

# INTERFACES GRAFICAS EM SISTEMAS DE CAD

Joaquim Jorge

INESC, Rua Alves Redol, 9

Neste artigo descreve-se um sistema gráfico interactivo, destinado ao suporte de aplicações de CAD. O sistema inspira-se na proposta de normalização gráfica CGI e foi desenvolvido no âmbito de um projecto de CAD destinado à indústria nacional de confecções. Actualmente o sistema encontra-se em utilização no contexto de outros projectos do INESC.

## Introdução

Nos últimos anos tem-se assistido a um desenvolvimento extremamente rápido das arquitecturas de computadores e sistemas gráficos, sobretudo na área dos pequenos sistemas. O tempo de vida das aplicações tende deste modo a superar o tempo de utilização típico dos equipamentos. O desenvolvimento de aplicações de CAD transportáveis entre sistemas torna-se numa necessidade a ter em conta de modo a preservar os investimentos realizados em código ao migrar as aplicações entre equipamentos de diversos fabricantes e com configurações díspares.

Por outro lado no contexto de aplicações de CAD utilizam-se uma variedade de periféricos gráficos: terminais gráficos, traçadores de caneta, mesas digitalizadoras, etc.

O desenvolvimento de interfaces gráficas normalizadas permite responder aos requisitos de portabilidade do software e independência dos dispositivos gráficos[2]. No entanto tradicionalmente a generalidade e portabilidade de aplicações gráficas tem sido obtida à custa de sacrifícios no desempenho dos pacotes de funções gráficas. Em aplicações interactivas a rapidez das funções gráficas pode consistir numa característica crítica que afecta a utilização do sistema como um todo.

Por outro lado, a diversidade dos periféricos existentes, coloca requisitos adicionais ao desenvolvimento de pacotes de software genéricos. Estes devem ser escritos de modo a que seja possível a configuração de novos dispositivos num tempo relativamente curto e com um mínimo de esforço de programação.

A escolha de um pacote de funções gráficas de base consiste numa das decisões importantes no projecto de um sistema de CAD incluindo



aplicações gráficas interactivas. O sistema CORTE, desenvolvido no INESC, constitui um exemplo de um sistema deste tipo.

O sistema deveria correr em máquinas UNIX multi-posto e de reduzido custo, de modo a ser possível a sua utilização por pequenas/médias empresas do sector de confecções. O sistema deveria ser facilmente transportável e configurável para dispositivos gráficos existentes no mercado. Por outro lado, as aplicações interactivas no sistema deveriam ser suficientemente rápidas por forma a permitir realização de operações complexas em tempos aceitáveis.

Os requisitos de portabilidade e independência dos dispositivos gráficos sugeriam a adopção da norma GKS. No entanto tal não foi possível por vários motivos:

- Não existir uma implementação de GKS sobre a configuração inicial de desenvolvimento
- Ser difícil garantir o desempenho interactivo do sistema, dadas as limitações de velocidade de processamento do micro-computador utilizado.
- A implementação de um pacote gráfico como o GKS, com um grande número de funções, tomaria uma grande parte do tempo disponível para o desenvolvimento do projecto

A decisão final recaiu sobre o desenvolvimento de um sistema gráfico simples baseado no modelo de referência do GKS, e satisfazendo os requisitos de portabilidade e desempenho enunciados

### A norma gráfica CGI

A proposta de norma gráfica CGI (Computer Graphics Interface)[3] permite a representação de uma vasta gama de imagens a duas dimensões numa vasta gama de dispositivos gráficos. A aplicação acede aos dispositivos gráficos através de uma abstracção semelhante ao conceito de "Workstation" no GKS: o dispositivo virtual. Múltiplos dispositivos virtuais podem ser simultaneamente acedidos por uma aplicação.

A norma CGI permite descrever e comunicar informação gráfica entre um programa de aplicação e um ou mais dispositivos gráficos de uma forma independente das características destes dispositivos

Os conceitos desenvolvidos no CGI definem a forma (sintaxe) e comportamento funcional (semântica) de uma forma muito semelhante ao GKS, embora seja dada uma maior atenção à representação e aspectos visuais das primitivas de saída gráficas. As funções CGI dividem-se em 6 grandes grupos.

- Controle, negociação e erros: Permitem especificar os modos de operação de outras funções da norma, permitem o início e término de sessão e controlam a operação do dispositivo virtual

- Primitivas gráficas de saída: descrevem os componentes visuais da figura no dispositivo virtual

- Funções de controle de atributos gráficos: Permitem o controle do aspecto (representação) das primitivas gráficas de uma forma modal.

- Funções de segmentação: permitem manipular grupos de primitivas e respectivos atributos

- Funções de entrada de dados fornecem a capacidade de obter valores de um dispositivo, controlando a forma e temporização dessa obtenção.

- Funções Raster podem ser utilizadas para gerar e manipular imagens em dispositivos raster. Esta constitui uma das diferenças mais notáveis entre o CGI e o GKS em termos de modelo de visualização, explicável pelo facto de se tratar de uma norma mais recente.

Uma característica importante do CGI consiste na definição de perfis, ou subconjuntos de funções especificadas na norma, permitindo o suporte de um número mínimo de funções em dispositivos conformes com a norma, de uma maneira mais flexível do que a definição de níveis no GKS.

## Implementação

A implementação CGI efectuada no INESC (CGI/INESC), teve por objectivos principais:

- Constituir um núcleo gráfico simples e eficiente, que possa funcionar em máquinas de reduzido porte
- Facilitar a configuração de novos dispositivos no sistema
- Elevado desempenho no desenho de primitivas gráficas e operações envolvendo interacção do operador
- Utilizar as capacidades de manipulação local de imagem, tais como segmentação sempre que estas se encontrem.

disponíveis, recorrendo a técnicas de emulação em dispositivos que não suportem este tipo de funções.

O pacote de funções está escrito em linguagem C, tendo sido transportado para vários sistemas operativos: UNIX, MS-DOS e AOS/VS, permitindo a utilização de vários dispositivos gráficos:

- terminais TEKTRONIX 4111 e 4114
- terminais CIPHER T5 (terminais de baixo custo implementando o repertório de comandos 4014)
- traçadores de caneta BENSON (vários modelos)
- Mesas digitalizadoras SUMMAGRAPHICS (ID-40 e MicroGrid) e IBM 5084
- Estações gráficas Digital, Data-General e IBM (PC/RT)
- Traçador de caneta especial desenvolvido pela EFACEC para o projecto Corte

Creemos terem sido atingido os objectivos com a realização deste pacote de funções gráficas. O número de dispositivos suportados atesta a facilidade de configuração do CGI/INESC. Indicamos em seguida as características de arquitectura que em nossa opinião tornam isto possível:

### **Sistema de coordenadas inteiro**

O CGI/INESC implementa um espaço de coordenadas virtual utilizando numeros inteiros com 32 bits de precisão. Todas as funções de desenho de primitivas requerem parametros inteiros e utiliza-se aritmética de virgula fixa sempre que possível. Isto possibilita o melhor desempenho possível em sistemas sem processador de virgula flutuante, mantendo uma precisão suficiente para suportar a maioria das aplicações que utilizam o CGI.

### **Seleccção de múltiplos dispositivos virtuais**

Cada dispositivo virtual aberto é acedido através de um identificador distinto. Este identificador permite às funções do CGI o acesso directo às funções de controlo, tabelas de descrição e de estado do dispositivo gráfico, tornando eficiente a escrita em vários dispositivos gráficos abertos simultaneamente sem desperdício de tempo devido a comutação de contexto.

### **Primitivas gráficas de saída**

O CGI/INESC suporta um subconjunto das primitivas indicadas na norma, nomeadamente linhas poligonais, polígonos, marcas e texto. Outras primitivas podem ser facilmente acrescentadas como rectangulos, arcos de circunferencia, elipses e "splines" cardinais. A interface gráfica mantém uma imagem dos atributos gráficos no dispositivo, por forma a minimizar o tráfego de comunicações em linhas assíncronas de baixa velocidade e os tempos mortos devido a mudanças de contexto.

## Transformações

O CGI/INESC mantém uma lista de transformações de visualização associada a cada dispositivo virtual aberto. Desta forma é possível a uma aplicação controlar a organização do ecran sem perca de eficiencia no desenho de primitivas gráficas.

## Clipping

O clipping de primitivas gráficas é efectuado contra os limites da superfície de visualização do dispositivo ou de um rectangulo no espaço VDC do dispositivo. Esta operação é especificada individualmente por dispositivo, a fim de se poder tirar partido das capacidades locais de clipping nos dispositivos que as possuam.

## Entrada de dados

Os aspectos de eco são tratados de forma comum nos vários dispositivos. O gestor de entradas transforma os dispositivos lógicos de entrada normalizados nos dispositivos físicos disponíveis. Este gestor comunica com o sistema por forma a obter dados de entrada, utilizando as funções de desenho de baixo nível associadas a cada dispositivo de modo a ecoar as acções do operador. Tipos de entrada de dados comuns foram realizados a um nível superior ao CGI/INESC de modo a uniformizar a interface utilizador e simplificar a adaptação a diferentes dispositivos físicos.

## Segmentação

O sistema gráfico foi desenhado de modo a utilizar as capacidades locais de segmentação dos dispositivos quando estas existirem. Um conjunto generico de funções de emulação de segmentação existe para fornecer esta capacidade em dispositivos que não suportam esta capacidade. Como excepção note-se que a função de *pick* não se encontra implementada. Isto não constitui problema para as aplicações até agora desenvolvidas, que

utilizam versões mais simples (e portanto mais eficientes) deste algoritmo.

### **Configurabilidade**

A configuração do sistema para diferentes dispositivos é facilitada pela agregação das tabelas de descrição, de estado e das funções de controle do dispositivo físico numa única única estrutura de dados. O identificador de dispositivo permite aceder directamente a esta estrutura através do CGI/INESC de uma forma transparente para os programas de aplicação. A biblioteca de funções de emulação disponíveis permite reduzir a configuração de um novo dispositivo á escrita de funções de controle e comunicação elementares e inicialização das tabelas de descrição numa mesma estrutura de dados.

## Bibliografia

[1] Joaquim Jorge, João Benardo, João Prieto "Sistema Corte: Um sistema de CAD para a Indústria de confecções", las Jornadas de PPP/AC, Lisboa, Dez 1987

[2] Functional Specification of the Graphics Kernel System(GKS) ISO/7942

[3] Information Processing Systems - Computer Graphics - ISO/DP9636 (CGI) -- Interim Draft, Abril 1988.