



Introdução à Computação Gráfica

Marcel P. Jackowski
mjack@ime.usp.br

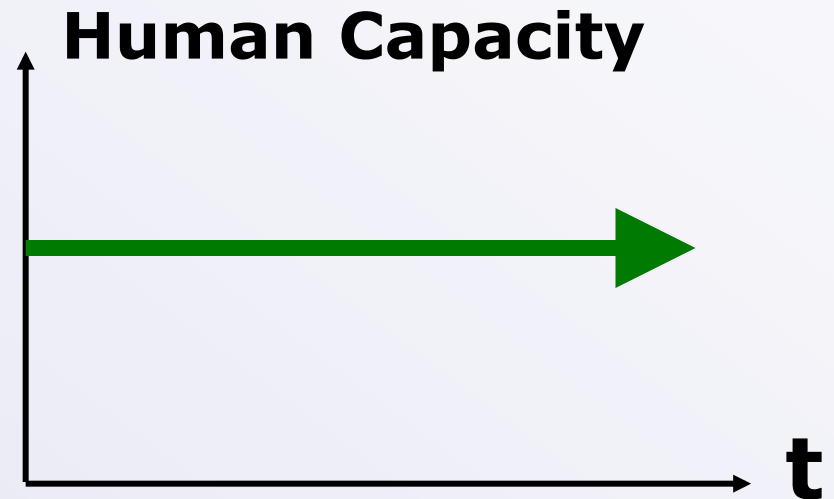
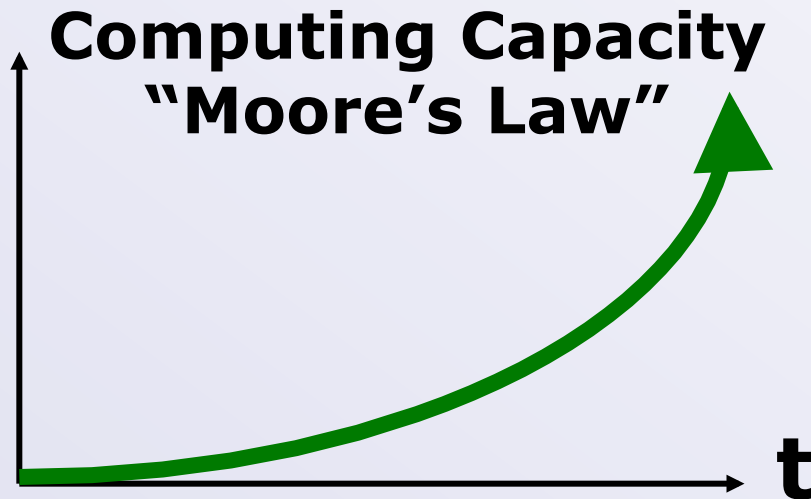
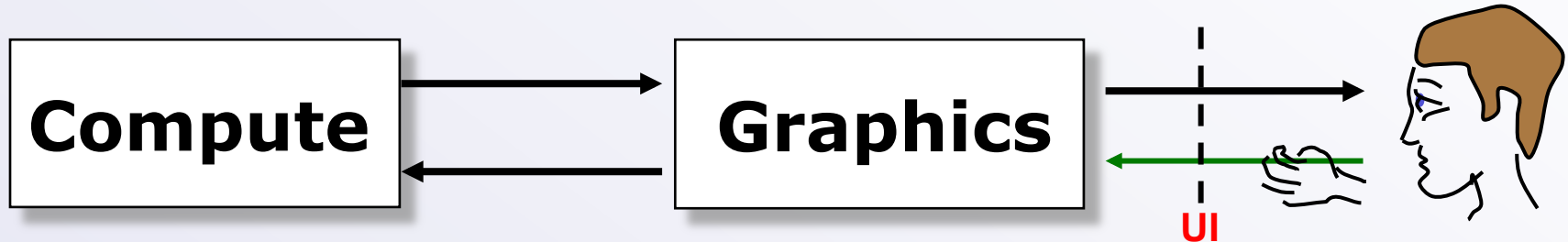
Aula #19



Perguntas

- Qual é o uso mais comum de CG hoje em dia ?
- O que faremos com este aumento contínuo em poder de processamento?
 - A maioria das aplicações já tem o que necessitam...
- Qual a percentagem de código de uma aplicação destinada à interface com usuário (UI) ?
- O que diferencia as diversas aplicações ?

Poder computacional vs. cerebral



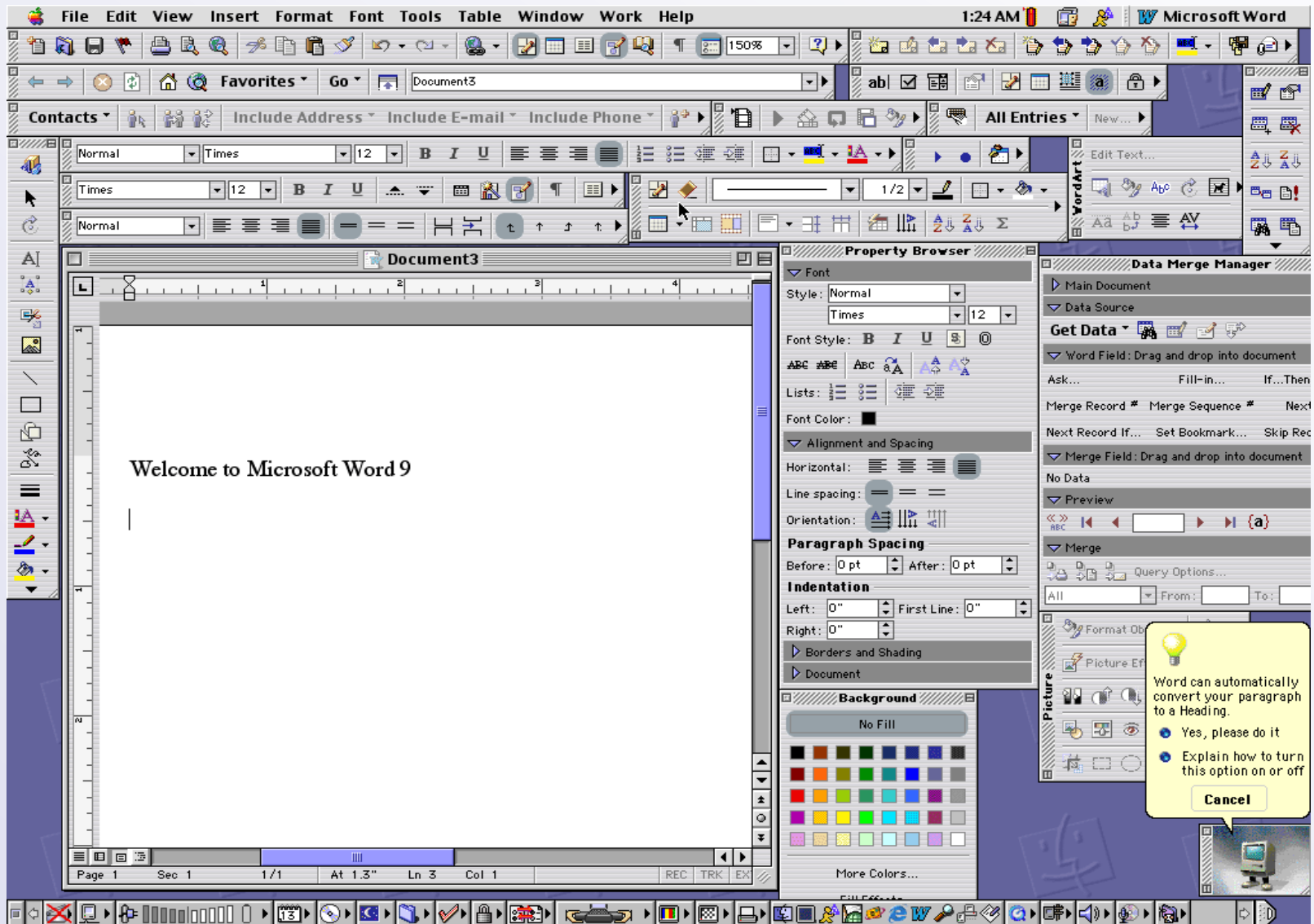
Use compute power in UI to increase b/w to the brain

Uls são importantes!

- Pode virar um caso de vida ou morte
- Queda de um Boeing 757 em 1995, em Cali, Colômbia, devido à uma ambiguidade em um dos comandos do piloto automático



Um extremo: Microsoft Word



Outro extremo: Myst - Revelations



Existe uma interface ideal ?

- Não! Elas são um mal necessário !
- Desejamos que a comunicação e o controle seja feita da mesma forma que na vida real
 - Interação com objetos e agentes
- Modelos de agentes: Jeeves, HAL-9000
 - Entender o contexto: físico, pessoal, social, etc...
 - Inferir intenção
- Futuro: interfaces cérebro-máquina
- Hoje: Uis transparentes e fluídicas

Interação homem-computador (IHC)

- Estudo da interação dos humanos com computadores;
- Desenvolvimento de sistemas de processamento de informações ergonômicos:
 - Eficientes
 - Fáceis de utilizar
 - Adaptáveis ao contexto de utilização

Como se dá essa interação ?

- Interface entre humanos e computadores se dá através de canais de entrada e saída:
 - Visão
 - Audição
 - Tato
 - Sistemas hápticos
 - Paladar ?
 - Olfato ?

Variabilidade de percepção

- Existe grande variabilidade no grau de percepção de estímulos
 - visuais
 - auditórios,
 - táteis, olfatórios e no paladar
- Sistemas IHC devem se adaptar a esta variabilidade, sempre que possível.

Interação homem-computador (IHC)

Disciplina preocupada com:

- Design
- Implementação
- Avaliação

de sistemas interativos de processamento de informações para humanos e seus fenômenos.

MAC0446/5768: Princípios de IHC

http://www.sigchi.org/

What is MedSquare about ? | Med...

SEM540: Lectures 10 to 12.</head>


Welcome — SIGCHI


Site Map

Accessibility

Contact

Google



The Association for Computing Machinery ()
Special Interest Group on Computer Human Interaction

Home

Connect

About SIGCHI

People


Resources

Publications

Conferences

Communities

You are here: [Home](#)

NEWS 

SIGCHI Blog

Feb 07, 2012

[Executive Committee Meeting Notes](#)
Notes of the January 2012 SIGCHI Executive Committee Meeting have been added.

Dec 04, 2011


[SIGCHI Member Judith Olson Named ACM "Athena Lecturer"](#)
The Association for Computing Machinery's Council on Women in Computing (ACM-W) has named Judith S. Olson of the University of California, Irvine as the 2011-2012 Athena Lecturer.

[More...](#)


Welcome

SIGCHI is the premier international society for professionals, academics and students who are interested in human-technology & human-computer interaction (HCI).

To get involved you can [join SIGCHI](#), join one of our [mailing lists](#), become a [volunteer](#), or visit your [local SIGCHI chapter](#).



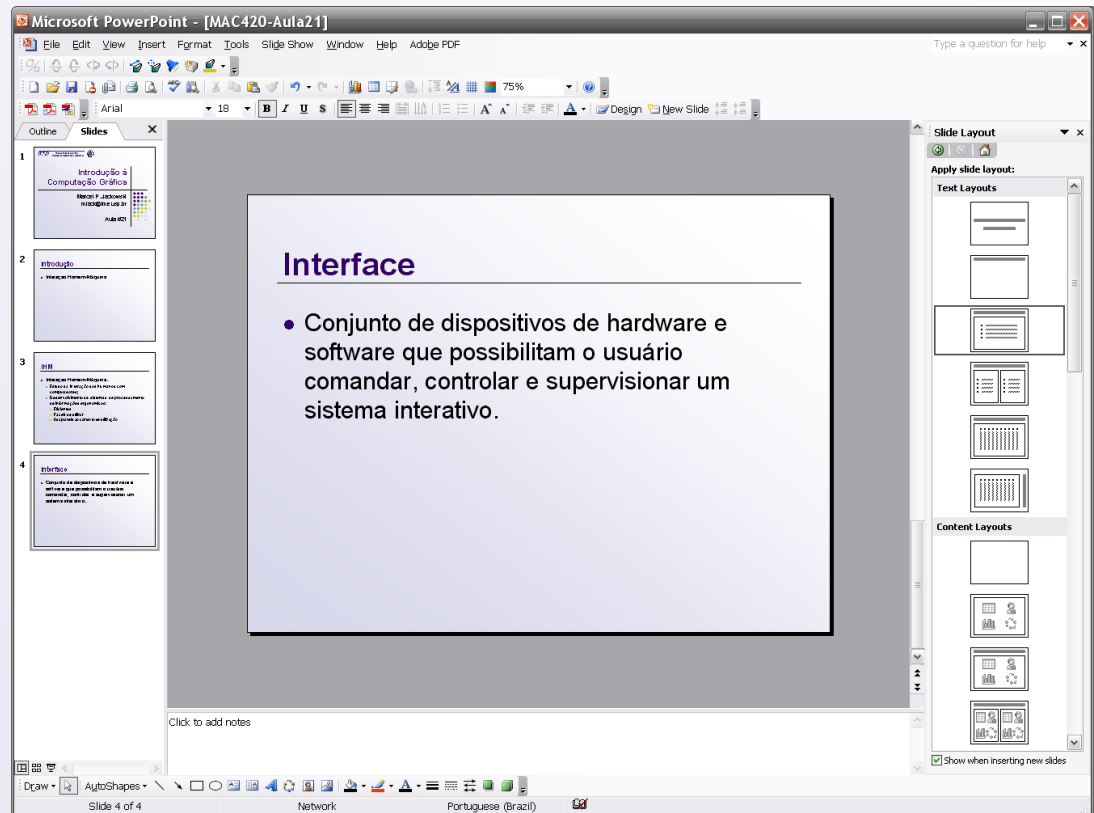
[SIGCHI Photo History](#)



Twitter / acmchi2012
acmchi2012: Hey #chi2012 attendees, help us make #chi2013 awesome. Fill out the conference survey! It's in your inbox.
May 22, 2012
acmchi2012: Thanks everyone! This has been a great #chi2012 ""Oh, please don't go—we'll eat you up—we love you so!" ~ Where the Wild Things Are
May 10, 2012
acmchi2012: At 3:50, help celebrate CHI's 30th birthday on the 4th floor!. #chi2012
May 10, 2012

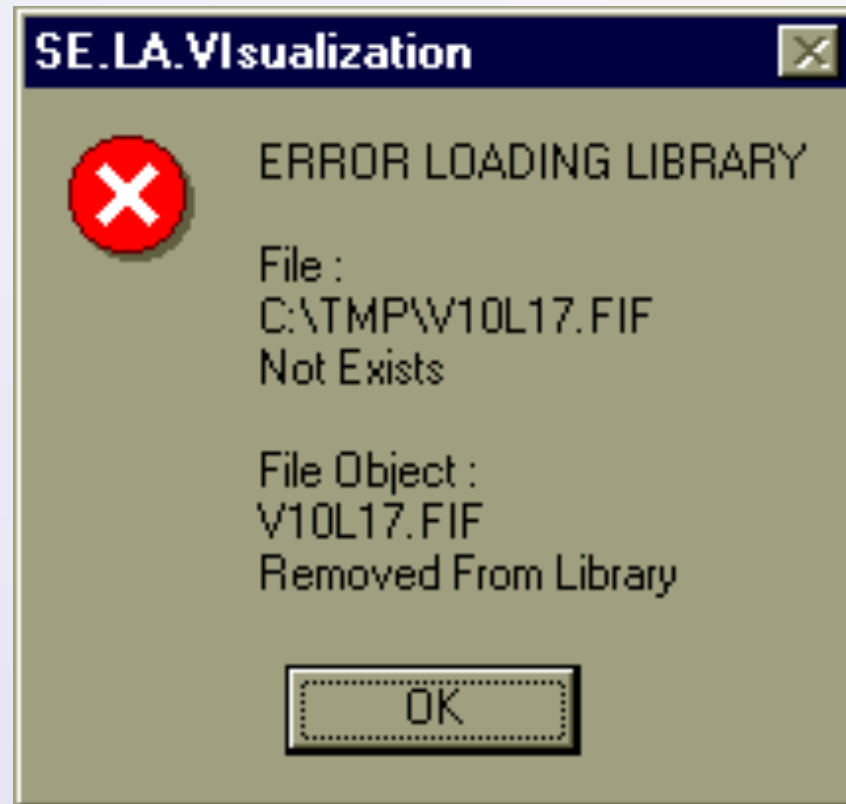
Interface

- Conjunto de dispositivos de hardware e software que possibilitam o usuário comandar, controlar e supervisionar um sistema interativo.

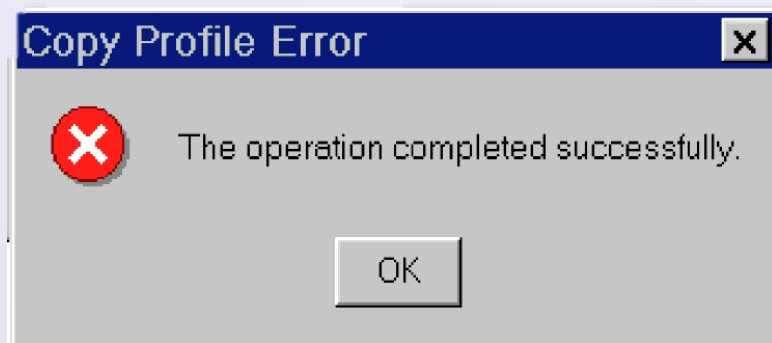
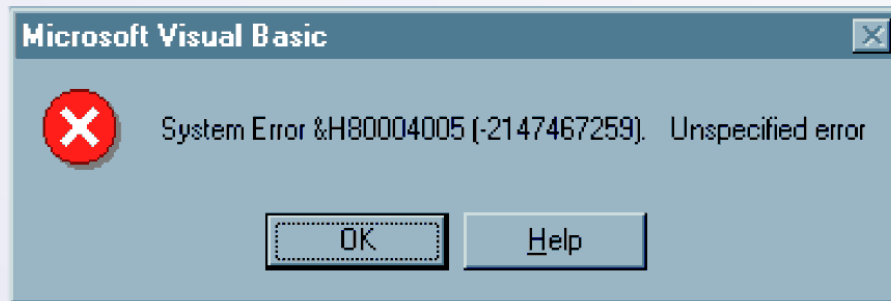


Design de interfaces

- O que há de bom e de ruim na interface abaixo ?



Alguns maus exemplos



Turning off the Office Assistant.

You've hidden me several times now. Would you like to permanently turn me off or just hide me again?

- No, just hide me
- Yes, turn me off
- Change other options



Interação vs Interface

- Componentes: botões, menus, cores, textos, animações, etc.
- Eles são suficientes para que um sistema seja utilizável ?
- Para medir a usabilidade:
 - Tempo de treinamento;
 - Velocidade de uso;
 - Taxa de erro;
 - Facilidade em recordar;
 - Satisfação objetiva;
 - etc;

Interação vs interface

- A interface conta muito, mas a interação a precede:
 - Sequência de ações necessárias para realizar uma tarefa
 - Adequação entre o sistema e o contexto de utilização.

Interface gráfica

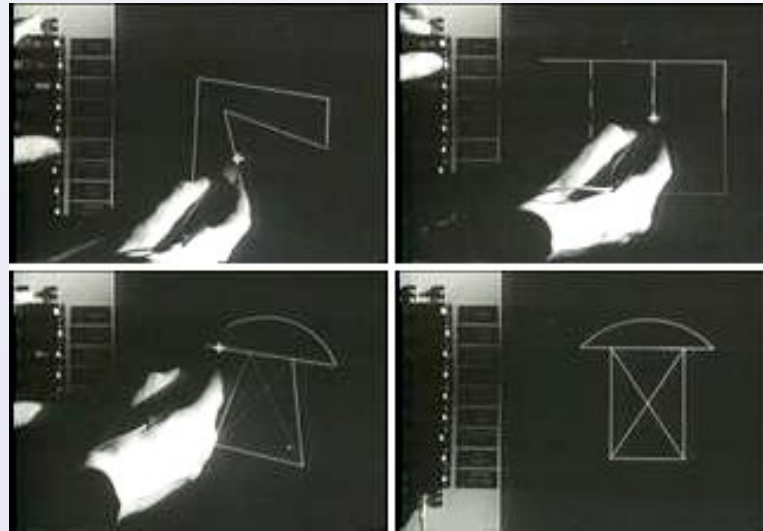
- Evolução:
 - Linha de comando
 - Acesso a uma função do sistema;
 - Menus e telas de entrada de dados
 - Acesso a uma aplicação (subconjunto de funções de um sistema)
 - Janelas múltiplas, interfaces icônicas e manipulação direta
 - Acesso a todas as funções de um sistema.

Interatividade

- Grau de interatividade:
 - Número e natureza das trocas de informações com os usuários.
- Dois elementos importantes:
 - Execução de várias tarefas em paralelo
 - Advento das interfaces gráficas

Breve história das UIs

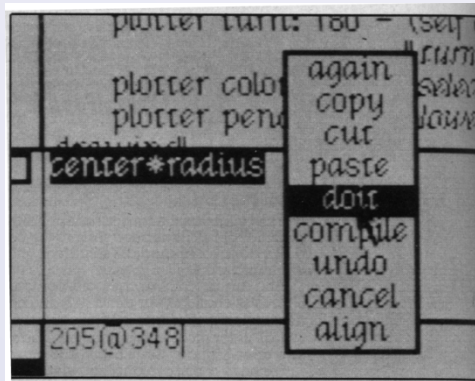
- 1963, Ivan Sutherland lançou o Sketchpad, contendo botões físicos, teclas, e mecanismos para translação e zoom.



- 1968, Engelbart demonstrou um sistema de hipermídia que incluía janelas, mouse, teclado e interface baseada em comandos
- 1970, Engelbart patentou o mouse
 - Apple “empresta” o mouse da Xerox PARC; que tinha “emprestado” de Engelbart

Breve história das UIs

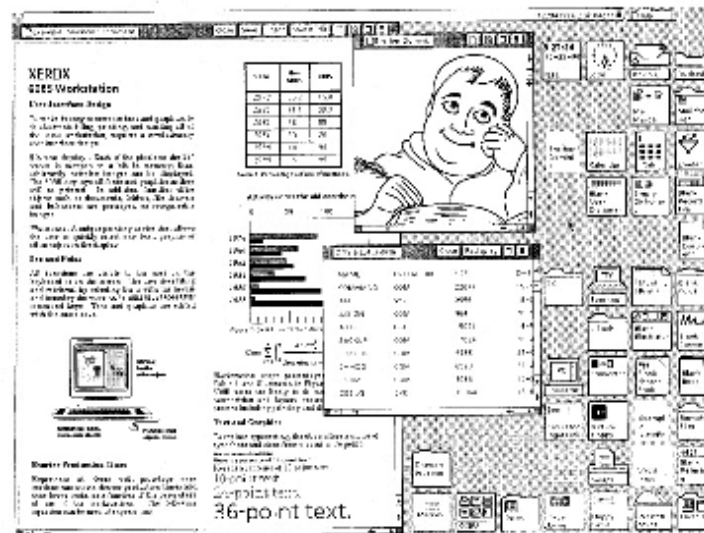
- 1973, Xerox PARC produziu o Alto, a primeira workstation pessoal. Era baseada em display gráfico bitmap, com mouse, Ethernet, e arquitetura cliente-servidor. Ela também tinha o primeiro editor de texto WYSIWYG, Smalltalk (Alan Kay, Adele Goldberg, et. al.), e GUI WIMP incluindo gerenciador de janelas



Smalltalk-80 screenshot

Breve história das UIs

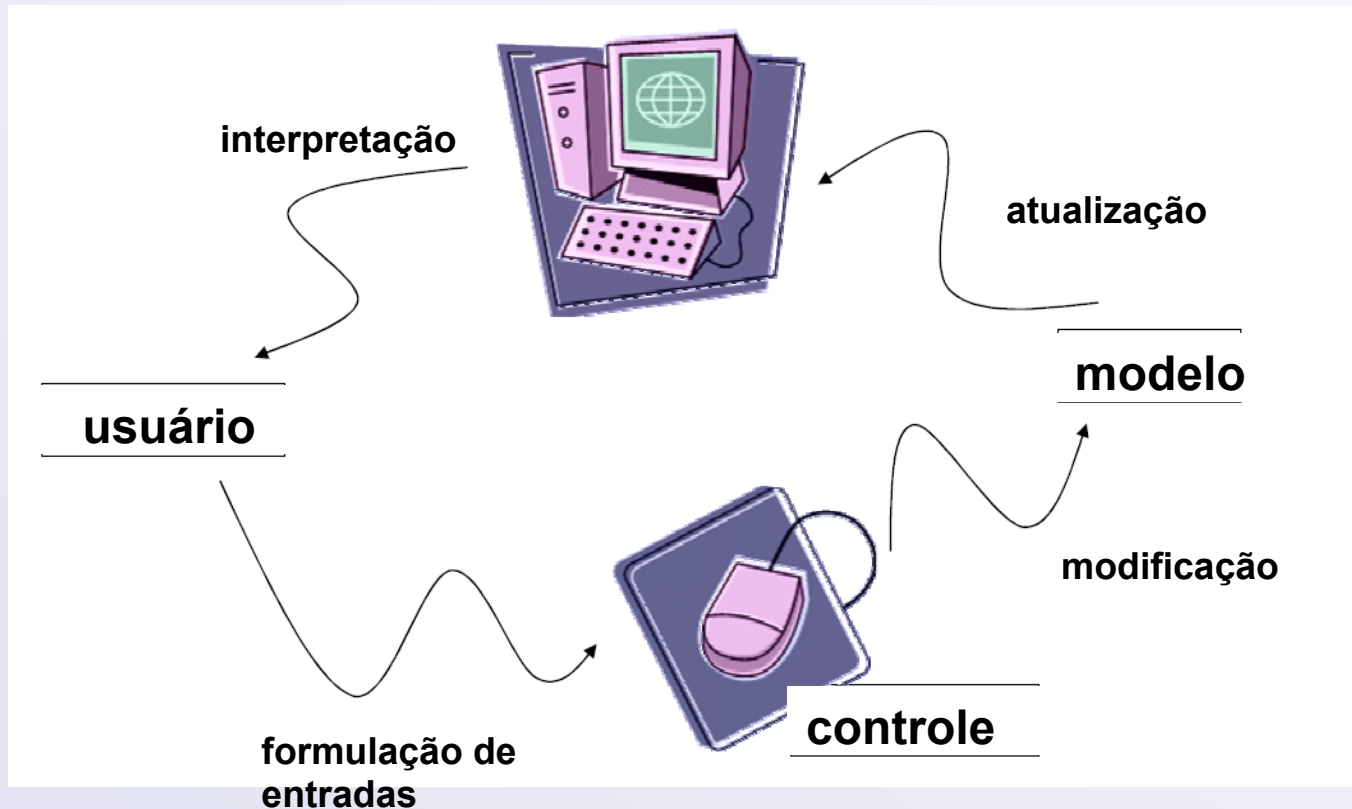
- 1981, Xerox apresentou o Star Information System baseado em janelas sobrepostas



- 1984, Apple lançou o primeiro Mac como o primeiro PC gráfico, baseado no Alto e o Star
- 1985, Microsoft Windows, é considerado uma imitação dos Macs com poucas melhorias
 - Famosa guerra legal entre Apple e Microsoft em relação ao Windows usar ícones, cursores, etc.
 - Apple perde todos os casos

O ciclo de interação

- Padrão MVC (Model-View-Controller)



Manipulação direta

"Point and click instead of remember and type"

Schneiderman, 1983

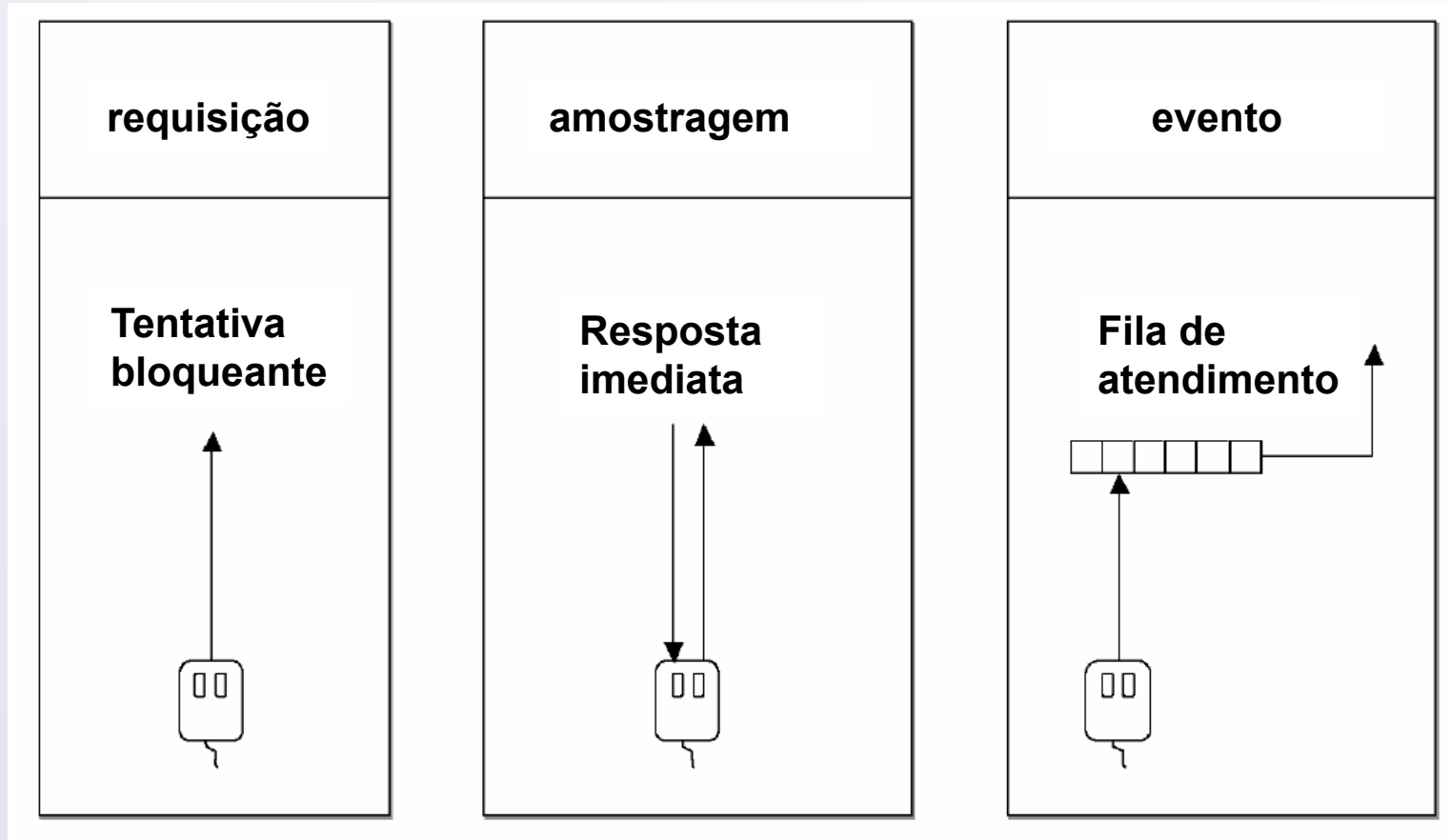
- Baseado em 3 princípios:
 - Representação visível dos objetos
 - Utilização de ações físicas para manipulação
 - Operações rápidas, incrementais e reversíveis



Periféricos de entrada

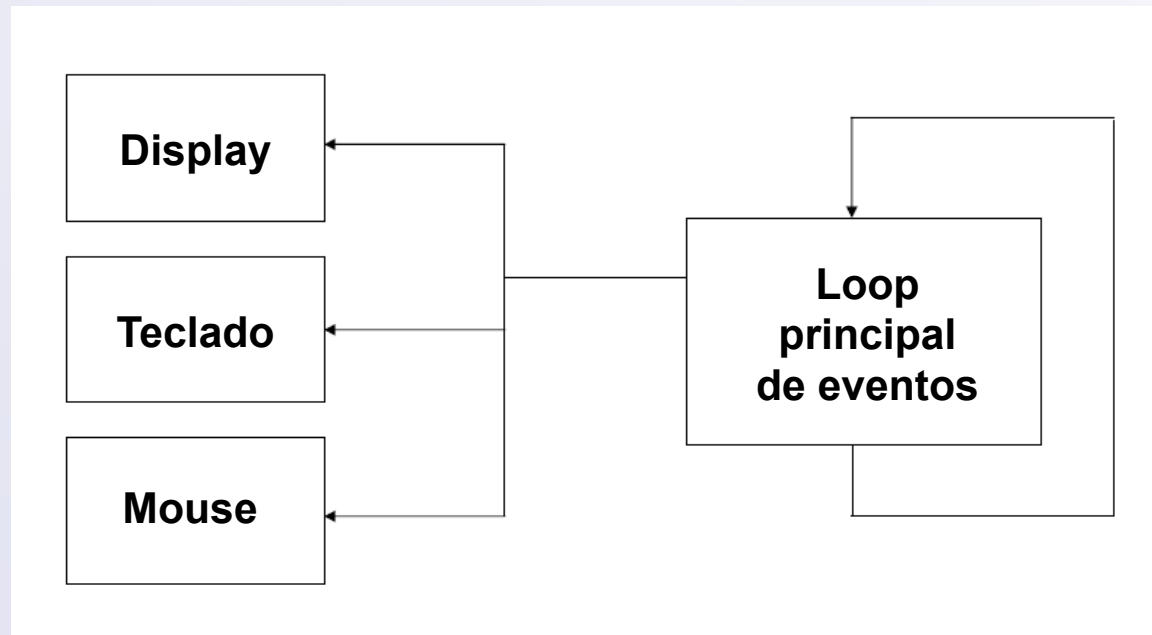
- Absoluta:
 - Tablets, telas sensíveis ao toque, caneta ótica
- Relativa:
 - Mouse, joystick, trackball, sensor de movimentos
- Direta:
 - Tela sensível ao toque
- Indireta:
 - Mouse, joystick, trackball

Gerenciamento de periféricos



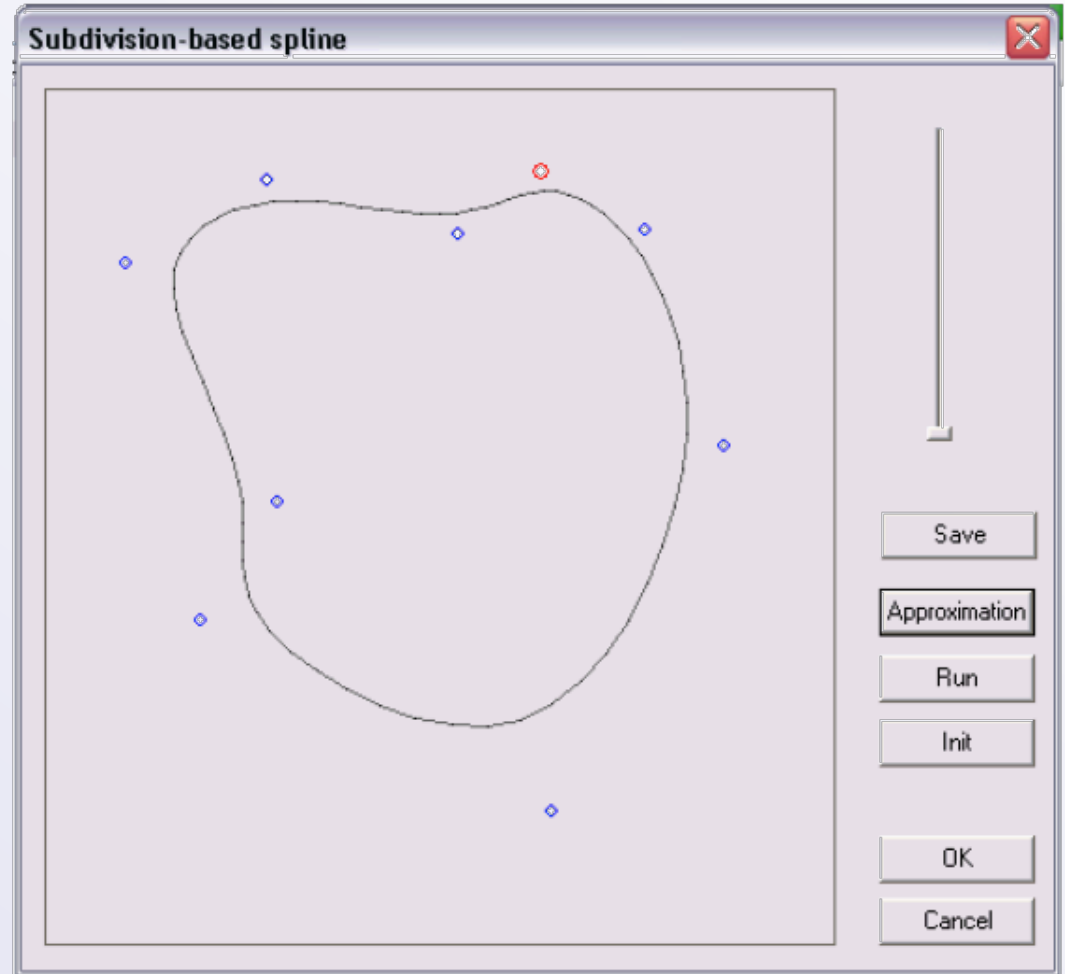
Programação por eventos

- O programa responde à eventos;
- Os eventos, por sua vez, são interceptados por funções que definem a sua utilização;
- As funções que processam os eventos devem saber o contexto e o tipo de evento passada na variável.



Interfaces WIMP (1980)

- Windows
- Icons
- Menus and controls
- Pointing device

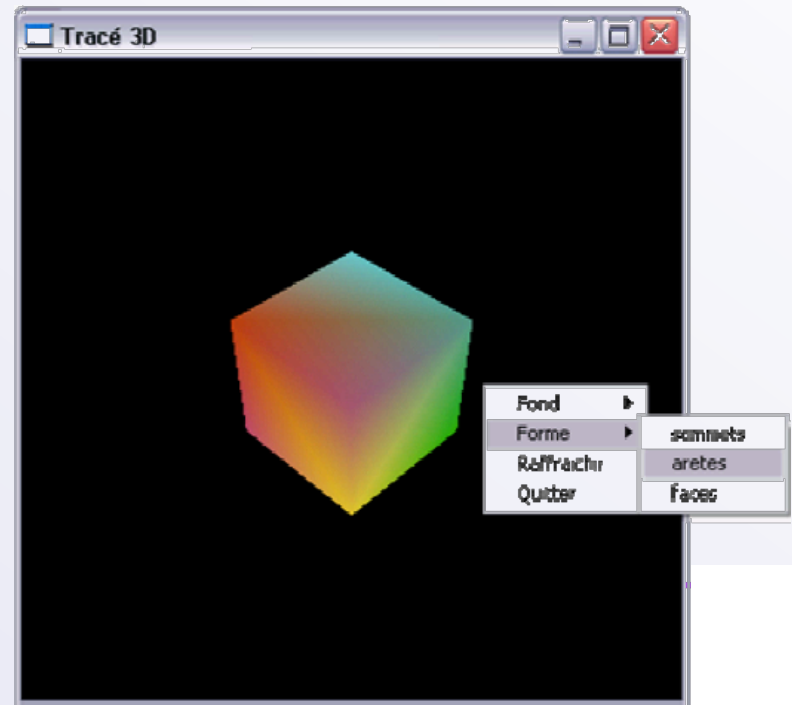


GLUT

- Interface programável pelo usuário

```
(...)  
menuFondId = glutCreateMenu (menuFond);  
glutAddMenuEntry ("Noir", 0);  
glutAddMenuEntry ("Rouge", 1);  
glutAddMenuEntry ("Vert", 2);  
glutAddMenuEntry ("Bleu", 3);  
glutAddMenuEntry ("Blanc", 4);  
//Menu principal  
glutCreateMenu (menuGeneral);  
glutAddSubMenu ("Fond", menuFondId);  
glutAddMenuEntry ("Raffraichir", 0);  
glutAddMenuEntry ("Quitter", 1);  
  
//attache le menu au bouton droit de la souris  
glutAttachMenu (GLUT_RIGHT_BUTTON);
```

criação da interface

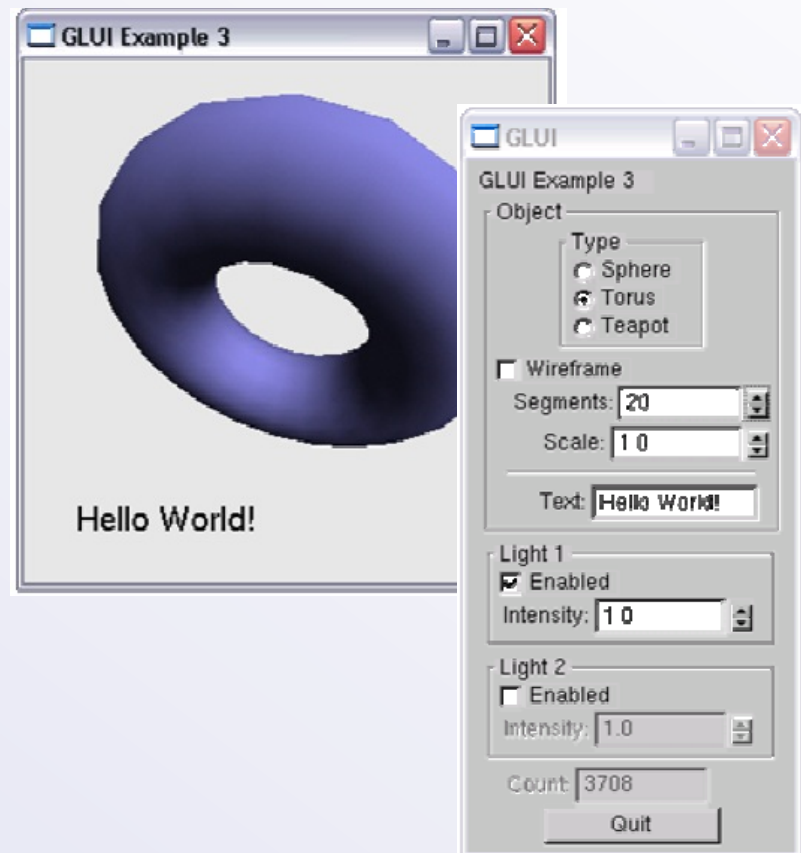


exemplo de interface

GLUI

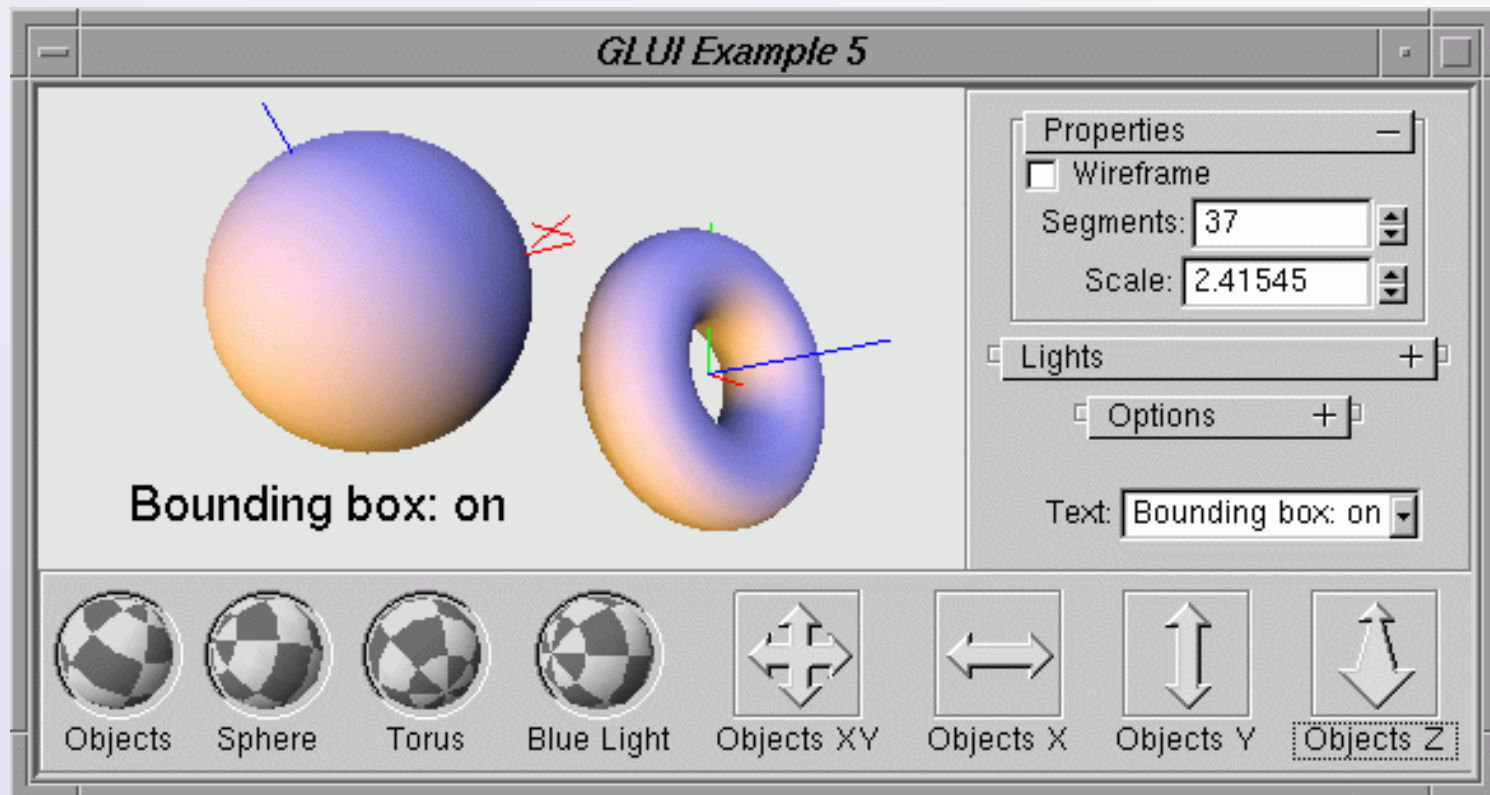
- Simples, porém utiliza-se
 - (void *)
 - variáveis globais
- Windows e Linux-Unix

```
(...)  
GLUI *glui = GLUI_Master.create_glui( "GLUI" );  
glui->add_checkbox( "Wireframe", &wireframe );  
GLUI_Spinner *segment_spinner =  
    glui->add_spinner( "Segments:",  
        GLUI_SPINNER_INT, &segments );  
segment_spinner->set_int_limits( 3, 60 );  
  
glui->set_main_gfx_window( main_window );  
  
/* We register the idle callback with GLUI,  
*not* with GLUT */  
GLUI_Master.set_glutIdleFunc( myGlutIdle );  
  
glutMainLoop();
```

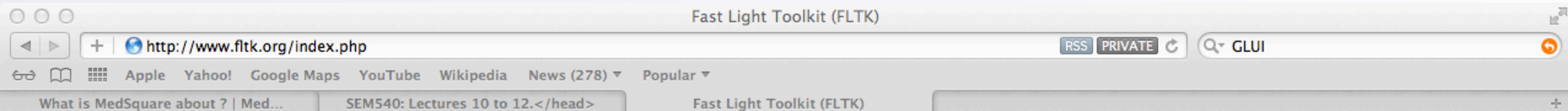


GLUI

- <http://www.cs.unc.edu/~rademach/glui/screen3.gif>



FLTK



Fast Light Toolkit

[FLTK Apps](#)[FLTK Library](#)[Forums](#)[Links](#)[Login](#)[\[Home | Articles & FAQs | Bugs & Features | Documentation | Download | Screenshots \]](#)

Quick Info

Stable Release: **v1.3.0**

Snapshots: **v1.3.x-r9573**
v2.0.x-alpha-r9296
v3.0.x-r9573

Nightly Builds: [View Link](#)

Developer Roadmap: [View](#)

FLTK (pronounced "fulltick") is a cross-platform C++ GUI toolkit for UNIX/Linux® (X11), Microsoft® Windows®, and MacOS® X. FLTK provides modern GUI functionality without the bloat and supports 3D graphics via OpenGL® and its built-in GLUT emulation.

FLTK is designed to be small and modular enough to be statically linked, but works fine as a shared library. FLTK also includes an excellent UI builder called FLUID that can be used to create applications in minutes.

FLTK is provided under the terms of the [GNU Library Public License, Version 2](#) with exceptions that allow for static linking.

Current Poll [\[Show All \]](#)

The FLTK website was designed with speed in mind. The design as a result is very

Recent Articles [\[Show All \]](#)

FLTK 3.0.x Weekly Snapshot, r9573

01:03 Jun 08, 2012 by mike, 0 comments

A new weekly snapshot of FLTK 3.0.x (r9573) is now available. [\[Read \]](#)

FLTK 1.3.x Weekly Snapshot, r9573

01:01 Jun 08, 2012 by mike, 0 comments

A new weekly snapshot of FLTK 1.3.x (r9573) is now available. [\[Read \]](#)

XMolView alpha 7

16:33 Jun 04, 2012 by etorres, 0 comments

XMolView (alpha 7) [\[Read \]](#)

FLTK 3.0.x Weekly Snapshot, r9564

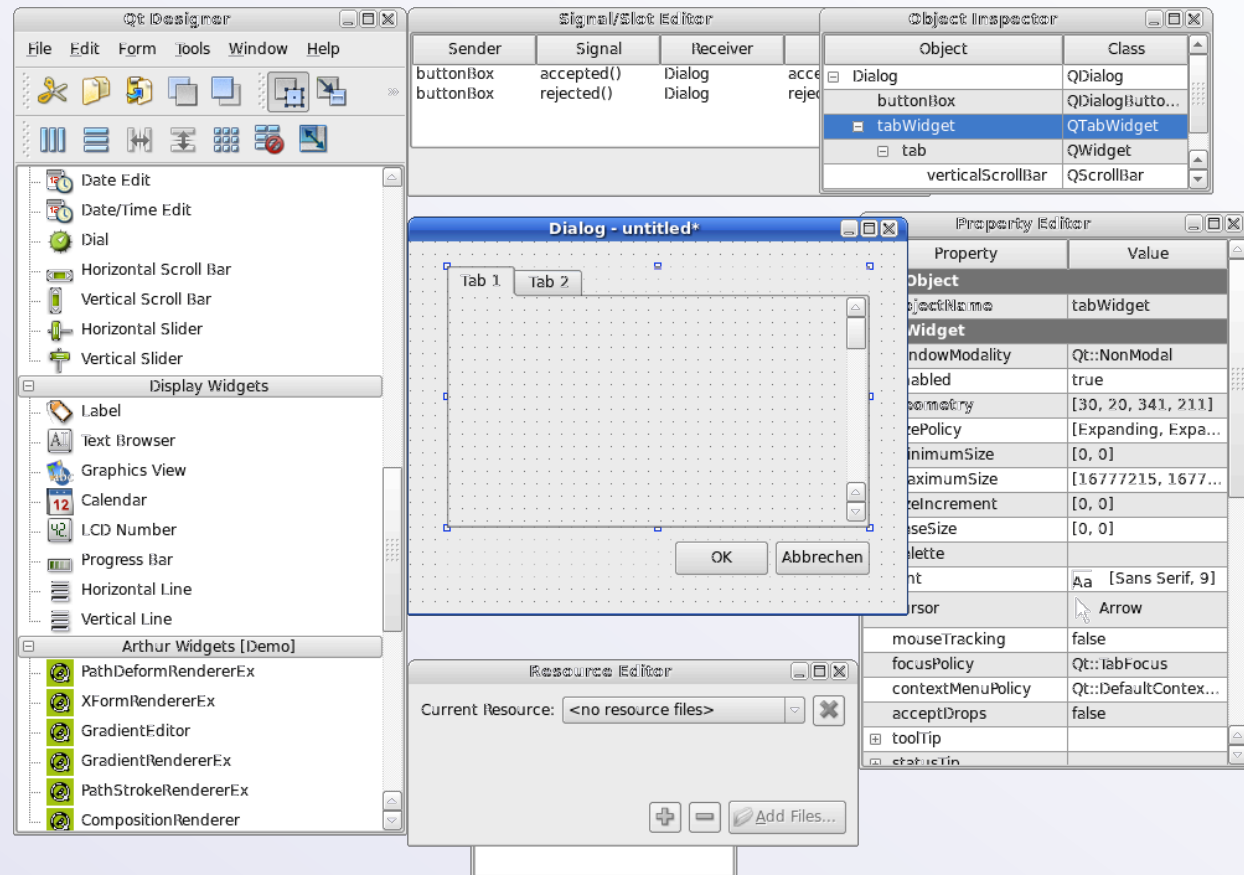
01:02 Jun 01, 2012 by mike, 0 comments

A new weekly snapshot of FLTK 3.0.x (r9564) is now available. [\[Read \]](#)



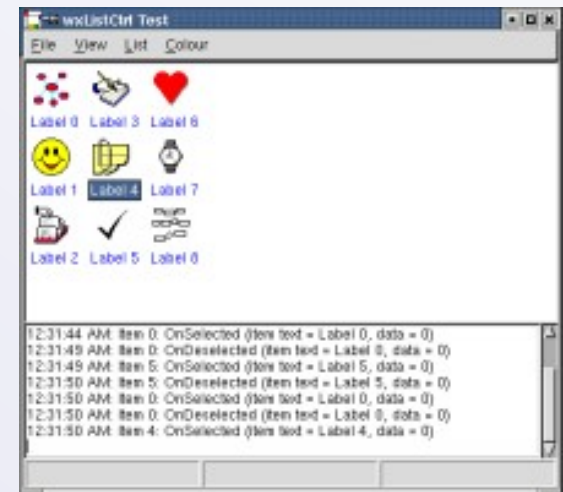
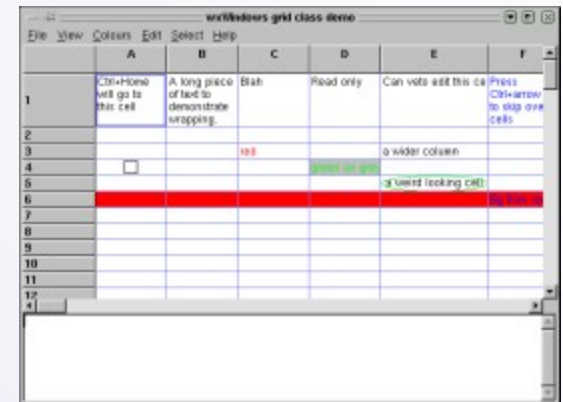
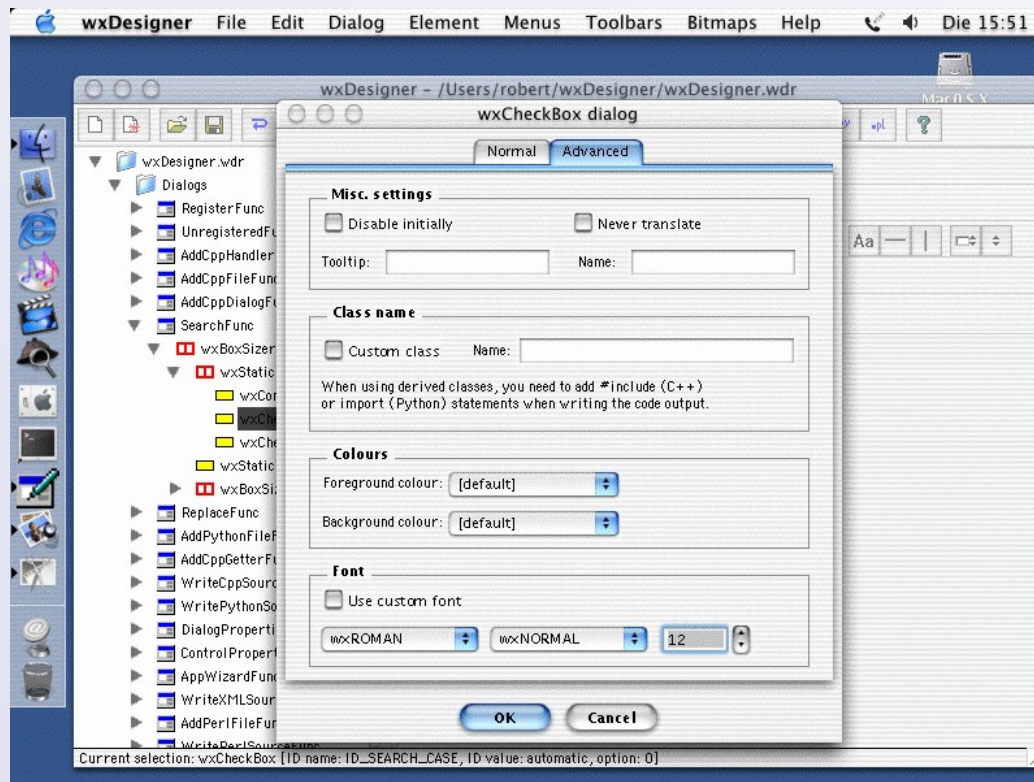
Qt (antes Trolltech, agora Nokia)

- Multiplataforma (Unix, Windows, Mac, etc)
- Qt designer
- Passagem de mensagens entre objetos;



wxWidgets

- Multiplataforma
- Baseada em objetos (C++)

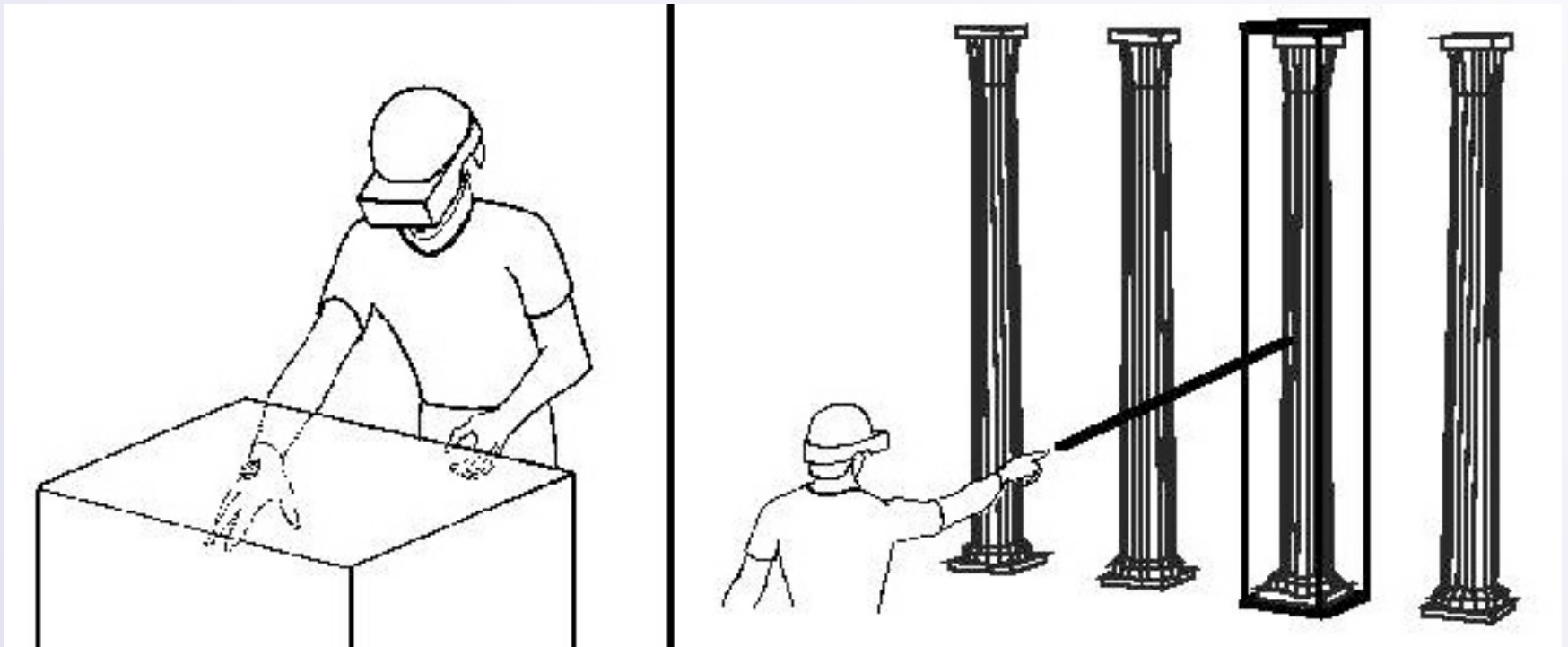


Realidade virtual (pós-WIMP)

- Salto em relação às interfaces gráficas (“WIMP”)
- Caracterizada pela sensação de presença ou imersão dentro da cena
- A qualidade desta imersão, ou grau de ilusão, ou quão real esta ilusão parece ser, depende da interatividade e do grau de realismo que o sistema é capaz de proporcionar

Realidade virtual

- Mover ou selecionar um objeto



Tarefas elementares

- Desenho
- Seleção
- Especificação de argumentos e propriedades
- Transformações
- Etc

Dispositivos virtuais

- Widgets 2D e 3D
 - Parte de sistemas de janelas, toolkit de UI ou ambiente 3D
 - Widgets – combinação de comportamento e geometria
- Motivação
 - Hardware avançado é caro e nem sempre disponível para todas as plataformas
 - A maioria dos usuários sabem como usar mouse e teclado
- Ineficiente ter que trocar de dispositivo frequentemente

Widgets 2D

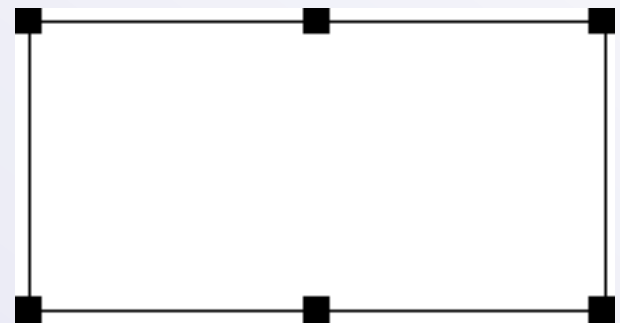
- Sistemas de janelas (e.g. X, Mac, Windows)

- window
- scrollbar



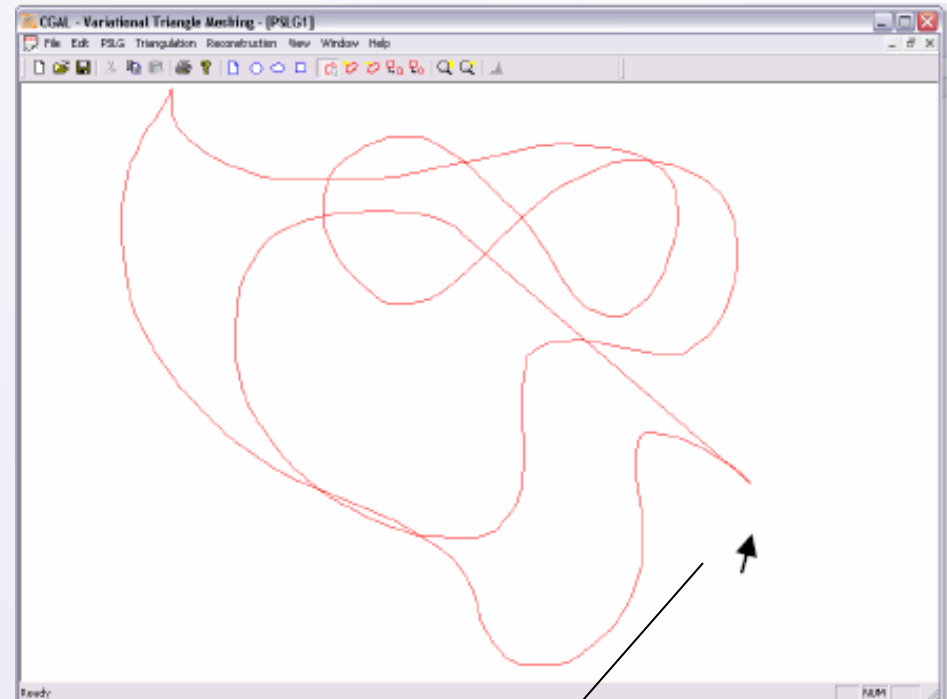
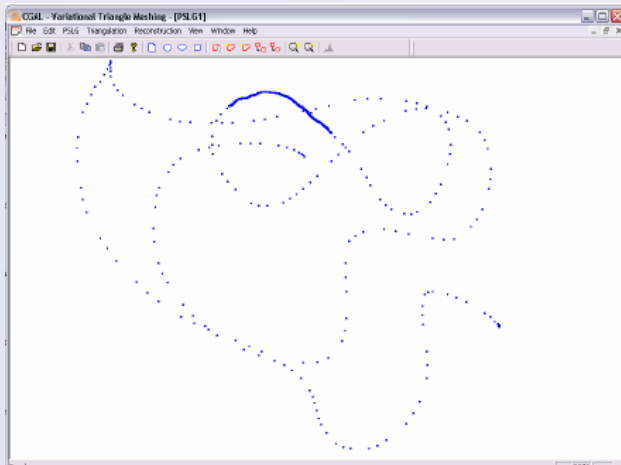
- UI toolkits (e.g. Java Swing/AWT, Motif, Windows Forms, Cocoa, Qt, WPF)

- button
- dialog box
- drawing area
- object handles



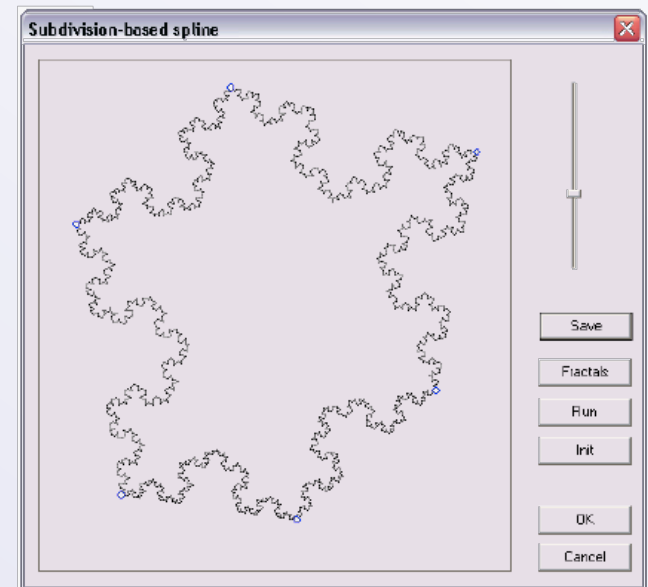
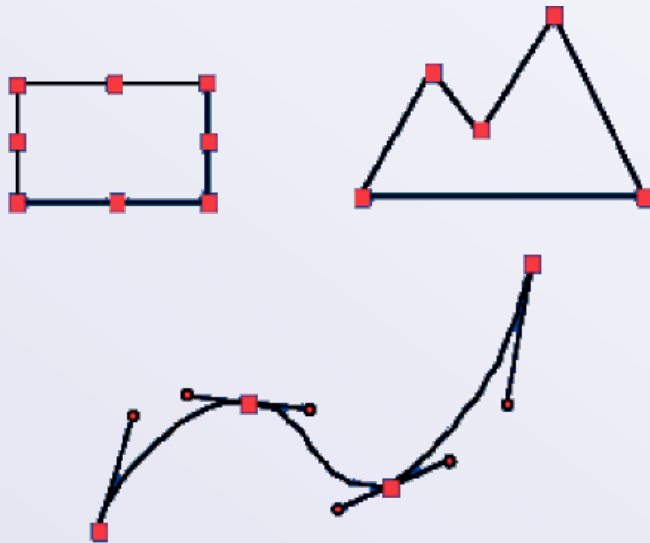
Tarefas elementares

- Desenho
 - Textos
 - Quantidades (potenciômetros)
 - Posições (pontos)
 - Traçados (amostragem)



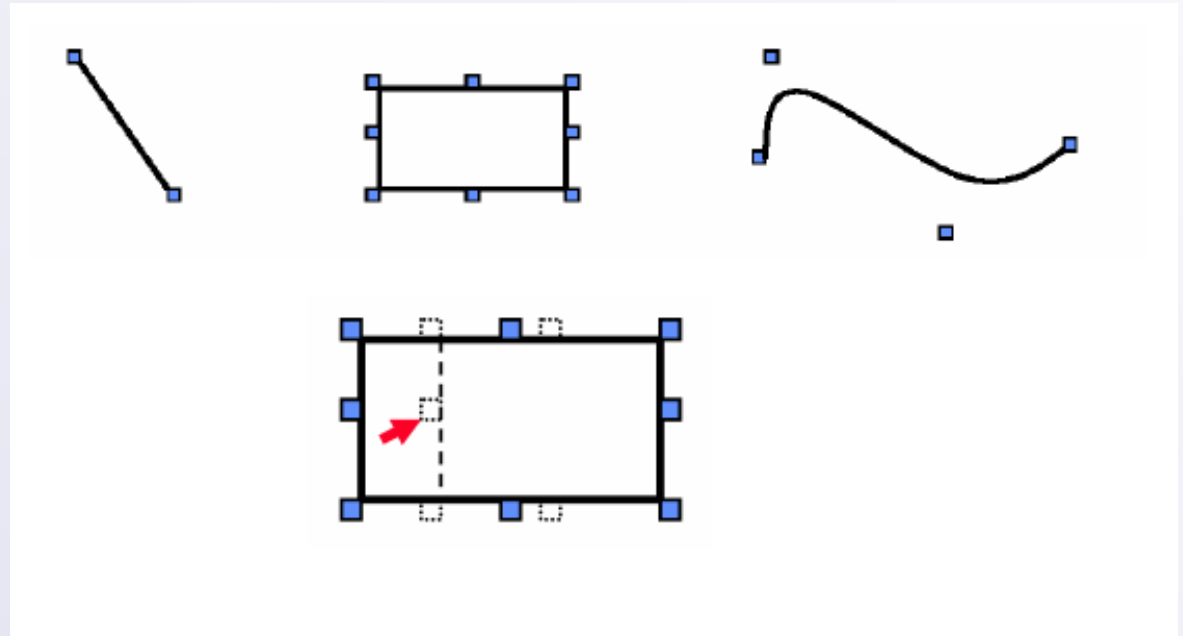
Tarefas elementares

- Transformações



Seleção 2D

- Seleção de um grupo de objetos
 - Apontador
 - Retângulo elástico
 - Laço
 - Proximidade
 - Hierárquica



Seleção 3D

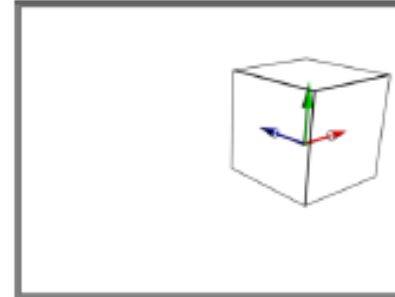
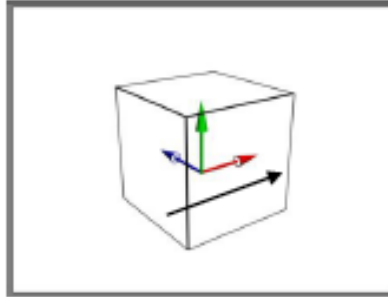
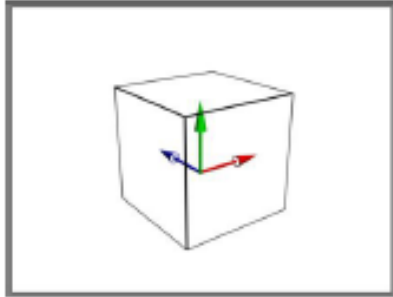
- Mais delicada:
 - Envolve transformações complexas
 - Inversa da projeção
 - Projeções em 2D (vários candidatos a seleção)

Widgets 3D

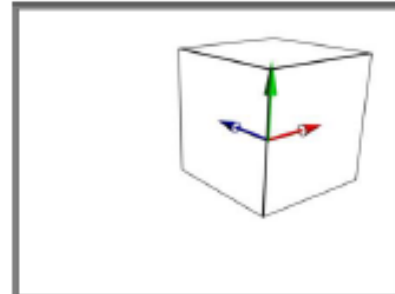
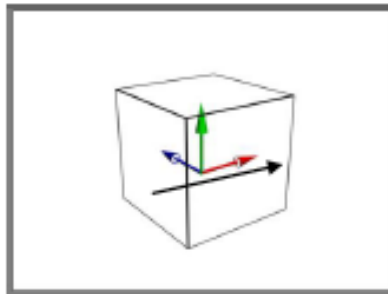
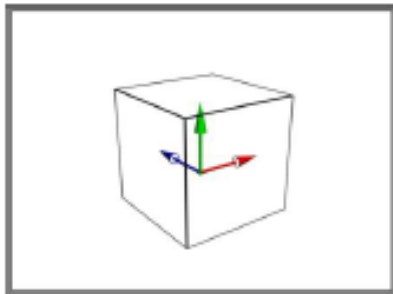
- Ambiguidade
 - Gesto em 2D vs movimento em 3D
 - Interface deve fazer importantes decisões
- Diferenças fundamentais entre 2D e 3D
 - Sistemas de coordenadas
 - Superfícies escondidas
 - Primitivas mais complexas (objetos 3D, e não janelas 2D)
- Combinar geometria & comportamento
 - Fazer com que o usuário consiga inferir a funcionalidade da widget de acordo com a sua geometria
 - Reduzir a distância cognitiva entre a função que você está realizando vs. a interação que você está fazendo.

Translação em 2D ou 3D ?

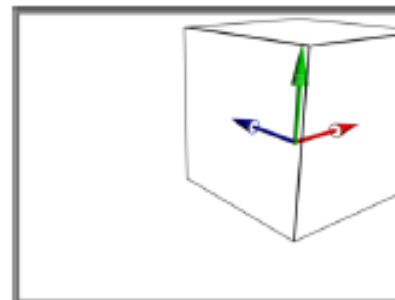
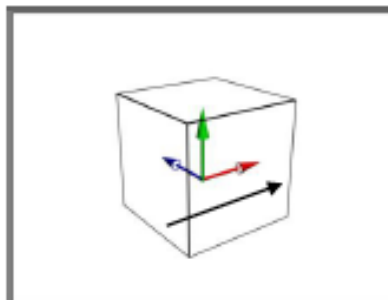
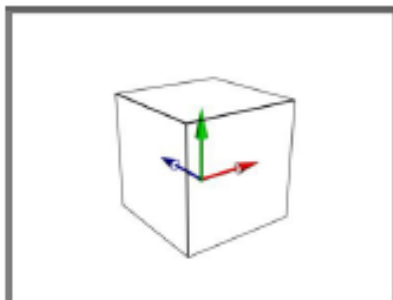
- Axis-aligned



- Screen-aligned

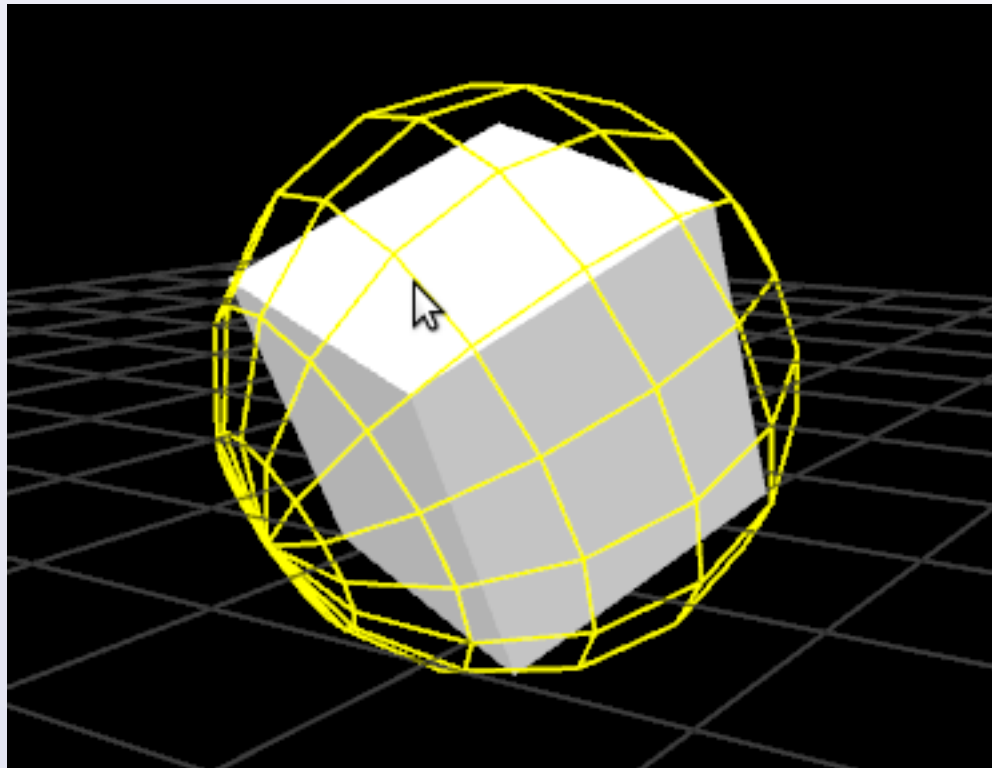


- Surface-aligned



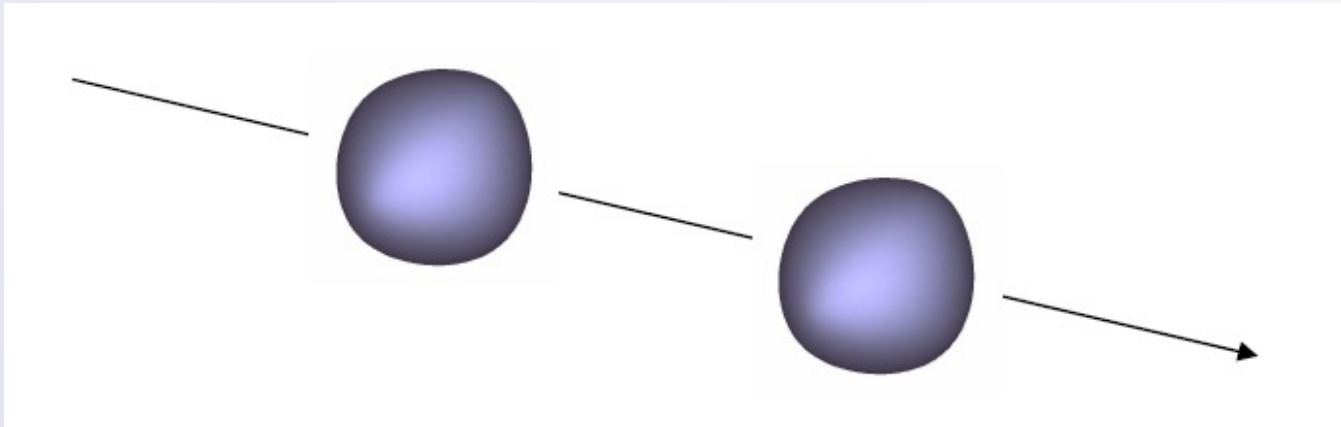
Esfera virtual (Chen'88)

- Projeta movimentos do mouse na superfície de uma esfera



Seleção 3D em OpenGL

- Noção de um buffer de seleção (*picking*)
- Especificação de um raio de seleção
- Acumulação da identificação dos objetos interceptados pelo raio



Seleção 3D em OpenGL

1. Especificação de um buffer de seleção que servirá para acumular informações dos objetos interceptados;
2. Entrar no modo de seleção;
3. Definir o raio de seleção (ex: cursor);
4. Inicializar a pilha de nomes dos objetos contidos na cena;
5. Desenhar os objetos e seus atributos de nomes colocados na pilha;
6. Sair do modo de seleção;
7. Examinar o buffer de seleção para determinar o objeto selecionado;

Seleção 3D em OpenGL

`void glSelectBuffer(GLsizei size, GLuint *buffer);`

- Retorna para cada objeto interceptado:
 - Número de identificadores na pilha de nomes quando o objeto é interceptado;
 - Os valores mínimo e máximo de z das primitivas que formam objeto
 - Os identificadores (nomes)

Seleção 3D em OpenGL

buffer [i] = número de nomes na pilha no i-ésimo hit.

buffer [i+1] = valor mínimo de z

buffer [i+2] = valor máximo de z

buffer [i+3] = nome na base da pilha

buffer [i+4] = nome seguinte na pilha (caso
buffer [i] seja maior que 1!)

Seleção 3D em OpenGL

`GLint glRenderMode (GLenum mode);`

Onde mode pode ter os valores:

- `GL_RENDER`, renderização na tela (default)
 - `GL_SELECT`, para seleção gráfica
 - `GL_FEEDBACK`, saída vetorial
-
- No modo `GL_SELECT`, cálculos da renderização são executados mas a cena não é desenhada na tela.
 - Quando retornamos ao modo `GL_RENDER` depois de deixar o modo `GL_SELECT`, o valor de retorno indica o número de objetos interceptados

Ativando o modo de seleção

```
glRenderMode(GL_RENDER);
```

```
    // desenha objetos
```

```
glRenderMode(GL_SELECT);
```

```
    // desenha objetos identificando-os com  
    nomes
```

```
nb_hits = glRenderMode(GL_RENDER);
```

```
    // Examina buffer de seleção
```


Raio de seleção

- `void gluPickMatrix (GLdouble x, GLdouble y,
GLdouble width,
GLdouble height,
GLint viewport[4]);`

`x,y`: coordenadas do centro do raio (posição do apontador)
expressas no sistema de coordenadas do viewport.

`width, height`: largura e comprimento do viewport obtido através da chamada:

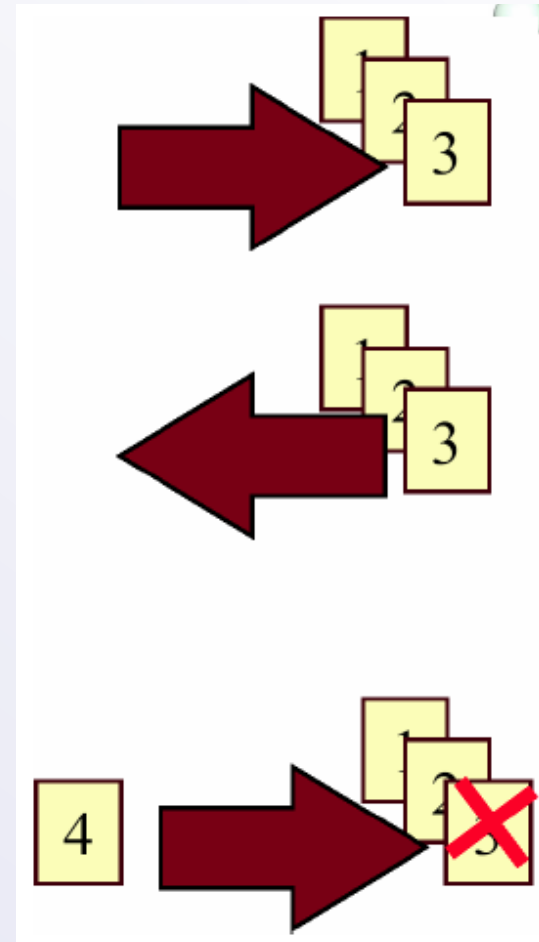
```
glGetIntegerv ( GL_VIEWPORT, (GLint *) viewport );
```

A pilha de nomes

```
void glInitNames ( void );  
void glPushName ( GLuint  
    nom );
```

```
void glPopName ( void );  
void glLoadName (GLuint  
    nom );
```

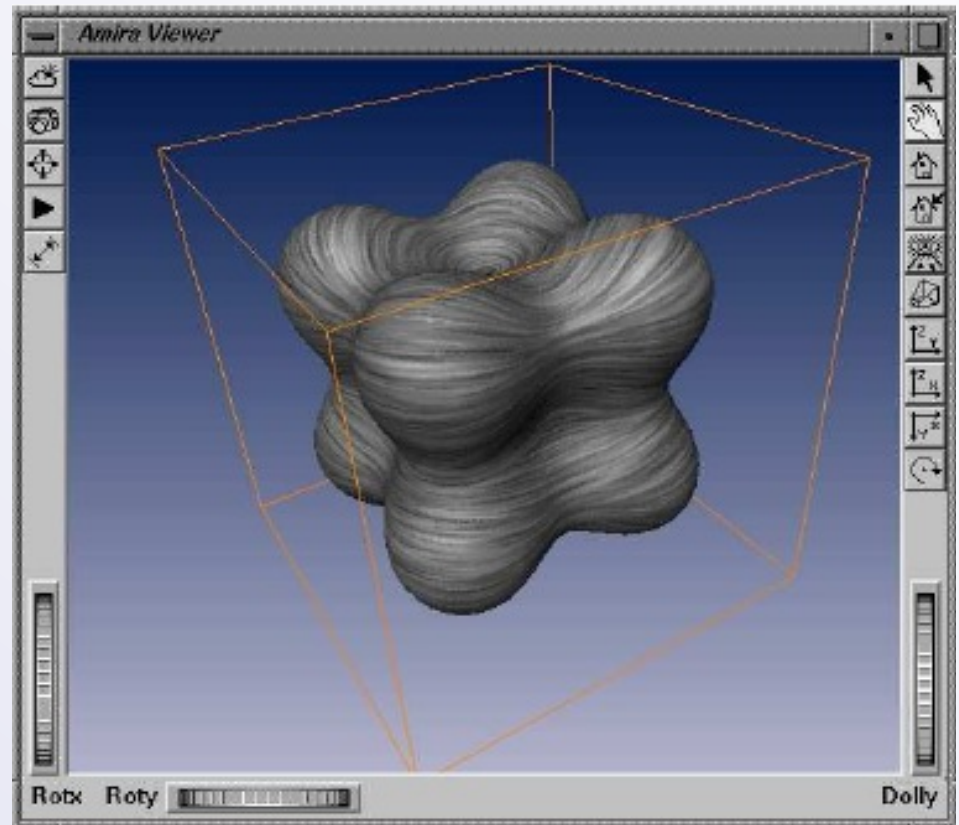
- Nota: estas chamadas são ignoradas se o modo corrente não for GL_SELECT



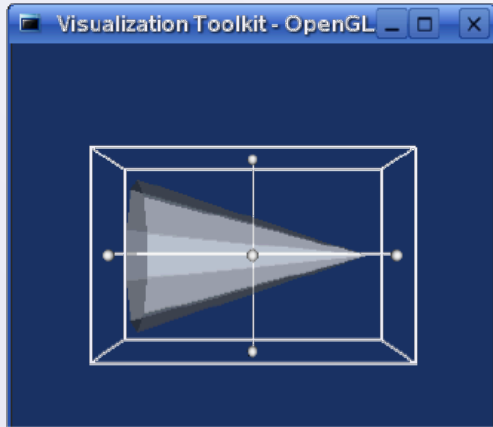
Demo Seleção

Manipulação

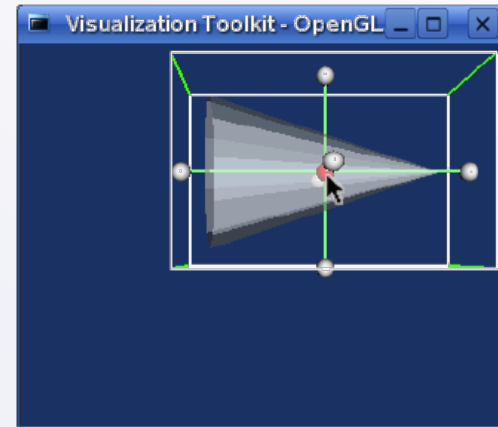
- Interação com um objeto:
 - Observador está fixo
 - Um ou mais objetos são selecionados
 - O observador manipula objeto(s) para inspeção/ modelagem, etc
- Problemas:
 - Objetos são em 3D
 - A tela é 2D
 - A interação é 2D
- Software:
 - OpenInventor
 - VTK



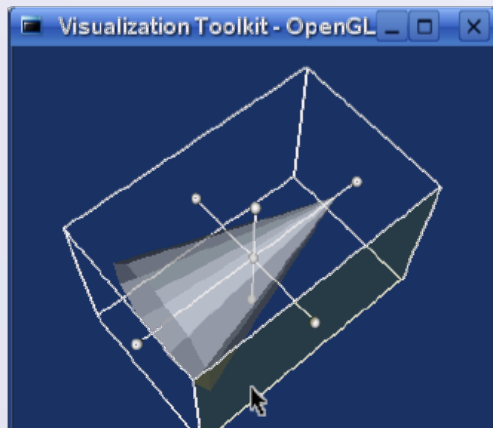
Exemplos de manipuladores



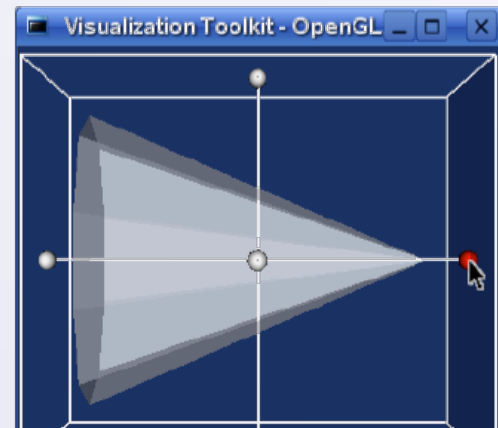
Box manipulator



Translação



Rotação



Escala

Demo MedSquare

- Colocando CG, IHC e processamento de imagens na prática
 - <http://medsquare.org>

Implementação

- Estruture-o de forma a ser facilmente mutável
 - Modularidade
 - Orientação à objetos
 - Linux: Qt, X11
 - Mac/iOS: Cocoa, Cocoa Touch
 - Windows: WPF (Windows Presentation Foundation)
 - Web: jQuery, Cappuccino Web Framework, Flex (Adobe), Silverlight (MSFT)
- Faça com que o esqueleto da próxima interface esteja disponível o mais rápido possível
 - Código de UI é difícil de escrever
 - Reuse !
 - Use Toolkits !

Avaliação da interface

Envolver usuários reais

- Novos e antigos
- Designers ou programadores não são bons testadores
 - “Funciona bem quando eu uso.”

Designers devem se fazer presentes em testes de usabilidade

- Faça com que eles não falem ou não usem as mãos!
- Devem escutar, observar, aprender e suar!

Metodologia de design experimental

- O que testar e como testar necessita de experimentação
 - Diferentes granularidades

Planeje seus investimentos

- Para produtos comerciais, uma UI pode custar tanto quanto o restante do sistema