

# Ambiente para Especificação e Desenvolvimento de Interfaces Gráficas

Joaquim A. Jorge  
(jaj@inesc.pt)  
Instituto de Engenharia de Sistemas e Computadores  
Rua Alves Redol, 9, 2º D<sup>to</sup>  
Lisboa

## Resumo:

Interfaces caligráficas abrangem uma vasta classe de programas interactivos organizados em torno do desenho de diagramas e esboços através de uma caneta e computador dedicado ("pen computer"). Este tipo de interfaces tem vindo a crescer em importância com o advento de computadores de caneta e equipamento periférico que possibilitam novos modos de entrada de dados e aplicações interactivas.

Estas inovações não têm sido acompanhadas de técnicas e mecanismos formais de especificação e desenvolvimento que tirem partido das novas modalidades de interacção.

Tal deve-se em grande parte à imprecisão, indefinição e ambiguidade típicas dos dados característicos deste tipo de interfaces, que combinam as dificuldades da concepção de diálogos complexos com a necessidade de incorporar técnicas de reconhecimento de padrões em ambientes convencionais.

A presente comunicação descreve um ambiente de suporte à especificação e desenvolvimento baseado em *gramáticas relacionais de adjacências difusas* (GRADs) que combinam resultados da teoria de linguagens visuais com técnicas de reconhecimento sintáctico de padrões.

O ambiente OnN consiste num reconhecedor de traços de caneta através de representantes difusos, um gerador de aplicações a partir de GRADs e uma biblioteca de interacção gráfica sobre X11. A simplicidade, natureza modular e fracamente acoplada de qualquer destes componentes, permite ao programador máxima flexibilidade na escolha e adaptação de módulos às necessidades específicas de cada aplicação.

## Introdução:

À multiplicação dos veículos de informação em programas interactivos tem correspondido uma transição gradual de modalidades discretas de interacção através de janelas, rato e teclado para técnicas de fluxo contínuo, baseadas em reconhecimento de voz, escrita cursiva, gestos e imagens.

Se a integração destas novas modalidades promete um aumento na facilidade de uso dos programas, por outro lado o seu processamento é complicado pelo volume de dados a processar, e difícil de analisar devido a ruído, variações de utilizador para utilizador e significado por vezes ambíguo e impreciso.

Em interfaces caligráficas, relações espaciais e visuais entre componentes de expressões visuais são mais importantes que relações de sequência e concatenação, fundamentais em linguagens textuais lineares [Selker89]. Por outro lado, a complexidade dos diálogos e conteúdo rico em informação dos objectos manipulados, encorajam o recurso a descrições sintácticas para especificar o comportamento e apresentação destes programas. A nossa proposta de GRADs como veículos formais para descrição de interfaces caligráficas interactivas assenta em quatro características principais:



- Gramáticas visuais integram formas sintácticas visuais e abstractas de um modo simples e coerente, possibilitando a descrição de um grande número de interfaces com diferentes características
- Lógica difusa quantifica graus de incerteza e imprecisão nos dados de entrada, permitindo ao desenhador de interfaces lidar com informação insuficientemente clara ou mal definida.
- Sinais visuais identificam informação de entrada e saídas geradas pela aplicação, descrevendo todos os componentes do diálogo (pronto, análise sintáctica e retorno) de uma forma consistente e unificada
- Análise interactiva de expressões visuais suporta o editar, desfazer e refazer de acções, apoiando o desenvolvimento de interfaces altamente dinâmicas

A apresentação deste trabalho decorre em quatro fases. Inicialmente descreve-se a estrutura abstracta de uma interface caligráfica em termos dos seus componentes e fluxos de informação. Em seguida procede-se a uma breve descrição semi-formal dos conceitos que servem de base às gramáticas relacionais de adjacências difusas, com exemplos explicativos dos conceitos de *varrimento aleatório* (“random scan”), *relação difusa* e *restrição de adjacência* e referem-se os mecanismos de suporte à análise sintáctica interactiva.

Finalmente descreve-se a aplicação prática das diversas componentes do sistema através de uma aplicação simples. Este exemplo serve para demonstrar a descrição sintáctica de uma aplicação interactiva através de GRADs, cuja descrição textual se inspira na linguagem YACC [Johnson74].

### Arquitectura de Aplicações Caligráficas

Interfaces interactivas baseadas em caneta requerem a solução de problemas específicos: a entrada de dados exige o reconhecimento de informação imprecisa e flexibilidade na recuperação de decisões erradas do reconhecedor. As interfaces caligráficas acrescentam a estes os problema da análise sintáctica de informação altamente estruturada de uma forma não convencional.

Os exemplos de interfaces caligráficas abundam em aplicações de engenharia (SDL, esquemas de circuitos eléctricos e electrónicos, diagramas lógicos, plantas de arquitectura, desenhos de peças mecânicas), ciência de computadores (redes de Petri, diagramas de estados, fluxogramas, diagramas de fluxo de dados), e outras (mapas cartográficos, expressões matemáticas, organigramas, diagramas de instalações fabris, etc.). Esta secção discute a organização em blocos funcionais do maior múltiplo comum desta família de aplicações e descreve de forma simplificada os fluxos de informação e controle.

A Figura 2 mostra num diagrama de blocos simplificado os principais componentes de uma interface caligráfica desenvolvida segundo a nossa abordagem.

Em seguida descreve-se de forma breve o instrumento formal das Gramáticas Relacionais de Adjacência Difusa e ilustra-se as suas capacidades descritivas através de um exemplo simples.

### Gramáticas Relacionais de Adjacências Difusas

GRADs permitem descrever a sintaxe e parte da componente semântica de interfaces interactivas. Formalmente as GRADs constituem uma extensão de gramáticas independentes de contexto com atributos [Knuth68] em três direcções fundamentais:

- A ordem dos constituintes de uma produção é completamente irrelevante. p. ex. se se tem uma produção do tipo:

$$\alpha \rightarrow \{ \mu, \beta, \gamma \} \text{ if (restrições) after \{ efeitos colaterais \}}$$

então todas as 6 permutações do conjunto  $\{ \mu, \beta, \gamma \}$  são consideradas substituições válidas de  $\alpha$ .

- Restrições difusas relacionam os atributos gráficos dos símbolos constituintes e associam graus de verosimilhança a produções na gramática, permitindo quantificar a ambiguidade e incerteza associados a uma expressão visual.
- Relações de adjacência agrupam componentes sintácticos de uma produção, definindo ligações “fortes” entre objectos visuais. Pela sua natureza as adjacências tem funções de busca associadas que possibilitam métodos expeditos de análise sintáctica [Jorge94].

Formalmente uma GRAD é um quintuplo  $(V_T, V_N, S, P, A)$ , em que  $V_T$  constitui o alfabeto dos Sinais Visuais (ou símbolos terminais),  $V_N$  designa o conjunto dos símbolos não terminais,  $S \in V_N$  identifica o símbolo inicial e  $P$  é um conjunto de produções da forma indicada acima. O conjunto  $A$  inclui todas as relações de adjacência utilizadas na gramática.

### Exemplo

Nesta secção mostra-se uma GRAD que descreve grafos orientados, com identificação dos constituintes formais. A figura 1 ilustra o alfabeto e o conjunto das restrições de adjacência. A figura 3 exemplifica uma expressão visual representando um grafo e mostra a respectiva árvore sintáctica.

$V_T = \{ \text{CIRC}, \text{SETA}, \text{TEXTO} \},$   
 $V_N = \{ n, a, \text{nos}, \text{arcos}, \text{grafo} \}, S = \text{grafo}$   
 $P = \{ p_1, p_2, p_3, p_4, p_5, p_6, p_7 \}, A = \{ \text{DentroDe}, \text{ApontaDe}, \text{ApontaPara} \}$   
 $p_1: \text{grafo} \Rightarrow \{ \text{nos}, \text{arcos} \}$   
 $p_2: \text{arcos} \Rightarrow \{ \text{arcos}, a \}$   
 $p_3: \text{nos} \Rightarrow \{ \text{nos}, n \}$   
 $p_4: \text{arcos} \Rightarrow \{ a \}$   
 $p_5: \text{nos} \Rightarrow \{ n \}$   
 $p_6: n \Rightarrow \{ \text{CIRC}, \text{TEXTO} \} \text{ if } (\text{DentroDe}(\text{CIRC}, \text{TEXTO}))$   
 $p_7: a \Rightarrow \{ n_1, \text{SETA}, n_2 \} \text{ if } (\text{ApontaDe}(n_1, \text{SETA})$   
 $\quad \quad \quad \& \text{ApontaPara}(\text{SETA}, n_2))$

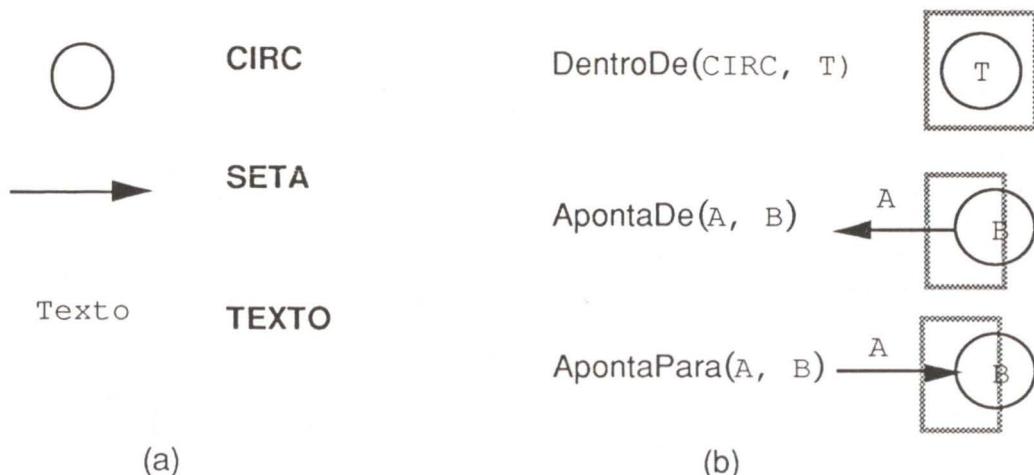


Figura 1: Sinais Visuais (a) e Adjacências (b)



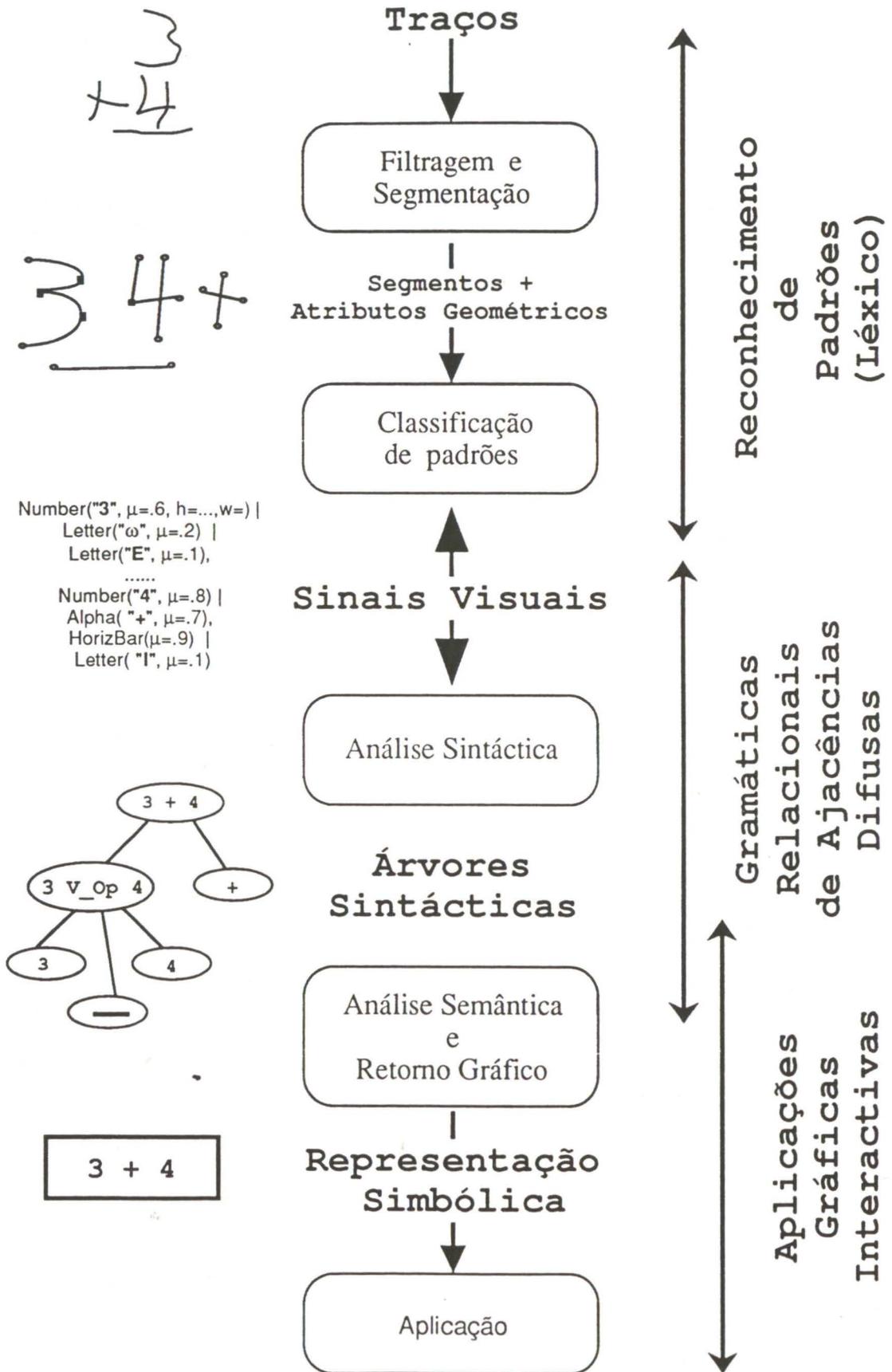


Figura 2: Diagrama de blocos de Interface Caligráfica

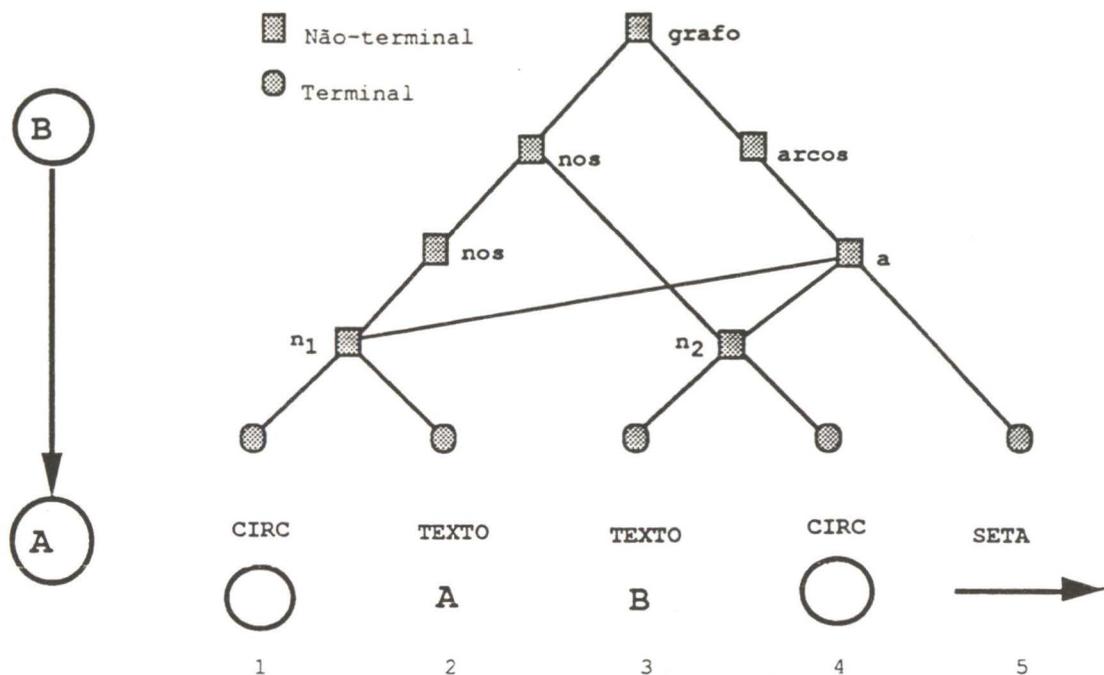


Figura 3: Uma expressão visual e respectiva árvore sintáctica

## Bibliografia

- [Apte93] Ajay Apte, Van Vo e Takayuki D Kimura, "Recognizing Multi-Stroke Geometric Shapes: An Experimental Evaluation", UIST, Atlanta GA, Novembro 1993.
- [Rubine91] Dean H Rubine, "The Automatic Recognition of Gestures", PhD Thesis, Carnegie Mellon University, Dezembro 1991.
- [Wittenburg91] Kent Wittenburg, Louis Weitzman e Jim Talley, "Unification-Based Grammars and Tabular Parsing for Graphical Languages", Journal of Visual Languages and Computing, 2[4], Dezembro 1991
- [yu91] Bing Yu e Shi-Kuo Shang, "A Fuzzy Visual Language Compiler", IEEE Workshop on Visual Languages, Los Alamitos, CA, Outubro 1990.
- [Golin91] Eric J Golin, "Parsing Visual Languages Using Picture Layout Grammars", Journal of Visual Languages and Computing, 2[4], 1991.
- [Golin93] Eric J Golin e Tom Maglieri, "A Compiler Generator for Visual Languages", IEEE Workshop on Visual Languages, 1993.
- [Knuth68] Donald E Knuth, "The Semantics of Context-Free Languages", Mathematical Systems Theory, 2[2], 1968.
- [Selker88] Ted Selker and Larry Koved, "Elements of Visual Language", IEEE Workshop on Visual Languages, Pittsburgh PA, 1988.
- [Zadeh94] Lotfi A Zadeh, "Fuzzy Logic, Neural Networks and Soft Computing", Communications of the ACM, 37[3], Março 1994.
- [Johson74] S C Johnson, "Yet Another Compiler-Compiler", Bell Labs, 1974.

