmente por dois grupos de indivíduos. A cada grupo foi fornecida uma seqüência diferente de testes, de acordo com o tipo de apresentação, visual ou textual, a que eram submetidos primeiro. Como ambos tipos foram gerados a partir da mesma representação do modelo, procurávamos ter indicações sobre as seguintes questões interligadas:

1. A MESMA REPRESENTAÇÃO DO MODELO PODE DAR SUPORTE EQUIVALENTE À GERAÇÃO DE AMBOS MODOS DE INTERFACE — VISUAL E TEXTUAL — EM PROBLEMAS DE CONFIGURAÇÃO ESPACIAL?

2. DEVE HAVER REPRESENTAÇÕES DIFERENTES DO MODELO PARA DAR SUPORTE AOS MODOS ALTERNATIVOS DE INTERFACE?

A partir da análise interpretativa dos dados obtidos no experimento, pudemos extrair indicações sobre como deve ser a natureza da interface para problemas de configuração espacial de *layout*. Observamos que o emprego de uma única representação para o modelo, enquanto atende satisfatoriamente à apresentação visual na interface, não é "per se" suficiente para utilizar todo o poder expressivo da apresentação textual. Observamos, ainda, que a apresentação textual da interface se adequa melhor para a descrição de tarefas que envolvam aspectos dinâmicos do que a apresentação visual.

2. Abordagem Experimental do Problema

Para construir o modelo da sala, começamos estruturando o espaço da sala onde poderiam ser colocados os móveis em 9 regiões quadradas seqüencialmente numeradas, como demonstrado na Figura 1. A porta da sala está localizada na região 7, e a janela se estende por duas regiões adjacentes — 6 e 9. A região 5 corresponde ao centro geométrico da sala, enquanto que todas as outras são adjacentes às paredes.

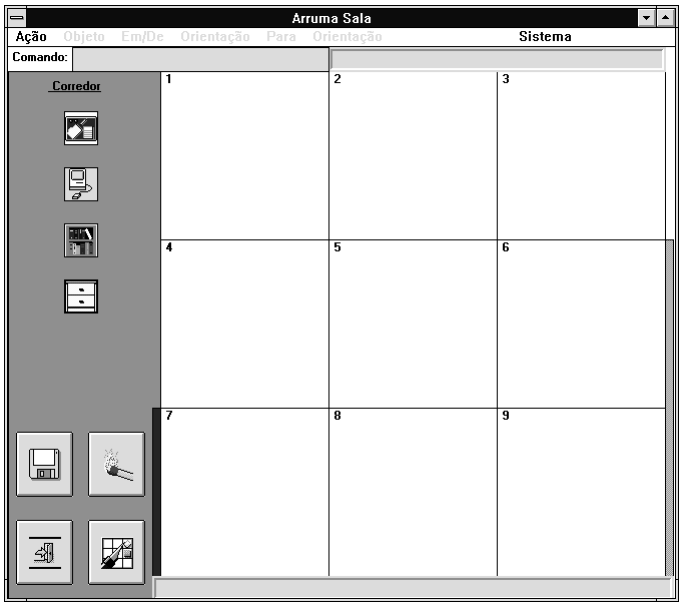


Figura 1 - Divisão do Espaço Físico de Sala em Regiões Fixas

Esta estrutura, independentemente de ser a melhor solução para o problema, foi considerada como conveniente, visto que todas as restrições sobre as possíveis localizações dos objetos podem ser facilmente estabelecidas em termos destas regiões. Por exemplo, mesas podem ser colocadas em qualquer região, exceto na 7 (em frente à porta); estantes só podem ser colocadas contra paredes (i.e. não nas regiões 5 e 6); e assim por diante. Além disso, os atributos de cada região permitem inferências sobre como localizar móveis contra paredes perpendiculares (e.g. a única parede disponível para uma estante na região 9 é a perpendicular à janela).

Os objetos são caracterizados por restrições físicas relevantes para a tarefa. Computadores só podem ser colocados em cima de mesas, armários e estantes não podem nunca ser colocados simultaneamente contra a mesma parede, e

em uma mesma região não pode existir uma mesa e um armário. Estantes, contudo, podem ser colocadas acima de mesas, contra a parede.

Apesar de haver modos de interface por menus e linguagem de comandos, o modo considerado mais adequado para o programa é a manipulação direta, na qual o espaço é representado esquematicamente em 2D, com uma visão de cima da sala e dos móveis. Os objetos são abstraídos visualmente em termos de formatos quadrilaterais equivalentes à sua superfície superior. Deste modo, computadores são representados por quadrados cuja área corresponde a aproximadamente 1/4 da área das mesas (as quais também possuem um formato quadrado, para diferenciar dos armários e das estantes). Armários e estantes, por sua vez, são representados pelo mesmo formato retangular, mas diferenciados por cores espectralmente distantes (amarelo claro e verde escuro, respectivamente).

Desta maneira, os usuários mobiliam a sala através de operações de *dragdrop* realizadas sobre os objetos disponíveis na tela (ver Figura 2). Mesmo com a disponibilidade de interações dirigidas por menus e por linguagem de comandos, os usuários rapidamente aderiram ao estilo de interação por manipulação direta. Além disso, nenhum deles demonstrou qualquer dificuldade em entender o que o programa faz e como funciona.

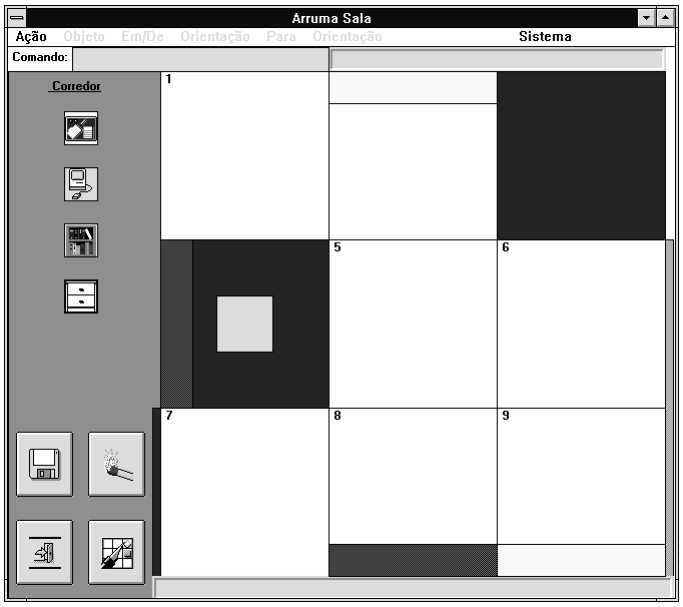


Figura 2 - Sala mobiliada com os diversos tipos de móveis disponíveis

Para os testes, as ações realizadas pelo usuário foram registradas em termos da linguagem adotada para representação do modelo. Por exemplo, se o usuário colocou uma mesa na região 1 e um computador sobre ela, a representação registrada seria, em termos gerais, assim:

PASSO 1: MOVER MESA-1 de (FORA) para (REGIÃO1)

PASSO 2: MOVER COMPUTADOR-1 de (FORA) para (REGIÃO1)

O modelo de representação é baseado em um elenco de primitivas de ação (MOVER, COLOCAR, RETIRAR), de objetos (MESA, COMPUTADOR, ESTANTE, ARMÁRIO), de trajetos articulados entre origem e destino (de - para), e de uma série de regras de restrições sobre ações. Ele pode ser visto como uma matriz onde a cada linha vêem-se as ações-objetos-trajetos, e nas colunas observa-se a seqüencialidade (aspecto temporal) dos passos do usuário.

É importante notar que o registro da história não inclui o fato de que o computador está sobre a mesa. Além disso, este tipo de informação é utilizado para avaliar localmente a ação do usuário e bloquear tentativas de se colocar

um computador sobre um armário. Por causa deste bloqueio da ação até no nível da interface, ela não é efetivada como uma etapa válida do processo de configuração e, portanto, não é representada na história da tarefa.

Outra consideração importante é a de que existe uma solução de compromisso sobre as inferências na representação da história, se as redundâncias são eliminadas — a partir de duas ações podemos escolher entre representar que a mesa e o computador estão na região 1 ou representar que a mesa está na região 1 e o computador está sobre ela. O custo da última alternativa, se comparada com a primeira, é a manutenção de referências entre objetos na história do processo. Como pode ser notado, a estratégia baseada em regiões, que foi adotada, trabalha com sistema de referência espacial de estrutura fixa, o qual fornece uma função de recuperação da história muito simples e eficaz.

Sendo assim, caso os usuários queiram visualizar todos os seus passos na solução do problema de configuração do *layout*, basta que seja fornecido, para o sistema, o arquivo com a representação da história do processo para que uma subseqüente animação mostre todas as ações realizadas. Esta característica suporta claramente a documentação da cadeia racional do processo de *design*, mas também impõe uma sobrecarga cognitiva aos usuários. Para mencionar a mais óbvia, esta estratégia de repetição visual dos passos oculta alguns conceitos abstratos que uma descrição do processo em linguagem natural poderia fornecer diretamente. Por exemplo, o uso da palavra RETORNA para sinalizar que um determinado móvel está sendo movido de volta a um lugar em que ele tinha estado anteriormente não possui equivalência na animação. O mesmo fato é verdadeiro para outros conceitos e palavras tais como TROCA e CENTRO.

Usamos a mesma representação da história do processo como entrada para a geração da animação e para um sistema de geração de texto, o qual trabalha com uma gramática de Português Brasileiro onde noções como as citadas anteriormente aparecem no texto narrativo sobre a história da solução do problema. Enquanto que com a animação a representação dos estados individuais e das sequências de estados foi satisfatória, no texto narrativo um certo número de problemas foi revelado. Dois destes problemas foram selecionados para demonstrar os custos de se ter multimodalidade em interfaces para sistemas de configuração de *layout*: o problema da referência espacial e o problema da abstração sobre um padrão de movimento.

O problema da referência espacial reside na baixa qualidade do texto gerado pelo sistema de referência espacial de estrutura fixa. Se por um lado, este sistema se mostrou vantajoso para reproduzir a história da configuração visualmente, por outro lado, ele apenas permite que se gere texto como o apresentado em (a). Claramente, seria muito mais conveniente que pudéssemos produzir um texto como o que é demonstrado em (b).

(a) "Pedro colocou a primeria mesa na região 1 e a segunda mesa na região 3. Em seguida, ele posicionou o primeiro armário na região 2."[Geração Automática]

(b) "Pedro colocou uma mesa na região 1 e outra mesa em frente a ela, contra a parede da janela. Em seguida, ele posicionou um armário entre as duas mesas." [Geração Manual]

A principal diferença entre (a) e (b) é a natureza dinâmica orientada a objetos do sistema de referências usado em (b) para a formação das expressões espaciais adequadas. Para (b) ser automaticamente gerado, o sistema original de estrutura fixa teria que ser substituído ou traduzido num sistema hierárquico e dinâmico orientado a objetos que possibilitasse que referências espaciais entre os objetos substituíssem as referências estruturais do espaço da sala.

Já o problema da abstração sobre um padrão de movimento reside na baixa qualidade do texto fornecido por uma representação plana da seqüência dos estados (como em (c) abaixo) quando comparada com o texto gerado por representações articuladas de vários padrões de mudança de estado (como em (d) abaixo). Noções como RETORNA (ou VOLTA), por exemplo, são resultantes de computações realizadas sobre toda a história da solução do problema (ou grandes porções dela), na qual a existência de dois comandos dizendo que MOVER OBJETO-X para (REGIÃO Y) de fato SIGNIFICA que o objeto retornou para uma posição prévia. Incluimos esta característica ao sistema de geração de texto e pudemos constatar o significativo custo computacional que é adicionado para

avaliar as etapas subseqüentes da história: uma base de conhecimento extra dedicada apenas ao modo de interface textual.

(c) "Pedro moveu a segunda mesa da região 9 para a região 3. [...] A seguir, ele moveu a segunda mesa da região 3 para a região 9." [Geração Automática]

(d) "Pedro moveu a segunda mesa da região 9 para a região 3. [...] A seguir, ele voltou a segunda mesa para a região 9." [Geração Automática]

Podemos concluir que, independente da qualidade ou da sofisticação da representação da história usada em nosso experimento, as apresentações visuais e textuais ideais das etapas de solução do problema não podem ser derivadas da mesma representação do modelo. Se apenas considerássemos a interface visual seria necessário um código muito simples para reproduzir com exatidão a seqüência animada equivalente à execução original. Contudo, o texto gerado diretamente a partir deste código, sem inferências espaciais adicionais, teria uma qualidade muito baixa. Por outro lado, se apenas considerássemos a interface textual, certos níveis de articulação seriam requeridos na representação do modelo (e.g. referência dinâmica orientada a objetos e abstrações sobre padrões de movimento), mas não teriam nenhum efeito perceptível sobre a sequência animada das etapas da solução do problema.

Com o intuito de medir somente a extensão de um dos problemas levantados, os efeitos de um sistema subótimo de referências de estrutura fixa em um ambiente multimodal, realizamos testes com reconstituições estáticas e dinâmicas dos estados de configuração de *layout*. Comparando os resultados obtidos por usuários com apresentação visual e textual, tentamos obter indicações sobre os impactos de uma representação inadequada do modelo sobre a interface. Procuramos respostas para as seguintes questões:

1. ALGUM DOS DOIS ESTILOS DE INTERFACE DE SAÍDA É CLARAMENTE SUPERIOR AO OUTRO PARA A APRESENTAÇÃO DOS ESTADOS DO PROBLEMA E DA ESTRATÉGIA DE SOLUÇÃO, A PONTO DE NÃO SE PRECISAR MANTER OS DOIS MODOS?

2. CASO SEJAM NECESSÁRIOS OS DOIS MODOS, HÁ UMA ESPECIALIZAÇÃO DE CADA MODO QUANTO À EXPRESSÃO PREFERENCIAL DE ASPECTOS DINÂMICOS OU ESTÁTICOS?

3. Testes Exploratórios: Descrição e Indicações

Para a realização dos testes, utilizando a aplicação geramos, primeiramente, um estado final de *layout* e, a partir dele, obtivemos uma figura colorida impressa e um texto descritivo deste estado. A seguir, produzimos uma nova configuração de *layout* e armazenamos a representação da história do processo em um arquivo. Utilizando este arquivo geramos automaticamente um texto narrativo [Pra94] e uma animação, que foram usados nos testes.

Como os testes visuais e textuais aplicados descreviam o mesmo estado final e o mesmo processo e queríamos evitar que as pessoas testadas se lembrassem da configuração do teste anterior, incluímos dois novos testes, ditos *dummy*. Estes testes consistiam de um teste visual e um outro textual que não exigiam memorização e para os quais o desempenho dos usuários não foi computado. No teste *dummy* visual o usuário deveria atingir um estado de configuração e, em seguida, transformá-lo para alcançar um estado final. No teste textual, ele deveria executar as ações descritas no texto.

O grupo de teste foi selecionado a partir de um conjunto homogêneo e especializado em computação. Dividimos o grupo de teste em dois subgrupos de quatro sujeitos, de acordo com a ordem de execução dos testes. Um grupo, chamado de TV, realizou primeiramente os testes que envolviam apresentações textuais e, posteriormente, fizeram os que envolviam apresentações visuais. Portanto, a ordem de execução dos testes do grupo TV foi:

1 - descrição textual da configuração do estado final de *layout* *

2 - descrição textual do processo de configuração de *layout* *
3 - teste *dummy* textual
4 - teste *dummy* visual
5 - descrição visual da configuração do estado final de *layout* *
6 - descrição visual do processo de configuração de *layout*

Por sua vez, o outro grupo, chamado de VT, realizou os mesmos testes, só que fazendo primeiramente os de descrições visuais e passando, em seguida, para os de descrições textuais. Portanto, a ordem de execução dos testes do grupo VT foi:

1 - descrição visual da configuração do estado final de *layout* *
2 - descrição visual do processo de configuração de *layout*
3 - teste *dummy* visual
4 - teste *dummy* textual
5 - descrição textual da configuração do estado final de *layout* *
6 - descrição textual do processo de configuração de *layout* [*]

A execução dos testes foi dividida em diferentes fases. Inicialmente, o testador explicava ao sujeito sendo testado como usar a interface. A seguir, o sujeito podia praticar com a interface até se sentir apto a usá-la. Neste momento, começavam efetivamente os testes. Para cada teste efetivo (1, 2, 5 e 6), o testador fornecia a descrição do teste, pedia que o indivíduo a memorizasse e, em seguida, recolhia a descrição pedindo que ele a reproduzisse. Ressaltamos que o indivíduo sendo testado usava o tempo que julgasse necessário para realizar a memorização e a reprodução da descrição. Por sua vez, nos testes *dummy* (3 e 4) não era requerida nenhuma memorização. Entregava-se a descrição ao indivíduo e pedia-se que ele a reproduzisse.

Nos resultados abaixo fazemos diversas comparações do desempenho dos grupos TV e VT, que apresentamos sob forma de tabelas e gráficos. Para obtermos os resultados, classificamos os erros cometidos pelos usuários em cinco categorias:

- -O/+P : mexeu no objeto errado, mas colocou-o na posição certa
- +O/-P : mexeu no objeto certo, mas colocou-o na posição errada
- -O/-P : ação sobre objeto errado em posição errada
- +A/-S : fez ação certa, mas na ordem errada
- Omissão : omissão de uma ação

Para cada um dos testes, contabilizamos os erros cometidos em cada classe. Cabe observar que os erros +A/-S são encontrados apenas nos testes narrativos, uma vez que os descritivos não têm uma seqüência definida.

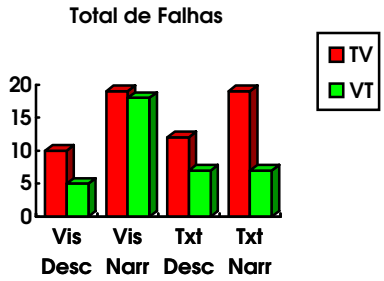
O primeiro gráfico contrasta a quantidade total de falhas que cada grupo obteve em cada um dos testes (visual descritivo, visual narrativo, textual descritivo e textual narrativo). O gráfico seguinte mostra o total de erros cometido por cada grupo em cada classe de erro. Em seguida, este total é subdividido em dois subtotais, os erros para a apresentação textual e para a apresentação visual. Os dois últimos gráficos da seção apresentam estes subtotais para os grupos TV e VT.

Tabela 1 — Desempenho Global:

[*] A descrição destes testes pode ser vista no apêndice

Grupo	Falha VisDesc	Falha VisNarr	Falha TxtDesc	Falha TxtNarr
TV	10	19	12	19
VT	5	18	7	7

Gráfico 1:



Indicações:

- O número de erros é sistematicamente maior no grupo TV que no grupo VT
- Exceto para o Teste Visual NARRATIVO, o grupo TV cometeu praticamente o dobro de erros do grupo VT
- O Teste Visual NARRATIVO foi o que apresentou maior dificuldade para o grupo VT
- O Teste NARRATIVO para o grupo TV apresentou a mesma dificuldade de execução para as apresentações textual e visual

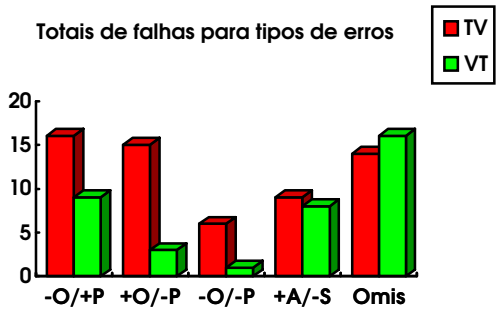
Comentários:

A narrativa visual foi uma tarefa difícil para ambos os grupos. Esta é uma indicação de que a interface de saída gráfica não é adequada para narrativa. Isto fornece evidência de que a interface visual não é suficiente, sugerindo a necessidade de multimodalidade.

Tabela 2 — Totais para tipos de erros:

Grupo	-Obj/ +Pos	+Obj/ -Pos	-Obj/ -Pos	+Ação/ -Seq	Omissão
TV	16	15	6	9	14
VT	9	3	1	8	16

Gráfico 2:



Indicações:

- Não são encontradas diferenças significativas nos erros +A/-S e Omissão cometidos pelos dois grupos. (Vale a pena ressaltar que os erros +A/-S ocorrem apenas nos Testes Narrativos, e que aproximadamente 2/3 dos erros de Omissão de cada grupo ocorrem no Teste Narrativo)

- O grupo TV cometeu um número significativamente maior de falhas (de 2 a 6 vezes maior) que o grupo VT nos problemas que envolvem apenas equívocos de objeto e posição (-O/+P, +O/-P, -O/-P)

- O tipo de erro mais freqüente do grupo VT é Omissão

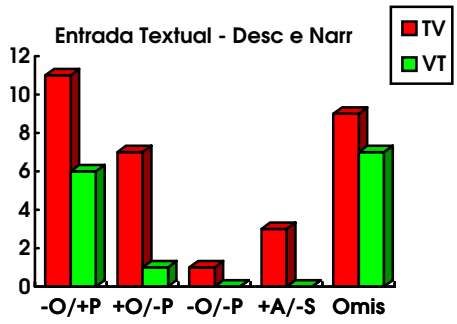
Comentários:

A Omissão foi um erro freqüente para ambos grupos e a quantidade cometida nos testes narrativos foi o dobro da quantidade cometida nos descritivos. Isto é decorrente de ser mais difícil processar uma tarefa do que uma descrição estática. Para que o usuário tenha a menor carga de memorização necessária, é preciso que a interface de saída apresente a narrativa na sua melhor forma expressiva.

Tabela 3 — Totais de erros para entrada textual:

Grupo	-Obj/ +Pos	+Obj/ -Pos	-Obj/ -Pos	+Ação/ -Seq	Omissão
TV	11	7	1	3	9
VT	6	1	0	0	7

Gráfico 3:



Indicações:

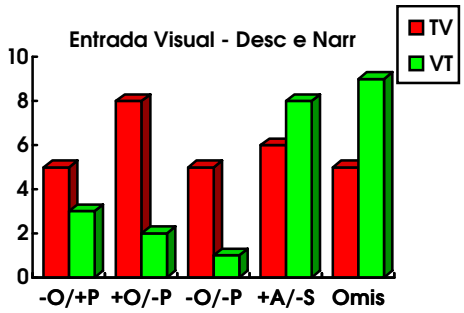
- O grupo TV cometeu, sistematicamente, maior número de erros que o grupo VT, com diferenças significativas EXCETO em -O/-P e Omissão

- Para ambos grupos os erros mais cometidos foram -O/+P e Omissão

Tabela 4 — Totais de erros para entrada visual:

Grupo	-Obj/ +Pos	+Obj/ -Pos	-Obj/ -Pos	+Ação/ -Seq	Omissão
TV	5	8	5	6	5
VT	3	2	1	8	9

Gráfico 4:



Indicações:

- Alta taxa de erros +A/-S e Omissão para os dois grupos, que são falhas comuns dos testes narrativos

- Para o grupo VT os erros +A/-S e Omissão foram, significativamente, maiores que os outros tipos

- Ambos grupos tiveram quantidades significativamente menores de erros -O/+P, na entrada visual do que na textual

- Ambos grupos tiveram quantidades significativamente maiores de erros +A/-S na entrada visual do que na textual

Comentários:

Contrastando os gráficos para a entrada visual e para a textual, observamos que com a apresentação textual os sujeitos praticamente não efetivam ações sem qualquer relação com a meta (errar objeto E posição). Aparentemente, o texto orienta melhor a ação do usuário à meta. Isto está totalmente em linha com resultados publicados sobre a influência do modo de interface na solução de problemas [Sve91].

4. Análise dos Resultados

Os resultados e as indicações da seção anterior ressaltam alguns aspectos da nossa aplicação multimodal e do seu modelo de dados.

Contrastando o desempenho global dos grupos TV e VT, observamos que o grupo VT obteve melhores resultados que o grupo TV. Comparando ainda os resultados de cada um dos grupos, nos dois primeiros e nos dois últimos testes (visual-textual e textual-visual, respectivamente) notamos uma diferença de desempenho. No teste de entrada textual, segundo teste do grupo VT e primeiro do TV, o grupo VT cometeu menos erros que o grupo TV. Por outro lado, no seu segundo teste, com a entrada visual, o grupo TV não obteve resultados superiores ao grupo VT na execução do seu primeiro teste (i.e. não parece que o grupo TV "aprendeu" com o primeiro teste narrativo, por exemplo, veja a **Tabela 1**).

Estas são indicações de que o modelo mental da aplicação construído pelo usuário a partir da representação visual pode ter sido melhor que o modelo construído a partir da representação textual. Esta constatação não é surpreendente, uma vez que se trata de uma aplicação de disposição de móveis em uma sala, que é um problema estritamente espacial.

Ao compararmos os resultados dos testes de entrada visual e textual, vimos que o erro -A/+S era significativamente maior para a entrada visual do que para a textual para ambos os grupos. Este erro ocorre apenas no teste narrativo, indicando então que a entrada textual parece mais apropriada para a descrição dinâmica de um processo que a entrada visual. Outra indicação de que as necessidades cognitivas do usuário para armazenar um processo podem estar melhor atendidas pela entrada textual do que pela visual foi observada na comparação do desempenho global. Notamos no **Gráfico 1** que o teste visual narrativo foi de dificuldade bem superior aos outros para o grupo VT. Além disso, o grupo TV teve a mesma dificuldade que o grupo VT para executá-lo.

Apesar de afirmarmos que a entrada textual parece superior à visual para o teste narrativo, o gráfico de desempenho global nos mostra uma contra-indicação. Comparando os testes narrativos de entrada visual e textual para o grupo TV, verificamos que a dificuldade de execução apresentada foi a mesma. No entanto, como já foi mostrado nas seções **1** e **2**, o texto automaticamente gerado pela aplicação e utilizado nos testes não é natural, uma vez que não suporta referências dinâmicas. Logo, a baixa qualidade da linguagem utilizada pode ter dificultado a compreensão e recuperação da informação textual pelo usuário, influenciando negativamente o seu desempenho no teste de entrada textual.

Para mostrar os efeitos negativos das deficiências da geração apresentada, imaginamos um modelo de dados específico para a representação textual e, conseqüentemente, mais apropriado para esta representação. Baseado neste modelo geramos, manualmente, textos correspondentes aos testes efetuados e realizamos novos testes.

5. Descrição do Novo Teste e Resultados Obtidos

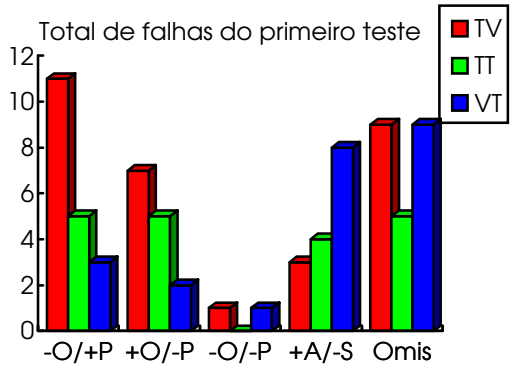
Geramos, manualmente, baseados em um modelo de representação com referências dinâmicas entre objetos, textos correspondentes aos gerados automaticamente pela aplicação e utilizados nos testes anteriores. Selecionamos um grupo de quatro pessoas para serem testadas. Pessoas estas que tinham o mesmo perfil dos participantes do teste anterior. E, uma vez selecionado o grupo, procedemos exatamente da mesma maneira como tínhamos procedido para o grupo TV. No entanto, para este novo grupo fizemos apenas o teste textual. Isso porque os resultados do teste textual eram suficientes para compararmos o desempenho dos textos com e sem referência dinâmica.

Para o novo grupo de teste (TT) contabilizamos os erros cometidos da mesma forma como tínhamos feito para os grupos TV e VT. Apresentamos a seguir os resultados em tabelas e gráficos, comparando as falhas resultantes do primeiro teste de cada grupo. Isto é, para o grupo TT e TV mostramos os erros obtidos no teste de entrada textual, enquanto que para o grupo VT mostramos os erros cometidos no teste de entrada visual. Para cada um dos resultados obtidos ressaltamos suas indicações.

Tabela 5 — Desempenho Global para a entrada textual para TT e TV e visual para VT:

Grupo	-Obj/ +Pos	+Obj/ -Pos	-Obj/ -Pos	+Ação/ -Seq	Omissão
TV	11	7	1	3	9
TT	5	5	0	4	5
VT	3	2	1	8	9

Gráfico 5:



Indicações:

- Melhoria no desempenho do grupo TT em relação ao TV
- Grupo TT cometeu menor número de erros de Omissão que os outros dois grupos
- Grupo VT cometeu menor quantidade de erros -O/+P e +O/-P, e maior quantidade de erros +A/-S que os outros dois grupos

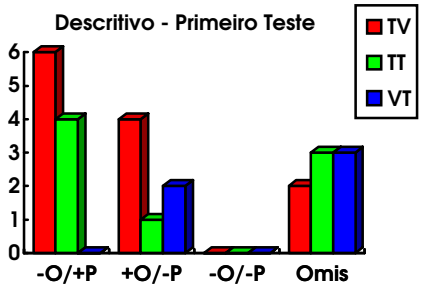
Comentários:

Comparando o desempenho dos grupos TT e TV vemos que o usuário do grupo TT cometeu menos erros envolvendo objetos e posições (-O/+P e +O/-P). Cabe ressaltar que foi exatamente as referências às posições que foram modificadas no novo modelo subjacente, passando de fixas para dinâmicas.

Tabela 6 — Teste Descritivo - Entrada textual para TT e TV e visual para VT:

Grupo	-Obj/ +Pos	+Obj/ -Pos	-Obj/ -Pos	Omissão
TV	6	4	0	2
TT	4	1	0	3
VT	0	2	0	3

Gráfico 6:



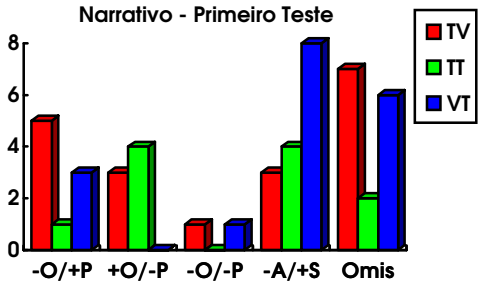
Indicações:

- Grupo TT teve melhor desempenho global que grupo TV
- Grupo VT teve melhor desempenho global que os grupos TT e TV

Tabela 7 — Teste Narrativo - Entrada textual para TT e TV e visual para VT:

Grupo	-Obj/ +Pos	+Obj/ -Pos	-Obj/ -Pos	+Ação/ -Seq	Omissão
TV	5	3	1	3	7
TT	1	4	0	4	2
VT	3	0	1	8	6

Gráfico 7:



Indicações:

- Grupo TT diminuiu significativamente erros -O/+P e Omissão em relação ao grupo TV e manteve equivalente à quantidade de erros +O/-P e +A/-S
- Grupo TT obteve resultados superiores ao grupo VT, a menos do número de erros +O/-P
- Grupo TT obteve o melhor desempenho dos três grupos nos erros -O/+P e Omissão

Comentários:

A quantidade de ações omitidas pelo grupo TT foi significativamente menor que pelos outros dois grupos. Esta é uma indicação de que o texto com referências dinâmicas é o mais facilmente memorizado pelo usuário.

Apesar do grupo TT ter utilizado o texto com referências dinâmicas, observamos uma pequena redução no seu desempenho no erro +O/-P em relação ao TV. Acreditamos que isto pode ter ocorrido devido a um possível desbalanceamento das anáforas no texto.

6. Análise dos Resultados para o Novo Teste

Os novos resultados obtidos focam principalmente o desempenho de cada um dos grupos para os testes descritivo e narrativo.

Observando o gráfico para o desempenho dos grupos no teste descritivo, vemos que o grupo VT obteve os melhores resultados. Esta é uma indicação clara de que, para a recuperação de estados estáticos da configuração de *layout*, a representação visual é melhor, ou seja, a memória visual parece imperar sobre a memória simbólica [Ioe94].

Já tínhamos verificado que para o teste narrativo a entrada textual é mais adequada. O gráfico para o teste narrativo é mais uma indicação da força desta tendência, uma vez que o grupo TT tem um desempenho superior ao do grupo VT. Comparando o resultado dos grupos TT e TV, podemos observar que o grupo TT comete sistematicamente menos erros que o grupo TV. Isto nos mostra a importância da qualidade do texto no desempenho do usuário.

É importante ressaltar que o texto gerado para o grupo TT não é, ainda, o ideal para a descrição do processo. Isto decorre do fato de que ao gerarmos o texto não nos preocupamos em balancear a quantidade de anáforas para que não fosse nem escassa, nem excessiva.

7. Conclusões

Neste trabalho pudemos observar os diversos aspectos necessários para o projeto de um *software* de configuração de *layout* que atenda ao máximo as necessidades cognitivas dos usuários.

Apresentamos uma interface multimodal baseada em uma mesma representação de modelo subjacente para a disposição de móveis em uma sala e questionamos se um dos modos, visual ou textual, seria superior ao outro como opção única de interação. Elaboramos testes para duas situações. Uma, na qual é necessário recuperar o estado final de uma configuração, e outra, na qual é preciso recuperar todo o processo para se atingir o estado final. A interface visual se mostrou melhor para a recuperação da configuração final. Além disso, ela permitiu que os usuários criassem, mentalmente, um melhor modelo de representação do espaço de trabalho.

No entanto, para a recuperação do aspecto dinâmico e seqüencial do problema (i.e. recuperação do processo), a interface textual foi superior à visual. A carga cognitiva imposta ao usuário pela interface textual é menor que a visual, uma vez que fornece de forma direta conceitos abstratos como, por exemplo, VOLTAR e TROCAR. Na interface visual, estes conceitos devem ser inferidos pelo usuário.

Ao estabelecermos a necessidade de se ter os dois modos de interface, nos perguntamos se um único modelo de dados seria suficiente para ambas representações. O modelo que adotamos satisfez inteiramente as necessidades da interface visual. Porém, o texto gerado foi pouco natural, uma vez que as referências espaciais apresentadas eram sempre fixas. Para melhorar a qualidade do texto seria necessário que referências de posição em relação a outros objetos fossem feitas. No entanto, o modelo utilizado não era adequado para a geração de referências dinâmicas, a menos de um cálculo inferencial pesado sobre ele, realizado pelo componente de geração de texto.

Alterando o modelo de referência fixa, conseguimos criar um novo modelo que atendesse à necessidade da interface textual. Não seria conveniente, porém, adotar este novo modelo como único, uma vez que ele contém dados redundantes e desnecessários para a representação visual. Por exemplo, representações explícitas das noções de TROCAR (de lugar com) e VOLTAR (a um lugar anterior) resultam em exatamente o mesmo efeito visual que uma representação muito mais simples onde a troca e o retorno estão ímplicitos.

O problema de configuração de *layout* mostrou necessitar de dois modos distintos de interface de saída, visual e textual, para melhor atender às necessidades do usuário. Por sua vez, cada um dos modos de interface requer informações diferentes para sua representação. Com isto, modelos diferenciados são necessários para cada um dos modos de representação da interface.

Portanto, não parece que a Multimodalidade seja adicionável sem ônus a uma interface gráfica única em aplicações de configuração de *layout*. O custo inferencial necessário para que o modo adicional (textual) atinja plenamente o seu potencial expressivo é o da EXPANSÃO do modelo gráfico e manutenção de ambos, na interface, ou da OPÇÃO pelo modelo EXPANDIDO como único e DEPURAÇÃO de suas redundâncias para uma representação gráfica satisfatória.

Referências

[KM94] - David Kirsch and Paul Maglio, "On Distinguishing Epistemic from Pragmatic Action". Cognitive Science 18, pp. 513-549, 1994.

[AHV93] - Yigal Arens, Eduard H. Hovy and Mira Vossers, "On the Knowledge Underlying Multimedia Presentations". In M.T. Maybury (Editor) Intelligent Multimedia Interfaces, AAAI Press, Menlo Park, pp. 280-306, 1993.

[Kra93] - Jürgens Krause, "A Multilayered Empirical Approach to Multimodality: Towards Mixed Solutions of Natural Language and Graphical Interfaces". In M.T. Maybury (Editor) Intelligent Multimedia Interfaces, AAAI Press, Menlo Park, pp. 328-354, 1993.

[Pra94] - Raquel O. Prates e André Luiz S. Clínio dos Santos, "Programa desenvolvido para Trabalho Final do Curso de Lingüística Computacional Interativa". Departamento de Informática, PUC-Rio, 1994.

[Sve91] - G.B. Svendsen, "The Influence of Interface Style on Problem Solving". International Journal of Man-Machine Studies 35, pp. 379-397, 1991.

[Ioe94] - Ioerger, "The Manipulation of Images to Handle Indeterminacy in Spatial Reasoning". Cognitive Science 18, pp. 551-593, 1994.

Apêndice - Descrições Utilizadas nos Testes

1 - Descrição Textual do Processo de Configuração de *Layout* - Geração Automática

Pedro resolveu fazer a arrumacao da sala. Foi uma tarefa muito complicada decidir a disposicao final dos moveis. Inumeras tentativas foram feitas neste sentido.

Inicialmente, Pedro colocou a primeira mesa e o primeiro computador na regiao 1. Em seguida, ele posicionou a segunda mesa na regiao 9, a primeira estante na regiao 2 e a segunda estante na regiao 8. Logo depois, colocou o primeiro armario na regiao 4.

Feito isso, Pedro moveu a segunda mesa da regiao 9 para a regiao 3. A seguir, ele voltou a segunda mesa para a regiao 9. Entao, retirou a primeira estante da regiao 2.

Finalmente, Pedro concluiu a arrumacao.

Ufa! Ate que enfim...

2 - Descrição Textual do Processo de Configuração de *Layout* - Geração Manual

Pedro resolveu fazer a arrumacao da sala. Foi uma tarefa muito complicada decidir a disposicao final dos moveis. Inumeras tentativas foram feitas neste sentido.

Inicialmente, Pedro colocou uma mesa na regiao 1, um computador sobre ela e posicionou outra mesa no canto da janela. Em seguida, pos uma estante na regiao 2 e outra na parede em frente, ao lado da segunda mesa. Logo depois, colocou um armario entre a primeira mesa e a porta.

Feito isso, Pedro moveu a segunda mesa do canto da janela para o lado da outra estante, mas em seguida voltou para a posicao anterior. Ai, retirou a estante da regiao 2.

Finalmente, Pedro concluiu a arrumacao.

Ufa! Ate que enfim...

3 - Descrição Textual da Configuração do Estado Final de *Layout* - Geração Automática

Pedro fez uma arrumacao em sua sala. Quando terminou os moveis estavam dispostos da seguinte forma:

Na regiao 2 tinha um armario. Na regiao 3, uma mesa. Na regiao 4 uma mesa, um computador e uma estante. Finalmente, tinha uma estante na regiao 8 e um armario na regiao 9.

4 - Descrição Textual da Configuração do Estado Final de *Layout* — Geração Manual

Pedro fez uma arrumacao em sua sala. Quando terminou os moveis estavam dispostos da seguinte forma:

Na regiao 2 tinha um armario e, ao lado dele, na parede da janela, uma mesa. Na regiao 4 tinha outra mesa com um computador e uma estante na parede acima dela. Finalmente, tinha outra estante na parede em frente ao armario e outro armario entre ela e a janela.

5 - Descrição Visual da Configuração do Estado Final de *Layout* - Geração Automática

Vide figura 2.

Obs.: Não são apresentadas as telas para a descrição visual do processo de configuração do *layout*