

# **Universidade Anhembi Morumbi**

**Apostila Introdução ao Director**

Prof. Delmar Galisi Domingues

# ÍNDICE

|   | Página |
|---|--------|
| <b>O Director</b> .....                                       | 6      |
| <i>A sua metáfora</i> .....                                   | 6      |
| <b>A janela Tool Palette</b> .....                            | 7      |
| <b>Importando os “atores” do seu filme</b> .....              | 8      |
| <b>A janela Paint</b> .....                                   | 8      |
| <i>Importando imagens</i> .....                               | 10     |
| <i>Alterando uma imagem já importada</i> .....                | 10     |
| <b>Internal Cast</b> .....                                    | 11     |
| <i>Reordenando a janela Internal Cast</i> .....               | 11     |
| <b>Trabalhando com Stages, Sprites e Score</b> .....          | 12     |
| <i>Configurações do Stage</i> .....                           | 12     |
| <i>Como funciona o Score</i> .....                            | 14     |
| <i>Atributos do sprite</i> .....                              | 15     |
| <b>Keyframes e Tweenings</b> .....                            | 16     |
| <i>Modificando o movimento</i> .....                          | 16     |
| <i>Animações mais realistas</i> .....                         | 16     |
| <i>Auto Distort</i> .....                                     | 17     |
| <b>Técnicas de animação</b> .....                             | 18     |
| <i>Criando um filme em looping</i> .....                      | 18     |
| <i>Revertendo a seqüência</i> .....                           | 19     |
| <i>Animando com a ferramenta Step Recording</i> .....         | 19     |
| <i>Animando com o Real-time</i> .....                         | 19     |
| <i>Aplicando um rastro de um sprite em outro sprite</i> ..... | 19     |
| <i>Space to Time</i> .....                                    | 20     |

|   |    |
|---|----|
| <b>Efeitos ink em sprites</b> .....   | 20 |
| <b>O Texto no Director</b> .....  | 21 |
| <i>Field cast members</i> .....   | 21 |
| <i>Adicionando textos diretamente no stage com a ferramenta field</i> ..... | 22 |
| <i>Cast Members Properties</i> .....  | 22 |
| <i>Alterando a fonte de um texto</i> .....                                  | 22 |
| <i>Alterando a borda de um field</i> .....                                  | 23 |
| <i>Sombreado um texto</i> .....   | 23 |
| <i>Text cast members</i> .....  | 23 |
| <i>Importando textos</i> .....  | 24 |
| <b>Tempo</b> .....  | 24 |
| <i>Trabalhando com o Control Panel (Painel de Controle)</i> .....           | 25 |
| <i>Especificando o tempo na janela Score</i> .....                          | 26 |
| <b>Transições</b> .....   | 26 |
| <b>Palheta de Cores</b> .....   | 27 |
| <i>Criando palheta de cores</i> .....                                       | 27 |
| <i>Aplicando a palheta em seu filme</i> .....                               | 28 |
| <b>Construindo um projeto</b> .....   | 28 |
| <i>Criando um projeto</i> .....   | 29 |
| <i>Distribuição</i> .....   | 30 |
| <b>Behaviors</b> .....  | 30 |
| <i>Atribuindo behaviors em objetos</i> .....                                | 31 |
| <i>Criando behaviors, sem conhecer Lingo</i> .....                          | 31 |
| <b>O Som</b> .....  | 32 |
| <i>Sons internos X Sons externos</i> .....                                  | 32 |
| <i>Escolhendo um fomato de som</i> .....                                    | 33 |
| <i>Adicionado sons em seu filme</i> .....                                   | 34 |
| <i>Alterando a duração dos sprites de som</i> .....                         | 34 |
| <i>Ancorando o som em um frame</i> .....                                    | 34 |
| <i>Som em looping</i> .....   | 35 |
| <i>Interatividade</i> .....   | 35 |

|   |    |
|---|----|
| <b>O Vídeo</b> .....  | 35 |
| <i>Colocando um vídeo digital no stage</i> .....                      | 36 |
| <i>O que é Direct to Stage</i> .....                                  | 36 |
| <i>As outras propriedades do Vídeo</i> .....                          | 37 |
| <i>Behaviors para vídeo</i> .....                                     | 37 |
| <i>Ancorando um vídeo em um frame</i> .....                           | 38 |
| <i>Exportando filmes no formato AVI</i> .....                         | 38 |
| <br>  |    |
| <b>Lingo: criando scripts no seu filme</b> .....                      | 39 |
| <i>Os eventos</i> .....   | 38 |
| <i>Instruções</i> .....   | 41 |
| <i>Acessando scripts de objetos</i> .....                             | 43 |
| <i>Que script tem prioridade?</i> .....                               | 43 |
| <i>Criando os seus próprios scripts com um filme interativo</i> ..... | 44 |
| <i>Algumas observações a respeito da sintaxe</i> .....                | 46 |
| <i>Comandos de navegação</i> .....                                    | 46 |
| <br>  |    |
| <b>Fundamentos para montagem de scripts</b> .....                     | 48 |
| <i>Scripts</i> .....  | 48 |
| <i>Janela Script</i> .....  | 48 |
| <i>Comandos</i> .....   | 48 |
| <i>Funções</i> .....  | 48 |
| <i>Variáveis</i> .....  | 48 |
| <i>Normalização sobre variáveis</i> .....                             | 49 |
| <i>Atribuindo valores a variáveis ou strings</i> .....                | 49 |
| <i>Message Window</i> .....   | 50 |
| <b>Operadores</b> .....   | 51 |
| <i>Operadores Aritméticos</i> .....                                   | 51 |
| <i>Operadores Lógicos</i> .....                                       | 52 |
| <i>Operadores de Concatenação</i> .....                               | 52 |
| <i>Outros Operadores</i> .....  | 52 |
| <b>Propriedades</b> .....   | 53 |
| <i>Controlando sprites por Lingo</i> .....                            | 53 |
| <i>Estruturas condicionais</i> .....                                  | 54 |
| <i>Estruturas de repetição</i> .....                                  | 55 |
| <i>Funções que verificam eventos</i> .....                            | 56 |
| <i>Compatilhando scripts</i> .....                                    | 56 |
| <i>Criando sub-rotinas</i> .....                                      | 57 |
| <i>Alterando o estado do botão</i> .....                              | 58 |
| <i>Manipulando sprites através do Lingo</i> .....                     | 59 |
| <i>Posição do mouse</i> .....   | 60 |
| <i>Limitando movimentos de sprite</i> .....                           | 60 |

|   |    |
|---|----|
| <i>Manipulando o formato do cursor</i> .....  | 61 |
| <i>Código de cursores</i> .....   | 61 |
| <i>Manipulando transições através do Lingo</i> .....                                | 62 |
| <i>Códigos de Transições</i> .....  | 63 |
| <i>Manipulando textos com Lingo</i> .....   | 64 |
| <i>Propriedades de fields</i> .....   | 64 |
| <i>Atribuindo textos a fields por Lingo</i> .....                                   | 65 |
| <i>Manipulando partes do texto</i> .....  | 65 |
| <i>Mudando partes do texto</i> .....  | 65 |
| <i>Juntando textos</i> .....  | 66 |
| <i>Simulando hipertextos no Director</i> .....                                      | 67 |
| <i>Como identificar um caracter, palavra, linha ou item no corpo do texto</i> ..... | 67 |
| <i>Funções que identificam a posição do texto no field</i> .....                    | 67 |
| <i>Controlando som através de Lingo</i> .....                                       | 69 |
| <i>Parando um arquivo de som interno</i> .....                                      | 69 |
| <i>Pré-carregando um arquivo de som</i> .....                                       | 70 |
| <i>A função soundBusy</i> .....   | 70 |
| <i>Tocando arquivos externos</i> .....  | 70 |
| <i>Controlando o volume do som</i> .....  | 70 |

## O Director

Historicamente, o Director é um programa para desenvolvimento de animações em 2D. Com o advento da multimídia anos atrás, o Director passou a controlar também diversas formas de mídia, como sons ou vídeos, tornando-se poderoso o suficiente para permitir a criação de verdadeiros aplicativos multimídia. Hoje é bastante popular como ferramenta para desenvolvimento de títulos hipermultimídia como enciclopédias, jogos, softwares educativos ou de treinamento, produtos de entretenimento, tutorias, demos e até mesmo apresentações. Para completar, já há algum tempo, a Macromedia arregaçou as suas mangas para torná-lo também uma ferramenta de auxílio para o desenvolvimento de páginas Web. Mas a sua maior popularidade continua sendo a de desenvolver produtos para serem distribuídos por CD-ROM.

### *A sua metáfora*

O Director usa a metáfora de um filme. É muito importante ter esta noção para entender o modo como este software “pensa”. Assim como um filme real, o Director trabalha com a mesma noção de “quadros por segundo”. Portanto assim que você abre um arquivo desenvolvido em Director, ele começa (literalmente) a rodar, de acordo com uma velocidade estipulada. Assim como um filme roda a 24 quadros por segundo. Aliás, no Director o arquivo (ou o seu projeto) é chamado de filme (ou *movie*). A diferença é que no Director os quadros são chamados frames.

No Director o filme é dividido por frames, que são ocupados por diversos elementos, como imagens, botões, textos etc. Se um botão estiver aparecendo nos quinze primeiros frames e a velocidade de seu filme for de 15 frames por segundo (fps), este botão vai aparecer na tela por 1 segundo.

Assim como num filme real, o filme do Director tem que ter o seus atores, que em conjunto formam o elenco (cast). O Director usa este mesmo conceito para apresentar os seus “atores”, que na realidade são chamados de *cast members* (membros do elenco). Mais tarde veremos que existe uma janela chamada Internal Cast que armazena todos os “atores” de seu filme, sejam eles imagens, textos, scripts, animações, palhetas de cores, sons ou vídeos.

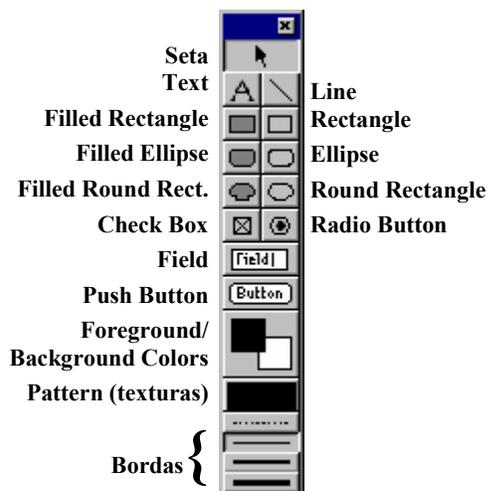
No Director, o cenário onde os atores vão atuar é chamado de stage (que por acaso quer dizer cenário em inglês). Quando um cast member (um “ator”) entra no palco ele passa a ser um sprite.

Além da janela Internal Cast, o Director tem muitas outras janelas, como, por exemplo, a janela Control Panel (Painel de Controle), uma espécie de controle de videocassete; a janela Score, uma maneira bastante conveniente de visualizar o seu filme no todo e modificar parte deles; a janela Paint, que permite criar ou alterar imagens de dentro do Director, sem precisar fazer uso de um programa de tratamento gráfico externo; a janela Message (de Mensagens) que permite executar instruções dinamicamente; a janela Tool Palette que fornece suporte para criar alguns elementos primitivos como retângulos, elipses, campos de texto, botões de pressionar ou de seleção etc. Vamos começar justamente por esta última janela.

## A janela Tool Palette

A janela Tool Palette, que pode ser acessada selecionado Window ⇒ Tool Palette (Ctrl + 7) apresenta uma série de elementos básicos de auxílio na construção de seu filme como retângulos, elipses, botões, campos de textos (texts ou fields) etc.

Para criar um objeto, basta selecionar a opção (Text, Line, Filled Rectangle, Rectangle, Filled Ellipse, Ellipse, Filled Round Rectangle, Round Rectangle, Check Box, Radio Button, Field, Push Button) arrastar no stage conforme as dimensões desejadas. A cor do objeto vai ser desenhada conforme a cor selecionada em Foreground. O estilo da borda será desenhada conforme a opção de Bordas. No caso, de Text, Check Box, Radio Button, Field e Push Button, a cor selecionada em Foreground será a cor do texto ou do rótulo do objeto.



Os objetos possuem duas cores: a cor do foreground e a cor do background. A cor do foreground é a cor que você usa para preencher a imagem. A não ser quando você “pintar” em uma área que já possui a cor de foreground. Neste caso, a cor escolhida é a cor do background. A cor de background é também a cor de fundo inicial da imagem.

A ferramenta Pattern coloca uma textura conforme a opção selecionada. A opção padrão é a de preenchimento cheio. Clique e mantenha pressionado o botão Pattern. Será aberto um menu suspenso com diversas opções de texturas. Selecione um objeto e escolha uma das texturas. O objeto será preenchido com a respectiva textura. Você pode ainda criar texturas personalizadas.

Para criar suas próprias texturas, siga os seguintes passos:

- 1 - Importe uma imagem pronta ou crie uma com a janela Paint do Director (veja próximo item). Esta imagem deve aparecer listada na janela Internal Cast.
- 2 - Em Tool Palette, mantenha pressionado o botão Patterns. Será aberto o menu suspenso que lista a biblioteca de texturas do Director.
- 3 - Clique em Tile Settings. Será aberta a janela de mesmo nome.
- 4 - Em Source, há duas opções: Cast Members e Built-in. Esta segunda opção corresponde a texturas já distribuídas pelo Director a partir da instalação. Se você quiser aplicar uma de suas imagens como textura, selecione a opção Cast Members e escolha a imagem que você deseja aplicar. Esta imagem estará ocupando o lugar de uma das texturas distribuídas pelo Director da opção Built-in. O ideal é que esta imagem tenha entre 16 e 128 pixels de altura e largura. Quando selecionar a sua imagem, pressione OK.

## Importando os “atores” do seu filme

No Director todos os “atores” de seu filme são inseridos da mesma maneira. Seja ele, uma imagem, um texto, um vídeo, uma animação, um arquivo de som, uma palheta de cores ou mesmo um outro filme feito em Director:

- 1- Clique no menu File ⇒ Import (Ctrl+R)
- 2- Selecione o tipo de mídia em Arquivos do Tipo.
- 3- Na caixa superior, selecione um ou mais arquivos desejados e clique em Add (para importar o(s) arquivo(s) selecionados) ou Add All (para importar todos os arquivos de um diretório específico).
- 4- Os botões Move Up e Move Down servem para você mudar a ordem em que eles aparecem em File List e conseqüentemente a ordem em que eles são importados. Isto representa a ordem em que eles aparecerão na janela Internal Cast.
- 5- Mantenha Standart Import em Media (veja tabela abaixo).
- 6- Clique no botão Import.

| <b>Import Media: Tipos</b>          |  |
|-------------------------------------|--|
| <b><i>Opção</i></b>                 | <b><i>Processo</i></b>   |
| Standart Import                     | A mídia é embutida no filme e pode ser editada pelo Paint.   |
| Link to External File               | A mídia é um vínculo e portanto é somente carregada no Director. Não pode ser editada, portanto. Arquivos de vídeo são obrigatoriamente deste tipo |
| Including Original Data for Editing | Como a opção anterior, mas trazendo também alguns dados internos do arquivo, como em arquivos criados em Photoshop                                 |
| Import PICT File as PICT            | Formato nativo do Director para Mac, que é tocado mais velozmente nesta plataforma, mas não no Windows.  |

## A janela Paint

Como já dissemos, a janela Paint é um mini-aplicativo que permite tratar imagens diretamente de dentro do Director. O Director não tem a pretensão de disponibilizar, com o Paint, um verdadeiro software de tratamento de imagens; muito menos concorrer com verdadeiras espécies do gênero como o PhotoShop, da Adobe. A intenção do Paint é somente fornecer soluções emergenciais. Por esta razão, não entraremos em detalhes a respeito do Paint, mas vamos fornecer algumas dicas do que há de mais importante neste tópico.

Para abrir a janela Paint, basta selecionar Window ⇒ Paint ou simplesmente pressionar Ctrl+5. Se você ativá-la com uma imagem já selecionada na janela Cast, ela é aberta com este objeto. Isto pode ser feito inclusive dando um duplo-clique na imagem da janela Cast. Se você abrir a janela Paint sem nenhuma imagem selecionada, o Paint é aberto vazio. Neste caso, você estará criando uma nova imagem a partir do zero. No menu View, você pode ativar uma régua para auxiliá-lo (View ⇒ Rulers ou Ctrl+Shift+Alt+R).

Vamos dar uma olhada em alguns dos seus botões:



navega entre cast member, que contenham imagens. Veja que ao lado há um campo para colocar o nome do cast member visualizado.



Permite criar um novo cast member a partir da janela Paint.



Botão Lasso: permite marcar uma parte da imagem, segundo rastro executado pelo mouse. A área marcada pode ser recortada, apagada, movida ou copiada. Há três opções para o Lasso: No Shrink, Lasso e See Thru Lasso. A opção Lasso (ou Shrink) seleciona a imagem, eliminando partes externas do background. A opção No Shrink seleciona exatamente a área “laçada” pelo mouse. Finalmente a opção See Thru Lasso é semelhante à opção Shrink, no entanto também elimina da seleção partes internas da imagem que apresentem a cor do background.



O botão Marquee faz a mesma coisa do que o botão Lasso, com a diferença que a área selecionada é baseada em um quadrado. Dando um duplo-clique neste botão, é selecionada a área útil completa da imagem dentro de um quadrado.



Botão Zoom – executa uma aproximação na imagem até 1:8. Para voltar ao normal, basta clicar com o botão direito do mouse e no menu suspenso escolher Zoom Out.



O botão da mãozinha permite arrastar a imagem pela tela.



Cria o ponto de registro (*registration point*) de uma imagem. Veremos adiante qual é função do *registration point*.



Pincel Brush – pinta um rastro de acordo com o padrão escolhido em Settings (acessado quando você mantém pressionado o botão Brush; a última opção do menu suspenso é Settings). Você pode criar um novo padrão, ao selecionar Custom na combobox apresentada, em vez dos padrões Standart que vêm junto com o Director.

Repare ainda que existe uma pequena caixa de seleção no canto inferior direito da janela Paint. É a lista de inks – uma espécie de efeitos para pintura. Veja a descrição de alguns deles:

*Normal*: marcar uma área da imagem com o lasso ou marquee  $\Rightarrow$  Ctrl + C  $\Rightarrow$  Selecionar o botão brush e o efeito Clipboard – o brush vai pintar segundo as cores da imagem da área de transferência.

*Gradients*: Duplo-clique no gradiente abre a janela Gradient Settings. Escolha os parâmetros deste gradiente como Direction (Top to Bottom; Bottom to Top etc), Spread (Equal; More Foreground etc) e modifique outros. Quando fechar a janela, para pintar segundo o gradiente criado, selecione o efeito Gradient na caixa Ink Options.

O efeito *Cycle* pinta um rastro com as cores da palheta.

O efeito *Darken* cria um rastro 10% mais escuro em relação às cores que já estão presentes na figura. O efeito *Lighten* faz o mesmo, só que 10% mais claro.

O efeito *Smooth* faz um rastro mais sereno, com se estivesse esfumando a imagem por onde o rastro passa. Já o efeito *Smudge* faz uma espécie de borrão. E finalmente o efeito *Smear* faz criar uma espécie de mancha de gordura.

Depois de marcada uma área, você pode copiar, recortar, apagar ou mover. Arrastando uma área selecionada em conjunto com a tecla *Alt*, você faz automaticamente uma cópia da área sem necessitar pressionar *Ctrl+C*. Você pode colar o conteúdo de uma imagem da área de transferência para um novo cast member.

### ***Importando imagens***

Já vimos como importar uma imagem no item acima “Importando os ‘atores’ do seu filme”. Se você já tentou importar uma imagem, pode ter percebido que pode aparecer uma nova janela, chamada *Image Options*. Esta janela traz uma série de opções que pergunta ao usuário como ele quer armazenar a imagem no Director. A janela apresenta estas opções:

| <b>Janela Image Options</b>        |   |
|------------------------------------|---|
| <b><i>Opções</i></b>               | <b><i>Descrição</i></b>   |
| Color Depth                        | Determina quantos bits por pixel terá a imagem  |
| Palette: Import                    | Permite que seja escolhida uma palheta de cores para ser remapeada nesta imagem, caso ela contenha mais do que 8 bits/pixel   |
| Palette: Remap To                  | Permite remapear a imagem para uma das palhetas de sistema do Director  |
| Dither                             | Se selecionada, esta opção tenta aproximar, segundo determinada técnica algorítmica, as cores da imagem pelas cores da palheta corrente, segundo uma técnica que tenta aproximar cores para dar a aparência de uma outra cor. |
| Same Settings for Remaining Images | Repete as opções escolhidas nesta janela, se mais de uma imagem foi selecionada em Import Files.  |

### ***Alterando uma imagem já importada***

A opção *Modify* ⇒ *Transform Bitmap* permite alterar a imagem já importada segundo os seguintes parâmetros:

| <b>Transform Bitmap</b> |   |
|-------------------------|---|
| <b><i>Opção</i></b>     | <b><i>Descrição</i></b>   |
| Size                    | Permite redimensionar a imagem em pixels segundo a sua largura ( <i>Width</i> ) pela sua altura ( <i>Height</i> ) |
| Scales                  | Redimensiona a imagem proporcionalmente a uma dada porcentagem.   |
| Maintain Proportions    | Se selecionada, esta opção mantém a proporção original da imagem quando você alterar o tamanho dela.              |
| Color Depth             | Determina o número de bits/pixel da   |

|                      |  |
|----------------------|--|
|                      | imagem. Se for escolhida 8 bits, aparecerá a palheta escolhida ao lado.                          |
| Palette              | Permite escolher uma das palhetas de sistema do Director ou alguma palheta já carregada no filme |
| Remap Color X Dither | Determina se esta imagem vai sofrer um processo de remapeamento com ou sem a técnica Dither.     |

## Internal Cast

É o repositório de recursos do Director. Em outras palavras, é uma coleção de *cast members*. Cast Members podem ser imagens, sons, vídeos, texto, palhetas de cores, animações e os scripts. Cada um contém os seus atributos ou propriedades, que nada mais são do que as características do respectivo cast member.

A idéia de um cast member é uma herança do conceito de *cast* do teatro ou do cinema, ou seja, é o elenco da peça teatral ou do filme. São os atores de seu filme, que estão elencados em sua janela Internal Cast e podem aparecer nas cenas que você desejar. Portanto se você mudar as características (ou mais tecnicamente falando, as propriedades) deste cast member, ele vai ser alterado em todas as cenas em que aparece.

Um cast member tem uma identificação automática assim que é criado: o seu número. No entanto, para efeitos de programação, como veremos mais adiante, é mais apropriado fornecer um nome ao cast member, algo que você pode fazer facilmente na área de texto que aparece em baixo da barra de título da janela Internal Cast.

Os botões com as flechinhas, que também aparecem logo abaixo da barra de título da janela Internal Cast, são os botões que permitem navegar entre os cast members de seu filme, selecionando respectivamente o cast member anterior ou posterior ao cast member que está selecionado no momento.

Um (ou mais) cast member pode ser copiado ou recortado e depois colado em qualquer parte da janela Internal Cast. Ele também pode ser movido para outra parte, o que aliás é uma opção melhor do que recortar e colar.

Um cast member também pode ser duplicado. Basta pressionar Ctrl+D e o cast member vai ocupar a primeira área livre da janela Internal Cast. Quando você duplica um cast member, mesmo o seu nome é duplicado. E dependendo do caso, você terá que alterá-lo.

### ***Reordenando a janela Internal Cast***

Você pode reordenar a janela Internal Cast pelo sequência Modify ⇒ Sort. (É necessário que ao menos dois cast members estejam selecionados). Quando fizer isto, você poderá reordenar os cast members segundo os seguintes critérios:

| <b>Janela Sort</b>   |   |
|----------------------|---|
| <b><i>Opções</i></b> | <b><i>Descrição</i></b>                                       |
| Usage in Score       | Reordena segundo a ordem em que eles aparecem na janela Score |

|              |   |
|--------------|---|
| Media Type   | Reordena por mídia (primeiro bitmaps, depois script, textos etc)                |
| Name         | Reordena por nome (alfanumérico). Os cast members sem nome aparecem por último. |
| Size         | Reordena por tamanho (bytes)  |
| Empty at End | Simplemente elimina os buracos, mas mantendo a ordem original.                  |

Obviamente, você pode executar reordenações sucessivas e de diferentes categorias. Por exemplo, primeiro por tipo de mídia e depois por nome. Neste caso, o Director vai primeiro reordenar os cast members por tipo de mídia e depois, por nome, sem interferir na ordem anterior.

### Trabalhando com Stages, Sprites e Score

Stage quer dizer palco. E é justamente esta a idéia que ele reflete. O stage é o palco onde ocorrem as ações. Como as ações são múltiplas e acontecem à medida que o tempo passa, o stage mostra os executando ações diferentes de acordo com o tempo. A tela em branco que aparece quando começa um novo filme no Director é o stage.

Basicamente um sprite é um cast member que foi colocado em um stage. Ele não é um objeto em si, mas um clone do cast member. Você pode colocar vários sprites em um ou diversos stages, embora o Director só acomode no máximo 120 sprites por stage.

*Dica: Quanto mais sprites em um stage, mais lento vai ser para carregar este stage.*

Quando você altera algum atributo do cast member, esta alteração se reflete automaticamente em todos os sprites. Porém, se você alterar o atributo de um sprite, isto não vai se refletir no cast member, nem nos demais sprites que foram clonados do mesmo cast member.

Cada sprite ocupa um canal de sprite. Como no máximo podem existir 120 sprites, existem 120 canais. O canal número 1 mostra o primeiro sprite colocado no stage. E é o sprite que está mais ao fundo do stage. À medida que mais sprites vão sendo colocados no stage, eles vão ocupando as camadas superiores.

Assim como no cinema, ou mais especificamente numa peça de teatro, temos um palco – que é o stage – e os seus atores – os cast members, também temos os frames. Os frames são como os quadros de um filme de cinema, que são mostrados a uma dada velocidade de, em geral, 24 quadros por segundo. Cada stage no tempo é um frame. Portanto o tempo de vida de um sprite é medido em frames. Assim, um sprite pode ocupar 10 frames, 28 frames, 100 frames etc. Tal qual um filme, cada frame pode mostrar uma parte de um cena que é ligeiramente diferente da de outra, e o que pode nos dar uma sensação de movimento.

O score é uma tabela que representa a montagem e a contagem do filme à medida que o tempo passa. Em sua janela – que é aberta através do menu Window ⇒ Score ou pressionando Ctrl + 4 – são mostrados os sprites nas linhas representadas pelos 120 canais e os frames representados pelas colunas. Quando um cast member é colocado no stage para tornar-se um sprite, o seu nome aparece representado nos frames que ele ocupa no Score.

Além de criar um sprite, arrastando um cast member para algum ponto do stage, você pode fazer o mesmo arrastando um cast member para um dos canais dos Score. Quando você usa este segundo método, o sprite ocupa o centro do stage.

Se você clicar em cima da barra azul que representa o sprite no Score, ele é selecionado no stage.

### ***Configurações do Stage***

Você pode alterar o tamanho do stage e sua posição na tela:

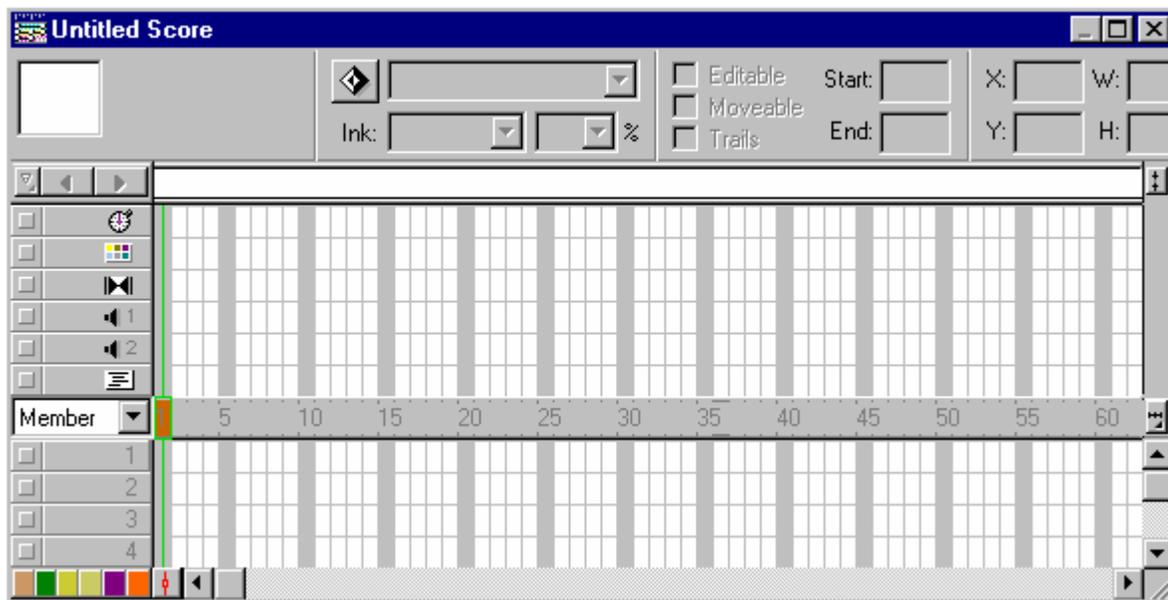
- 1 – Selecione Modify ⇒ Movie ⇒ Properties (ou pressione Ctrl + Shift + D).
- 2 – Na janela Movie Properties que foi aberta, você pode alterar vários parâmetros do seu filme:

| <b>Movie Properties</b> |  |
|-------------------------|--|
| <b><i>Opção</i></b>     | <b><i>Descrição</i></b>  |
| Stage Size              | Você define o tamanho básico do stage, segundo valores mais usuais como 640 X 480 ou 800 X 600.<br>Para definir um tamanho de stage personalizado, selecione Custom e digite os valores nos campos Width (largura) e Height (Altura).  |
| Stage Centered          | Aqui você define a posição do stage em relação à janela do Director. As opções são Centered (Centralizado), Upper Left (Canto Superior Esquerdo) e Other. Ao escolher esta opção, você deve definir uma posição personalizada nos campos Left X Top, que decresce respectivamente o valor para o afastamento em relação à esquerda e o topo do ponto inicial da janela |
| Default Palette         | Define a palheta de cores padrão e inicial para o seu filme até que seja encontrada uma nova entrada no canal palheta de cores do Score.   |
| Stage Color             | Define a cor de fundo (background) do seu stage.   |
| Remap Palettes          | Quando selecionado, o Director remapeia a palheta corrente no momento em que cast members com diferentes palhetas aparecerem no stage  |
| Allow Outdated Lingo    | Permite a você incluir comandos Lingo usados pelo Director 4.0 que já não são aceitos.   |

### Como funciona o Score

O score, como você já deve ter percebido, é uma maneira alternativa de você visualizar o que existe no seu filme.

Para a abrir a janela Score, selecione Window ⇒ Score ou pressione Ctrl + 4.



Por padrão todos os sprites inseridos no stage ocupam 28 frames, a partir do frame corrente. Para alterar o número de frames que um determinado sprite ocupa, basta clicar em cima do primeiro ou do último frame da barra do sprite na janela Score e arrastar para o tamanho desejado.

No entanto, você pode alterar este padrão de acordo com a sua conveniência. No momento que você alterar este padrão, todos os sprites inseridos a partir de então ocuparão este novo padrão, mas os anteriores manterão o número de frames que ocupavam anteriormente.

Para alterar este padrão, selecione File ⇒ Preferences ⇒ Sprite. Aparecerá a janela Sprite Preferences.

Em *Span Duration*, coloque o novo valor. Veja as opções desta janela.

| Sprite Preferences |   |
|--------------------|---|
| Opção              | Descrição   |
| Stage Selection    | Se você selecionar <i>Entire Sprite</i> , quando você selecionar um sprite no stage, você seleciona os sprites de todos os outros frames que ele ocupa. Quando você seleciona um sprite no stage com a opção <i>Current Frame Only</i> , só é selecionado o frame corrente. |
| Span Defaults      | Com <i>Display Sprite Frames</i> selecionado você ativa a opção <i>Edit Sprite Frames</i> para todos os novos sprites. Isto significa que, em   |

|               |   |
|---------------|---|
|               | para cada frame, cada novo sprite criado será tratado independentemente. Por exemplo, se você apagar um sprite na janela Score, ele será eliminado somente no frame selecionado.  |
| Span Duration | Em <i>Frames</i> , você especifica a duração padrão de cada sprite quando criado.<br><i>Width of Score Window</i> : determina a duração do sprite em função do tamanho da área visível na janela Score.<br><i>Terminate at Markers</i> : faz a duração do sprite ser definida até o primeiro marker (veremos adiante) encontrado. |

### ***Atributos do sprite***

Um sprite possui atributos adicionais em relação aos atributos próprios do cast member.

Uma maneira de visualizar seus atributos é pressionar o botão direito do mouse por cima do sprite e selecionar no menu suspenso Properties, para abrir a janela Sprite Properties.

No entanto, na própria janela Score você pode visualizar estas propriedades ou atributos. Quando você selecionar um sprite na janela Score (ou seja, você clica em cima da barra que representa o sprite), todas as características deste sprite são mostradas acima, logo abaixo da barra de título da janela Score.

Há uma pequena janelinha que mostra um preview do sprite com um símbolo representativo do tipo de objeto no canto inferior direito. Ao lado são mostrados nesta ordem: o número do sprite em relação ao canal que ocupa no Score, o número do cast member que o representa e a janela Cast de que ele foi originado.

Na lista a seguir, é apresentado o behaviour (comportamento – veremos adiante) que possui. Em ink, o efeito que apresenta e a porcentagem de cores de claridade.

Mais a frente você tem em que canais este sprite começa e termina (Start e End), sua posição na stage (X e Y) e suas dimensões (W e H).

A barra que representa o sprite no score possui um ponto inicial, chamado *head*, e representado inicialmente por uma pequena elipse no frame; um ponto final, chamado *tail*; e o seu corpo propriamente dito, chamado *body*, e que é delimitado pelo head e pelo tail. Finalmente *playback head* é o ponto que indica o frame corrente de um filme.

Outra maneira de alterar a duração de um sprite é usando a opção Extend Sprite do menu Modify. Na janela Score, selecione um frame qualquer na barra acinzentada que indicam os números dos frames. Quando você selecionar Modify ⇒ Extend Sprite, o sprite que estiver selecionado, vai ter a sua extremidade alterada para este frame.

Já a opção Tweak do menu Modify permite mover um sprite segundo uma distância específica. Selecione um sprite e pressione Modify ⇒ Tweak. Altere os valores da caixa e pressione Tweak.

*Dica: Se você inserir uma imagem que mostre um fundo branco, você pode torná-la transparente. Selecione o sprite ao mesmo tempo que pressiona CTRL e depois selecione Matte. Pronto! O fundo sumiu.*

## **Keyframes e Tweenings**

Quando você cria um sprite no stage, ele teoricamente estará parado (a não ser que um outro sprite no fundo que esteja se movimentando dê a ele uma impressão de movimento). No entanto, o Director permite a você criar no sprite um movimento que percorre um determinado caminho.

Para existir um caminho, é necessário haver um ponto inicial, um rastro deste caminho e um ponto final. Na verdade, este caminho existe, está presente assim que você cria um sprite. No entanto você não o vê, pois é como se os seus pontos iniciais e finais, por padrão, fossem coincidentes. Ou seja, isto significa, em outras palavras, que este sprite está parado.

Na realidade, o que é criado quando você desenha um sprite são pontos de ruptura no rastro, que permite a você desviar a seqüência de determinado caminho. Estes pontos de ruptura são denominados keyframes e inicialmente são coincidentes.

Para inserir um keyframe em um sprite, selecione o frame em que você deseja introduzi-lo na janela Score, e no menu Insert, selecione Keyframe. Depois, selecione o sprite no stage, e no canto superior esquerdo há um pequena elipse de cor vermelha. Clique nela e arraste para um outro ponto do stage. Pronto, você já pode ver o rastro.

Este rastro é chamado de tweenings. Quando você rodar o filme (Control ⇒ Play), você poderá ver um movimento, que reflete o desenho do tweening.

### ***Modificando o movimento***

Selecionando um sprite e pressionando Modify ⇒ Sprite ⇒ Tweening (Ctrl + Shift + B), você abre a janela Sprite Tweening, que permite configurar o movimento que você criou.

As opções mais importantes desta janela são:

*Curvature*: Define se a curvatura do rastro vai ter um formato mais linear ou curvo. As opção vão de Linear (completamente reta) a extreme (com o máximo de curvatura).

*Continuous at Endpoints*: Tenta tornar o rastro o mais contínuo possível entre os seus pontos extremos.

*Speed: Changes* – a velocidade do rastro é mais brusca

*Smooth Changes* – a velocidade é mais suave

*Easi-In/Ease-Out*: define se a velocidade vai ser incrementada do início para o fim ou vice-versa.

### ***Animações mais realistas***

Para animações se tornarem mais realistas, não basta criar um movimento em uma imagem (como o efeito tweening). É necessário que esta imagem também se altere com o tempo. Ou que uma outra imagem com pequenas alterações apareça no lugar da anterior. Esta pequena diferença é que vai dar a idéia de estar havendo uma animação. Este processo, muito comum na produção de desenhos animados, é conhecida como cel animation. É também bastante utilizada como brincadeira de crianças, quando elas desenhavam nas extremidades

dos seus cadernos pequenas figuras que vão se alterando à medida em que as páginas vão sendo folheadas.

O Director fornece uma série de ferramentas que permite a você alterar imagens. Estas ferramentas estão localizadas na janela Paint. Portanto você precisa primeiro selecionar um bitmap no cast member e depois pressionar Ctrl + 5.

Em seguida, com a imagem selecionada através do botão Marquee ou Lasso, você pode alterar a imagem com um destes botões – as ferramentas que mencionamos no parágrafo anterior:

*Flip Horizontal*: efeito de espelho horizontal

*Flip Vertical*: efeito de espelho vertical

*Rotate Left*: rotação à esquerda

*Rotate Right*: rotação à direita

*Free Rotate*: rotação livre a partir dos vértices da imagem

*Skew*: um efeito de distorção que torna a imagem oblíqua

*Warp*: altera a extensão de um dos lados

*Perspective*: aplica uma perspectiva à imagem

*Smooth*: suaviza o preenchimento da imagem

*Trace Edges*: cria uma linha a traço da imagem

*Invert*: inverte as cores da imagem. Em Imagens 8 bits, a inversão é feita para cor oposta da palheta. Em imagens 16 ou 24 bits, para o seu complemento.

*Lighten*: torna as cores mais claras

*Darken*: torna as cores mais escuras.

*Fill*: pinta a área selecionada com a cor do foreground

*Switch Colors*: altera as cores dos pixels com a cor do foreground para a cor do background

### ***Auto Distort***

O Director possui uma ferramenta que distorce uma imagem para você, gerando novas imagens que refletem cada parte desta distorção. Por exemplo, quando você chuta uma bola e ela se choca com a parede, neste momento ela sofre uma pequena distorção; na realidade um achatamento seguido de uma distensão. O Director cria justamente uma imagem para cada um dos momentos do achatamento.

Veja como fazer isto:

1 – Abra o Paint (Ctrl + 5).

2 – Crie um bitmap a partir de uma elipse, com o desenho de uma circunferência.

3 – Selecione toda a imagem com o botão Marquee (Shrink).

4 – Pressione Ctrl. Quando você faz isto você pode achatar a imagem selecionada. A partir da borda de seleção da circunferência, achate a imagem

5 – Selecione Xtras ⇒ Auto Distort. Aparece a janela correspondente.

6 – Coloque na caixa Generate o número de imagens intermediárias que você deseja que sejam criadas entre a circunferência original e a segunda que foi distorcida. Depois pressione Begin.

→ Foram criados um novo cast member para cada fase da distorção e você poderá usar posteriormente para criar um efeito de animação.

Cuidado! Todos os cast members foram criados herdando o nome do cast member de origem. Dependendo do caso, é conveniente alterar os nomes deles posteriormente.

## Técnicas de animação

O Director fornece algumas técnicas de animação que vão um pouco além da construção manual que você faz no tweening. Elas fáceis, flexíveis – pois você pode alterá-las posteriormente e precisas. Em geral são identificadas por nomes, como Film Loop, Step Recording, Real Time ou Space to Time.

### *Criando um filme em looping*

Um filme em looping (Film Loop, para o Director) é um cast member que incorpora sozinho vários outros cast members, que juntos, formam as células de uma animação. Por exemplo, você pode criar com a ferramenta Auto Distort a animação de uma bola que se choca com uma parede e sofre um processo de achatamento – veja no item anterior “Auto Distort”. As células formadas podem ser colocadas em frames seqüenciais, de forma que quando o filme for rodado, elas dêem a idéia de uma animação.

A ferramenta Film Loop faz algo mais. Ela incorpora todas os cast members em um cast member único. Este cast member não apresenta mais um ícone que identifica um bitmap, mas um ícone que identifica um Film Loop .

No entanto você não pode apagar os cast members de origem, porque na realidade um Film Loop não contém os cast members em si, mas sim apontamentos para eles.

Vamos ver o processo para se criar um Film Loop:

- 1 – Selecione na janela Cast as células que fazem parte na animação (por exemplo, a seqüência com a circunferência que se achata).
  - 2 – Na janela Score, selecione o frame a partir de onde você deseja inserir a animação.
  - 3 – Selecione Modify ⇒ Cast To Time.
  - 4 – Selecione Insert ⇒ Film Loop.
  - 5 – Dê um nome para o seu Film Loop, na janela aberta.
- Pronto. O seu filme em looping foi criado e já aparece listado na janela Cast como um novo cast member. Você pode inserir esta animação em outras partes de seu filme justamente porque ela é um novo cast member. Basta arrastar para um stage ou para um frame da janela Score. Mas repare que este cast member ocupa exatamente o número de frames correspondente ao número de células que a animação possui.

Não se esqueça: você pode apagar os sprites originais, mas nunca apague os cast members correspondentes ao Film Loop.

*Obs. 1:* Quando você arrasta um cast member do tipo Film Loop para uma frame da janela Score, a animação vai ocupar o número de frames definidos em Span Duration nas Preferências do Sprite (File ⇒ Preferences ⇒ Sprite) e não o número de células que tem naturalmente a animação.

*Obs 2.:* Quando um Film Loop é criado e você o seleciona na janela Score, toda a animação fica selecionada. Você pode querer editar apenas um sprite do seu Film Loop. Para fazer isto selecione-o e pressione Edit ⇒ Edit Sprite Frames. Assim você poderá editar somente um sprite de sua animação. Para reverter o processo, selecione Edit ⇒ Edit Entire Sprite.

### ***Revertendo a seqüência***

Você pode criar uma animação que extenda a bola, após achatá-la. O processo é o mesmo do anterior, só que antes de selecionar Modify  $\Rightarrow$  Cast To Time (passo 3), você deve selecionar Modify  $\Rightarrow$  Reverse Sequence. Em vez de criar uma animação da seqüência de 1 a n, você criará uma animação para a seqüência de n a 1.

### ***Animando com a ferramenta Step Recording***

A ferramenta Step Recording não é muito diferente do que criar uma animação criada através da manipulação direta do tweening de um sprite. A diferença é o modo de construí-la. Vamos a ela:

- 1 – Selecione um cast member e arraste para o stage.
  - 2 – Selecione na janela Score o primeiro frame deste sprite.
  - 3 – Selecione Control  $\Rightarrow$  Step Recording. (Repare que esta opção passou a estar selecionado, significando que o Step Recording está “ligado”)
  - 4 – Pressione o botão Step Forward do janela de Controle. Você caminhou para o frame seguinte.
  - 5 – Arraste o seu sprite para outro ponto do stage. Repare que o rastro da animação começou a aparecer.
  - 6 – Prossiga com os passos 4 e 5 até onde desejar.
  - 7 – Quando estiver satisfeito, selecione novamente Control  $\Rightarrow$  Step Recording para “desligar” esta ferramenta.
- Pronto sua animação está pronta.

### ***Animando com o Real-time***

Uma outra maneira de fazer uma animação é utilizando a ferramenta Real-time. Com esta ferramenta, você cria uma animação à medida que arrasta um objeto na tela. O rastro criado reflete os pontos por onde você arrastou o sprite.

Para conseguir maior controle sobre a animação, é interessante diminuir o *Tempo* do filme para que ele rode a uma taxa de frames por segundo menor.

- 1 – Na janela de controle, altere o Tempo para 5 fps.
- 2 – Arraste um cast member para o stage.
- 3 – Diminua para apenas um o número de frames por ele ocupado na janela Score.
- 4 – Selecione Control  $\Rightarrow$  Real-time. Uma borda grossa e vermelha fica em volta do sprite.
- 5 – Comece então a arrastar o sprite de forma a criar um rastro, sem soltar o botão do mouse.
- 6 – Quando estiver satisfeito, solte o botão do mouse. A animação vai refletir o rastro por onde você arrastou o sprite. Veja também como ficou o sprite na janela Score.

### ***Aplicando um rastro de um sprite em outro sprite***

É possível aplicar o mesmo rastro que um sprite possui em um outro de modo a eles possuírem tweenings coincidentes.

- 1 – Na janela Score, selecione o sprite que contenha o tweening que você deseja copiar para outro sprite.
- 2 – Pressione Ctrl+C para copiar o sprite para a Área de Transferência.
- 3 – Selecione um frame livre e pressione Ctrl + V. Mantenha o sprite selecionado.

- 4 – Na janela Cast, selecione o cast member que você deseja que receba o rastro.
- 5 – Selecione Edit ⇒ Exchange Cast Members.
- 6 – Teste, pressionando Play na janela de controle.

### ***Space to Time***

Outra técnica de animação bastante útil e muito simples de implementar é através da ferramenta Space to Time. Com ela, você desenha em apenas um frame, o caminho que um ou mais sprites percorreriam. Veja este exemplo:

- 1 – Em File ⇒ Preferences ⇒ Sprite, altere Span Duration para 1.
- 2 – Desenhe uma elipse. Ela se torna um cast member.
- 3 – Arraste o cast member da elipse várias vezes para o mesmo frame, de forma a criar vários sprites da mesma elipse. Altere o tamanho ou cor de algumas delas. Repare que as elipses formam uma espécie de caminho entre elas.
- 4 – Selecione todas (Ctrl+A).
- 5 – Selecione Modify ⇒ Space to Time. Será aberta uma janela de mesmo nome.
- 6 – Escolha um valor, por exemplo, 5. Este número corresponde ao número de pontos existentes entre cada keyframe. Cada elipse será um keyframe. Repare que as elipses que ocupavam o mesmo frame, ocuparam cada uma um frame. Por isto, esta técnica é chamada Space to Time.

### **Efeitos ink em sprites**

Assim como você aplicar efeitos ink em cast members, você pode aplicar estes mesmos efeitos diretamente em sprites. Basta selecionar o sprite e na janela Scores escolher o efeito na lista Ink.

O efeito escolhido vai interferir na execução do filme, pois alguns efeitos requerem mais memória para serem renderizados.

| <b>Efeitos inks</b>    |  |
|------------------------|--|
| <b><i>Nome</i></b>     | <b><i>Descrição</i></b>  |
| Copy                   | É o padrão e o mais rápido efeito. Cria um quadrado branco em volta da imagem  |
| Background Transparent | Os pixels do sprite que possuírem a mesma cor do background ficaram transparente   |
| Transparent            | Pixels brancos tornam-se transparentes e os pretos mantêm-se pretos. As outras cores criam um efeito de transparência                    |
| Reverse                | Pixels brancos tornam-se transparentes e os pretos mostram a cor complementar do pixel que está além dele. Mais útil para sprites 1-bit. |
| Matte                  | Se o fundo for branco, torna transparente este quadrado delimitado por ele   |
| Ghost                  | Os pixels pretos tornam-se brancos e os brancos tornam-se transparentes. Mais útil para imagens 1-bit.                                   |
| Not Copy               | Como o efeito Copy, só que cada pixel é mostrado com a cor complementar do gráfico original  |
| Not Transparent        | Cada pixel branco é mapeado como transparente, enquanto cada pixel preto torna-se transparente. Mais útil para imagens 1-bit.            |
| Not Reverse            | Mapeia cada pixel branco para o complementar do cor do background, e   |

|              |  |
|--------------|--|
|              | cada pixel preto torna-se transparente. Mais útil para imagens 1-bit.  |
| Not Ghost    | O complementar do efeito Ghost. Brancos tornam-se pretos e pretos, transparentes   |
| Mask         | Se o cast member imediatamente posterior for 1-bit, o sprite aparece somente onde o há pixels pretos.                    |
| Blend        | Garante que o sprite use a porcentagem blend especificada na janela Sprite Properties                                    |
| Add Pin      | Soma o valor da cor de cada pixel à cor do background, tornando-o mais suave   |
| Add          | Faz a soma da cor do foreground com a cor do background. Se algum pixel excede os 100%, a cor mais escura é substituída. |
| Subtract Pin | Como Add Pin, mas subtraindo. O sprite fica mais escuro em geral.  |
| Lightest     | Compara o valor da cor dos pixels do foreground e do background e retorna a mais suave das duas.                         |
| Subtract     | Como Add, mas subtraindo os valores  |
| Darkest      | Compara o valor da cor dos pixels do foreground e do background e retorna a mais escura das duas.                        |

## O Texto no Director

O Director especifica três elementos para criar textos:

- *Text cast members*: que serão explicados logo adiante e se caracteriza por ocupar uma significativa quantidade de memória e por não poder se alterado no modo de execução; somente no modo de desenvolvimento. Também não são a melhor escolha quando você deseja animar uma seqüência de textos. É a opção natural para grandes corpos de texto que aparecem na tela apresentando conteúdos temáticos. Na realidade, no modo de execução, estes textos são rodados como se fossem bitmaps, o que resolve o problema das fontes.
- *Field cast members*: embora o aspecto visual não seja o seu forte, este elemento ocupa muito menos memória e seu texto pode ser editado pelo usuário. É a melhor opção para campos de textos em formulários ou quando a idéia é criar algum tipo de interatividade em resposta ao seu conteúdo. Falaremos dele em capítulos posteriores.
- *Painted text*: é na realidade uma das ferramentas da janela Paint (que já vimos anteriormente). O seu grande inconveniente é que o texto criado é transformado em um bitmap e não pode ser alterado depois de desenvolvido.

### *Field cast members*

Os textos podem ser incluídos basicamente de três maneiras:

- Adicionando textos diretamente no stage
- Entrando com texto na janela Text
- Importando textos

### ***Adicionando textos diretamente no stage com a ferramenta field***

Você pode adicionar textos diretamente no stage através da ferramenta Text da janela Tool Palette  
(Para abri-la: Window ⇒ Tool Palette; ou Ctrl + 7):

- 1 – Com Tool Palette aberto selecione o botão Field
- 2 – Clique e arraste na tela. O campo é formado.
- 3 – O ponto de inserção do field estará piscando automaticamente. Entre com o texto.

#### ***Cast Member Properties***

Pressionando Modify ⇒ Cast Member ⇒ Properties, você acessa a janela que permite mudar as suas propriedades:

O primeiro campo indica o nome do cast member

Em Framing, você tem quatro opções:

*Adjust to Fit*: expande o campo verticalmente à medida que o texto vai sendo introduzido.

*Scrolling*: introduz uma barra de rolagem no field.

*Fixed*: não expande o field à medida que o texto vai sendo introduzido. Parte do texto fica escondido se passar dos limites do field.

*Limit do Field Size*: o texto é limitado ao tamanho do field.

Em Options, há três opções:

*Editable*: torna o field editável no modo de execução

*Word Wrap*: faz o texto automaticamente cair para a linha de baixo, assim que os limites do field são atingidos. Caso contrário, o texto é truncado e é necessário usar a tecla Enter para cair para a linha de baixo.

*Tab to Next Field*: ativa a opção de alternar entre fields com a tecla Tab.

A opção Unload controla como o Director remove o cast member da memória se esta estiver baixa. As opções variam de 3-Normal (o cast member vai ser removido da memória sempre que necessário) a 0-Never (o cast member mantém-se sempre na memória)

#### ***Alterando a fonte de um texto***

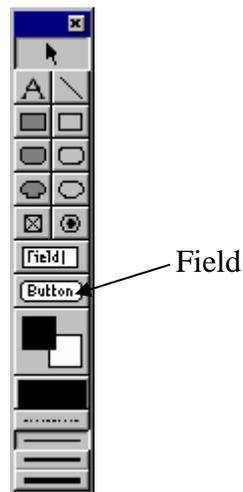
Para alterar a fonte de um field:

- 1 – Selecione o field.
- 2 – Pressione Modify ⇒ Font (Ctrl + Shift + T).
- 3 – Na janela Font, escolha a fonte desejada, seu tamanho (em Size), bem como suas características estilísticas (Italic, Bold, Underline) e sua Cor (em Color). Pressione OK.

Para alterar a fonte de somente uma parte do texto:

- 1 – Dê um duplo-clique no field e selecione somente o texto, palavra, ou parágrafo desejado.
- 2 – Siga os passos 2 e 3 do item anterior.

A janela Tool:



***Alterando a borda de um field***

- 1 – Selecione o field.
- 2 – Pressione Modify ⇒ Borders ⇒ Line e escolha a opção desejada.

***Sombreado um texto***

- 1 – Selecione o field.
- 2 – Pressione Modify ⇒ Borders ⇒ Text Shadow e escolha a opção desejada.

***Text cast members***

A janela Text na realidade é um mini-processador de textos como o Word. No entanto, ao contrário deste aplicativo da Microsoft, a janela Text não cria documentos com múltiplas páginas, mas é a melhor opção no Director para criar grandes blocos de texto.

Como o Director usa o formato RTF (Rich Text File) para textos internos, você pode tranquilamente formatar o seu texto como faria normalmente em um processador de textos comum. Assim, quando este texto for introduzido no stage, ele vai aparecer com todas as opções de formatação que você atribuiu ao seu texto.

Para abrir a janela Text, pressione Insert ⇒ Media Element ⇒ Text ou pressione Window ⇒ Text (Ctrl + 6). (Você criar um campo para text cast members também com Tool Palette – Ctrl+7, selecionar o botão Text , clicar e arrastar direto no stage). Para inserir o seu texto no stage:

- 1 – Escreva o seu texto normalmente e use à vontade todas as opções de formatação apresentadas no botões da janela, como o botão para negrito, para itálico etc.
- 2 – Quando terminar o seu texto, feche a janela. O seu texto se tornará um cast member. O nome para este cast member é “Text” e não field.
- 3 – Arraste o cast member de texto que você criou para a janela Score ou diretamente no stage.

Repare que o *text* apareceu ajustado aos limites da caixa de texto. Você pode alterar este atributo normalmente, assim como outras opções. Selecione o *text* e pressione Modify ⇒ Cast Member ⇒ Properties (Ctrl + I). É aberta a janela Text Cast Member Properties. Repare que esta janela tem muitas semelhanças com a janela Field Cast Member Properties. No entanto, há algumas diferenças. Vamos a elas:

Em Framing, você tem somente três opções

*Adjust to Fit*: expande o text verticalmente à medida que o texto vai sendo introduzido.

*Scrolling*: introduz uma barra de rolagem no text.

*Cropped*: permite aumentar o tamanho da caixa do text, mesmo que o texto que apareça seja muito pequeno.

*Obs.:* O Que é Anti-Alias? Os textos com fontes muito grandes (mais ou menos a partir de 20 pontos) geralmente aparecem na tela serrilhados, o que é uma opção estética desagradável. O atributo Anti-Alias tenta resolver este problema ao introduzir alguns pixels acinzentados nos limites da fonte, que suavizam a visualização. No entanto, esta característica terá um efeito melhor se o fundo da tela for branco.

Em Anti-Alias, há três opções:

*All Text*: atribui esta característica em todo o texto do text.

*Large Than*: especifica o anti-alias somente a textos com fontes maiores do que as especificadas no campo que aparece em frente.

*None*: não especifica o atributo anti-alias a nenhum texto.

A opção Unload controla como o Director remove o cast member da memória se esta estiver baixa. As opções variam de 3-Normal (o cast member vai ser removido da memória sempre que necessário) a 0-Never (o cast member mantém-se sempre na memória)

### ***Importando textos***

Uma maneira de introduzir textos no seu filme é importando textos de arquivos externos. Esta opção é bastante apropriada quando você deseja introduzir um texto que você já digitou em outros processadores de texto como o Microsoft Word e não deseja ter que digitá-los novamente.

Para que a formatação do texto original seja mantida, é necessário que o texto seja salvo com o formato RTF (Rich Text File). Se o processador que você usa possui suporte para este formato, basta usar a opção Salvar como de seu programa e escolher esta opção.

Para inserir este texto no Director, basta pressionar File ⇒ Import (Ctrl + R) e selecionar o arquivo de texto externo (seja ele RTF ou TXT). Assim que ele é inserido, ele se torna um text cast member.

## **Tempo**

Dissemos, na introdução, que um projeto criado no Director trabalha conforme a estrutura de um filme ou de uma animação convencional. Cada tela representa um frame. No momento que o filme começa a ser rodado, os frames são mostrados de acordo com uma certa velocidade estabelecida. Por exemplo, 15 frames por segundo (fps), ou 30 fps, ou 60 fps. Como cada frame apresenta uma composição visual diferente, para nossos olhos, há a sensação de estarmos assistindo a uma animação. Quanto menor a velocidade, menor a sensação de realidade que um filme apresenta. Um filme que roda a uma taxa de 2 ou 3 frames por segundo parece estar rodando em câmera lenta.

Por outro lado, embora seja uma tentação fazer um filme rodar a velocidades muito altas – porque teoricamente teríamos uma aproximação cada vez maior da realidade – esta é uma tentativa inútil. Primeiro, porque a partir de uma certa taxa, nossos olhos não conseguem mais perceber a diferença existente entre uma velocidade alta ou uma mais alta ainda – a sensação é a mesma. Um filme a 60 fps causa-nos praticamente a mesma sensação de um filme a 90 fps. Em segundo lugar, porque a grande média dos computadores atuais não consegue processar as imagens que são mostradas nos frames a velocidades tão altas. A média dos computadores domésticos só consegue *renderizar* imagens a uma taxa que varia entre 10 e 15 fps. Assim, é esta a velocidade ideal de trabalho.

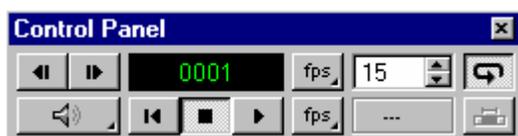
No Director, na realidade, quando você escolhe a velocidade do filme, você está determinando o seu valor máximo. Se, por exemplo, você especificar a velocidade a 100 fps, o Director vai tentar rodar o filme a esta taxa, mas se não conseguir, vai escolher um

valor menor e mais apropriado à configuração do seu computador e que também depende da complexidade de seu filme. Stages com imagens muito pesadas, scripts complexos e outros recursos pesados, como animações, tornam o carregamento do frame mais lento. Isto vai causar flutuações na velocidade que o Director vai tentar operar.

Você pode alterar a taxa de fps de três maneiras:

- Pela janela Control Panel
- Pela janela Score
- Por programação

### ***Trabalhando com o Control Panel (Painel de Controle)***



O Painel de Controle permite controlar o andamento do filme

| <b>Painel de Controle</b> |   |
|---------------------------|---|
| <b><i>Controle</i></b>    | <b><i>Descrição</i></b>   |
|                           | Volta um frame  |
|                           | Avança um frame   |
|                           | Mostra o frame corrente   |
|                           | Permite selecionar a taxa entre frames por segundo (fps) ou segundos por frame.   |
|                           | Estabelece a velocidade. Você pode alternar os valores através das flechinhas ou inserindo o número diretamente na caixa de texto.  |
|                           | Selecionado o primeiro ícone, o filme roda em <i>looping</i> . Selecionado o segundo ícone, o filme roda até o fim, mas pára no último frame.                               |
|                           | Controle de Volume. Clicando, abre um menu suspenso com as opções: de 0-mudo a 7.   |
|                           | Volta o filme para o primeiro frame.  |
|                           | Pára o filme.   |
|                           | Roda o filme.   |
|                           | Há dois estados. Se o botão estiver pressionado, só serão mostrados os frames selecionados na janela Score, quando o filme tocar. Caso contrário, todo o filme será tocado. |

### ***Especificando o tempo na janela Score***

Um dos canais da janela Score, aquele que tem o formato de um relógio, é o canal Tempo. Neste canal você pode alterar a velocidade (em fps) de seu filme na entrada de cada frame. Assim, por exemplo, se o frame 1 tiver uma taxa de 15 fps e o frame 16 a uma taxa de 1 fps, o filme vai rodar 15 frames no primeiro segundo e quando chegar no frame 16 vai ficar parado lá por um segundo.

Além de determinar a taxa máxima de fps do seu filme, no canal Tempo da janela Score você pode especificar uma pausa, dizer para o Director esperar até haver uma interação do usuário e esperar por um determinado ponto de um vídeo, som ou animação para continuar rodando.

Para alterar o funcionamento do tempo de seu filme:

1- Abra a janela Score, se ela não estiver aberta (Window ⇒ Score) e clique no botão Hide/Show Effects Channels  para expandir a janela Score, de modo a você poder visualizar o canal Tempo .

2 – Selecione o frame desejado e clique Modify ⇒ Frame ⇒ Tempo. Ou dê um duplo-clique direto no frame. Vai ser aberta a janela Frame Properties: Tempo. As opções desta janela são:

| <b>Frame Properties: Tempo</b>    |   |
|-----------------------------------|---|
| <b><i>Opção</i></b>               | <b><i>Descrição</i></b>   |
| Tempo                             | É a opção padrão. Aqui você pode especificar mudanças de velocidade na entrada de cada frame. Em cada frame que você altera, o número escolhido para a taxa é mostrado em cima do frame do canal Tempo. Você pode inclusive copiar e colar os parâmetros escolhidos de um frame para outro. |
| Wait                              | Estabelece uma pausa no frame corrente com o tempo especificado.  |
| Wait for Mouse Click ou Key Press | Estabelece uma pausa no frame especificado e o filme fica parado até que o usuário clique o botão em algum lugar da tela ou pressione uma tecla qualquer.   |
| Wait for Cue Point                | Estabelece uma pausa no filme até o momento que um som ou vídeo, selecionado em Channel, termine de tocar.  |

*Cuidado:* Se você escolher a opção Wait for Mouse Click ou Key Press, o filme começará a tocar assim que você clicar em qualquer lugar da tela. Se você quiser que o filme só volte a rodar quando clicar um botão determinado, você terá que usar recursos de programação.

### **Transições**

Transições são efeitos visuais que ocorrem entre dois frames. Os efeitos ocorrem com uma determinada duração, mas são independentes da velocidade que você especificou em Tempo (o relógio pára até a transição terminar). Assim como o Tempo, as transições possuem um canal específico  na janela Score: o canal Transition.

Quando você deseja colocar uma transição entre dois cenas (uma em cada frame), você deve especificá-la no frame correspondente ao primeiro frame da segunda cena. Não no último frame da primeira cena. Uma vez criada uma transição, ela é colocada na janela Cast

como um cast member. Caso você queira usá-la novamente entre outras duas cenas, basta arrastar esta transição da janela Cast para o frame correspondente no canal Transition.

Para colocar uma transição entre duas cenas, dê um duplo-clique no primeiro frame da segunda cena na canal Transition da janela Score. A janela Frame Properties: Transition aparecerá, com as seguintes opções:

| <b>Frame Properties: Transition</b> |  |
|-------------------------------------|--|
| <b>Opções</b>                       | <b>Descrição</b>   |
| Categories / Transitions            | O Director divide as transições em categorias, que são apresentadas na caixa de texto Categories. A transição é na realidade escolhida na caixa Transitions.                       |
| Duration                            | Indica a duração aproximada da transição completa em segundos.   |
| Smoothness                          | Na realidade indica o número de pixels envolvido num instante de tempo. Quanto mais à direita, menor o número de pixels.   |
| Affects:                            | <i>Entire Stage:</i> A transição é efetuada em todo o stage.<br><i>Changes Area Only:</i> A transição só afeta aquelas áreas do stage que possuem diferenças entre os dois frames. |

## **Palheta de Cores**

Quando você importa uma imagem, o Director permite que você remapeie esta imagem para uma das palhetas carregadas no filme. Se você importar a imagem sem remapeá-la a uma das palhetas já presentes no filme, o Director importa a imagem com a palheta de cores que ela possui e transforma esta palheta em um novo cast member. Mas todo cuidado é pouco.

Um das vantagens no Director, com já vimos, é que ele permite remapear uma imagem, assim que ela é importada, para uma determinada palheta de cores escolhida por você. O Director já possui um certo número de palhetas de cores padrão. No entanto você pode criar a sua própria palheta de cores.

Você ainda pode posteriormente alterar a palheta de cores de uma imagem já importada, seguindo estes passos:

- 1- Selecione na Cast Window, uma imagem.
- 2- Selecione Modify ⇒ Transform Bitmap, para mostrar a janela correspondente.
- 3- Selecione a palheta de cores que você deseja na lista “Palette”.
- 4- Clique OK.

A sua imagem estará remapeada para esta nova palheta de cores.

### ***Criando palheta de cores***

O Director possui uma ferramenta que possibilita criar ou alterar palhetas de cores. (Você não pode alterar as palhetas de cores padrão). Para acionar esta ferramenta, siga estes passos:

- 1 – Selecione Window ⇒ Color Palettes ou pressione Ctrl+Alt+7.
- 2 – Em princípio vai aparecer selecionada a palheta de cores padrão Windows (System – Win). Você pode alterar as cores de palheta, selecionando uma das cores e aumentando ou diminuindo os valores referentes a H (tom), S (saturação) e B (brilho). Como esta palheta

não pode ser alterada, pois é uma palheta padrão, qualquer alteração que você tentar fazer vai forçar a abertura da janela Create Palette. Nela, você especifica, em “Name”, o nome desta nova palheta que está sendo criada.

3 – Quando você fizer todas as alterações que desejar, clique OK para terminar.

Depois de criada, a sua palheta já vai aparecer listada nas palhetas de seu filme e você vai poder usá-la como se fosse qualquer outra.

### ***Aplicando a palheta em seu filme***

Em princípio a palheta de cores de seu filme é a System–Win, mas você pode mudar isto:

1 - Pressione Modify ⇒ Moveis ⇒ Properties

2 – Na janela Movie Properties, escolha uma nova palheta de cores para o seu filme na caixa de seleção Default Palette.

A partir desta alteração a palheta de cores padrão do seu filme é esta que foi escolhida. No entanto ela não tem que ser necessariamente a palheta de cores de seu filme durante todo o tempo. Existe um canal para palhetas de cores no Score, especificado com este símbolo , que permite que você altere a palheta de seu filme dinamicamente. Portanto, se em determinado momento, você quiser que o seu filme passe a ter uma outra palheta de cores como padrão, basta atribuir uma nova palheta ao frame que deseja no respectivo canal do Score. Vejamos:

1 – Abra a janela Score. Se o canal de palheta de cores não estiver aparecendo, clique no botão Hide/Show Effects Chanel  no Score para que você visualize os canais de efeito desta janela.

2 – No canal palheta de cores , clique no frame onde você deseja atribuir a nova palheta de cores e selecione Modify ⇒ Frame ⇒ Palette. Será aberta a respectiva janela. (Você pode também abrir esta janela dando um duplo-clique no frame correspondente).

3 – Escolha a palheta de cores que deseja na caixa de seleção Palette. A partir deste frame, quando o filme estiver rodando, a palheta de cores padrão do seu filme vai ser esta escolhida, até que entre uma nova palheta seja solicitada novamente.

## **Construindo um projeto**

Quando você terminou o seu filme, você provavelmente vai querer que ele seja visto por outras pessoas. Você tem duas alternativas: entregar o seu filme para alguém que já tenha o Director em seu computador ou criar um projeto.

Um projeto é um filme já um formato executável. Ou seja, se um usuário não tiver o Director instalado no computador dele, ele não poderá visualizar o seu filme, porque este arquivo em si não é um arquivo executável. Nenhum dos tocadores de mídia disponíveis, como o Media Player do Windows, rodam um filme feito em Director. A Alternativa é criar um projeto, que é um filme empacotado com todos os elementos necessários para que ele rode sozinho. Neste sentido, o projeto sempre vai ser um arquivo maior do que o filme.

Para efeitos de terminologia, a partir de agora vamos chamar filme do arquivo de desenvolvimento, que portanto precisa do Director para rodar. E de projeto do arquivo já pronto para distribuição. No Windows, quando você cria um projeto, o arquivo passa a ter a extensão EXE; ao contrário do filme, que tem a extensão DIR. No Macintosh, os arquivos

não tem extensão, portanto você tem que ter a preocupação de salvar o projeto com um nome diferente do filme. Para criar um projeto de um filme feito no Windows, você tem que fazê-lo no Windows. O mesmo para o filme feito no Mac.

### ***Criando um projeto***

Uma coisa deve ficar clara de antemão. Quando um projeto é criado, ele não pode ser editado. Se quiser fazer alguma alteração, você vai ter que fazê-lo a partir do filme e gerar o projeto novamente.

Quando você cria o projeto, o Director pergunta se você deseja adicionar outros filmes associados a este projeto. Você pode também adicionar cast libraries, Xtras, ou outros arquivos necessários para rodá-lo convenientemente. Você pode criar até um projeto que rode vários filmes em seqüência. Para criar um projeto, siga estes passos:

- 1 – Abra o seu filme, se ele não estiver carregado no Director.
- 2 – Selecione File ⇒ Create Projector. Será aberta a janela de mesmo nome.
- 3 – O seu filme vai aparecer listado na grande caixa. Selecione-o e pressione o botão Add para incluí-lo na caixa File List.
- 4 – Se você for incluir um ou mais filmes a este projeto, selecione-os da mesma maneira com o auxílio da caixa de seleção Look in. Mas cuidado. A ordem em que os filmes aparecem listados em File List é significativa: é exatamente a ordem em que eles aparecerão em seqüência.
- 5 – Clique no botão Options para especificar alguns parâmetros importantes. Depois clique OK. (Veja na tabela abaixo as especificações da janela Projector Options).
- 6 – De volta à janela Create Projector, pressione Create. Aparecerá a janela Save Projector As.
- 7 – Escolha um nome para o seu projeto e pressione OK.

Com o projeto já criado, feche o Director e dê um duplo-clique no arquivo de projeto criado. Pronto, o seu projeto está pronto e rodando!

| <b>Projector Options</b>           |   |
|------------------------------------|---|
| <b><i>Opções</i></b>               | <b><i>Descrição</i></b>   |
| Create For                         | Permite escolher a plataforma que você deseja para criar um projeto   |
| Playback:<br>Play Every Movie      | Se você escolher esta opção e seu projeto tiver mais de um filme, eles rodaram em ordem   |
| Playback:<br>Animate in Background | Se você escolher esta opção, o seu projeto continuará tocando, ainda que a janela de outros aplicativos é que esteja ativa. Caso contrário, se você começar outro aplicativo, o seu projeto vai ficar em estado de pausa. |
| Options:<br>Full Screen            | Roda o projeto usando a janela inteira.   |
| Options:<br>In a Window            | Roda o seu projeto numa janela com o tamanho de seu stage.  |
| Show Title Bar                     | Se escolher esta opção e seu projeto estiver rodando em uma janela, aparecerá a barra de título com o nome de seu projeto.  |
| Stage Size:<br>Use Movie Settings  | Cada filme será mostrado no tamanho em que foi criado. (Caso o seu projeto tenha mais de um filme)  |

|                                       |  |
|---------------------------------------|--|
| Stage Size:<br>Match First Movie      | O projeto ficará sempre do tamanho do primeiro filme. (Caso o seu projeto tenha mais de um filme)  |
| Center                                | Centraliza o projeto na tela   |
| Media: Compress<br>(Shockwave Format) | Comprime o projeto para ser usado na Internet. Isto mantém o tamanho do arquivo o menor possível para fazer downloads. No entanto, o tempo para carregar o projeto é aumentado, porque ele tem que ser descarregado antes. |
| Xtras: Include<br>Network Xtras       | Se o projeto vai ser usado na Internet, selecione esta opção. Ela garante que os Xtras requeridos pelo seu projeto sejam incluídos.  |
| Xtras: Check<br>Movies for Xtras      | O Director verifica todos os filmes do projeto para estar certo de incluir todos os Xtras necessários.   |

### **Distribuição**

Uma vez criado e testado, você pode distribuir o seu projeto. Você pode fazê-lo através de disquetes, por CD-ROM ou via Internet, através de um download. Neste último caso, o seu projeto vai rodar somente quando terminar de ser carregado em seu disco. Para rodar diretamente de um navegador, o processo é um pouco diferente e veremos adiante.

A melhor maneira, sem dúvida, para distribuí-lo é por CD-ROM. Inclusive porque você nem precisa instalá-lo no disco rígido do computador. Ele pode rodar direto do CD. E mesmo que o seu arquivo final seja muito grande, provavelmente ele vai ser pequeno perto da capacidade máxima de um CD (650MB). De mais a mais, se você recorreu a muitos arquivos de vídeo ou som, que em geral são bem grandes, também não precisarão ser copiados para o disco rígido.

### **Behaviors**

Um projeto multimídia sem interatividade perde grande parte de seu charme. Torna-se puramente um slide-show, sem a navegação do usuário. No entanto, um dos pontos mais críticos no aprendizado do Director é a criação de scripts ou comportamentos que propiciem a interatividade do usuário com o programa. Os scripts são formados por códigos de programação e se o desenvolvedor não tiver vivência nesta área, ele vai ter dificuldades iniciais para aprender a linguagem. A linguagem de programação do Director é chamada de Lingo – nos capítulos posteriores falaremos sobre ela.

Para facilitar a vida destes desenvolvedores que não têm interesse em conhecer a fundo o Lingo, o Director criou uma espécie de biblioteca de códigos já prontos – os *behaviors*. Os behaviors são agrupados em arquivos com extensão CST e podem ser vinculados ao seu projeto. Para fazer isto, siga estes passos:

- 1 – Pressione Modify ⇒ Movie ⇒ Casts.
- 2 – Na janela Movie Cast, clique no botão Link.
- 3 – Na janela Abrir selecione o arquivo CST que você deseja vincular. Depois pressione Abrir.
- 4 – Na janela Movie Cast, clique em OK.
- 5 – Finalmente, para visualizar os behaviors, pressione Choose Cast  na janela Internal Cast e escolha o cast que você abriu.

Aparecerá listada na janela Internal Cast, os behaviors do cast que você abriu.

O Director fornece uma biblioteca de behaviors que vêm junto com o software, mas que para serem acessados, você deve pressionar Xtras ⇒ Behavior Library.

Esta janela possui behaviors para os scripts mais usuais utilizados no Director, como por exemplo, navegar entre frames e congelar a navegação, parando o filme em um determinado frame.

### ***Atribuindo behaviors em objetos***

Uma vez com uma biblioteca de behaviors à sua disposição, você pode atribuí-los a um ou mais objetos de seu filme. O processo é bastante simples. Vamos a um exemplo.

Vamos dizer que você queira que o filme rode normalmente, mas que ele congele no 15º frame. Como isto não é possível, segundo a lógica de funcionamento do Director, é preciso criar um script em Lingo para isto. No entanto na biblioteca Behavior Library, já há listado um behavior que faz isto: o behavior Hold on Current Frame. Basta então arrastar este behavior para o objeto necessário. Vejamos:

1 – Abra a biblioteca Behavior Library, se ela já não estiver aberta: Xtras ⇒ Behavior Library.

2 – Procure pelo behavior Hold on Current Frame.

3 – Clique e arraste este behavior até o 15º frame no canal de script  do Score.

Pronto. Você já pode testar. Abra o Painel de Controle (Ctrl +2) e pressione o botão Play. Quando o filme chegar no 15º frame, ele vai parar. E você não criou nenhum script para isto!

É bom lembrar que o número de behaviors é limitado. Em algum momento, você vai perceber que aquele comportamento que você desejaria a um determinado botão não existe listado em Behavior Library. A grande riqueza é você poder criar os seus próprios behaviors e armazená-los em arquivos com extensão CST, para poder utilizá-los quando você precisar.

Em um primeiro momento, você pode usar os behaviors já fornecidos pelo Director, mas é bom que com o tempo você perca o medo e comece a criar os seus próprios códigos usando o Lingo. Uma das vantagens dos behaviors é a de justamente quebrar os primeiros bloqueios iniciais na hora de programar, mesmo porque você pode visualizar o código de um behavior:

1 – Selecione o 15º frame no canal de script do Score.

2 – Pressione o botão Behavior Inspector  na barra de tarefas.

3 – Aparecerá listado o behavior que você usou neste frame – o Hold on Current Frame. Selecione este behavior.

4 – Pressione o botão Script Window .

Será aberta a janela de Script, com todo o código deste behavior. Aprenda a visualizar sempre o script para se habituar com a linguagem Lingo. Com o tempo, você estará criando os seus próprios behaviors e scripts.

### ***Criando behaviors, sem conhecer Lingo***

Vamos criar um behavior, que altere a posição de um retângulo, quando você pressioná-lo:

1 – Crie um retângulo no meio do Stage.

- 2 – Pressione o botão Behavior Inspector  na barra de tarefas.
- 3 – Pressione o botão Behavior Popup . Selecione New Behavior.
- 4 – Na janela Name Behavior, dê um nome para o behavior que você vai criar: “Muda Cor”.
- 5 – Pressione OK.
- 6 – Logo abaixo, há um pequeno triângulo  voltado para a esquerda. Clique nele para expandir a janela Behavior Inspector.
- 7 – Pressione o botão Event Popup .
- 8 – Aparecerá um menu suspenso com uma lista de vários eventos. Escolha o evento mouseUp, que significa quando soltar o botão do mouse de cima do retângulo (Veremos nos capítulos posteriores, maiores detalhes sobre a linguagem Lingo).
- 9 – Pressione agora o botão Action Popup , à direita. Vamos escolher uma ação para quando pressionarmos o retângulo – no caso, mudar a posição dele. No menu suspenso aberto pressione Sprite ⇒ Change Location.
- 10 – Na janela Specify Location, deixe como está – o que significa que o retângulo vai ser posicionado na posição 1,1 da tela. Pressione OK.
- 11 – Abra o Painel de Controle (Ctrl+2) e pressione o botão Play.
- 12 – Clique em cima do retângulo. Veja que ele mudou de lugar!

## O Som

Colocar som no seu filme é uma tarefa bastante simples no Director. Basta importá-lo como você faria com qualquer outro elemento. Depois de importado, o som torna-se um cast member e fica armazenado na janela Internal Cast:

- 1 – Selecione File ⇒ Import (Ctrl + R)
- 2 – Procure na pasta correspondente ao arquivo de som que você deseja importar, selecione-o e pressione o botão Add. Você também pode selecionar mais de um arquivo de som simultaneamente com a tecla Shift.
- 3 – No campo Media, você pode escolher uma das opções na lista: Standart Import, Link To External File ou Include Original Data for Editing (leia a respeito destas opções no próximo item – “Sons internos X Sons externos”).
- 4 – Quando você acabou de selecionar todos os arquivos que deseja importar, clique o botão Import. Todos os arquivos de som que você importou vão aparecer na janela Internal Cast.

### *Sons internos X Sons externos*

Arquivos de som internos e arquivos de som externos são armazenados de modo diferente. Um arquivo de som é interno, quando você o importa com a opção Standart Import no campo Media da janela Import Files. Arquivos de som internos são embutidos no seu filme. Isto significa que todos os dados do arquivo de som ficam incorporados no arquivo DIR. Neste caso, se você desejar, pode até mesmo apagar o arquivo de som original, porque quando for tocado, o Director não vai buscá-lo no seu caminho de origem, pois ele agora está internamente incorporado ao seu filme.

Arquivos de som externos, por outro lado, são vínculos. Ou seja, seus dados não são incorporados no filme. Quando você pede para tocá-los, o Director vai buscá-lo no caminho

especificado, onde está armazenado. Você, portanto, não pode apagá-lo, pois se fizer isto, o Director não vai encontrá-lo. Para tornar um arquivo externo, selecione a opção Link To External File no campo Media da janela Import Files.

A vantagem do arquivo de som interno é que você não precisa se preocupar na hora de distribuir o seu projeto, pois este arquivo de som está incorporado no filme. No entanto, se você tiver muitos arquivos de som, eles podem aumentar demasiadamente o tamanho do arquivo final. Ao contrário, arquivos de som externos não aumentam o tamanho de seu filme, pois são armazenados separadamente. Além disso, arquivos de som externos têm a vantagem de já começarem a tocar, enquanto o resto do arquivo ainda está sendo carregado. Arquivos de som internos não fazem isto, têm que ser completamente carregados, antes de começarem a ser tocados.

Uma boa política é armazenar pequenos arquivos de som como internos e arquivos de som grandes como externos.

Para saber posteriormente, depois de importados, se um arquivo de som é interno ou externo, basta pressionar o botão direito do mouse em cima do arquivo de som – na janela Internal Cast ou direto no Score – e selecionar Cast Member Properties. Se abaixo do nome do cast member – que em princípio é o mesmo do arquivo em si, mas que pode ser alterado – aparecer o caminho de onde o arquivo de som foi importado, este som é externo. Caso não apareça nada, este som é interno.

Finalmente, há a opção Include Original Data for Editing do campo Media da janela Import Files. Caso você escolha esta opção, quando importar um arquivo de som, você poderá editar e atualizar rapidamente o seu arquivo de som quando der um duplo-clique na janela Internal Cast. Será aberto um programa editor de arquivos de som para que você possa alterá-lo. Para especificar um editor externo para arquivos de som, siga estes passos:

- 1 – Selecione File ⇒ Preferences ⇒ Editors.
- 2 – Na janela Editors Preferences, selecione em Description, o formato de arquivo de som que pretende editar.
- 3 – Pressione o botão Edit.
- 4 – Pressione o botão de seleção Use External Editors.
- 5 – Clique em Browse.
- 6 – Na janela Abrir, selecione o programa que deseja usar para editar o seu arquivo de som.
- 7 – Clique em Abrir e depois duas vezes em OK.

### ***Escolhendo um formato de som***

O Director suporta basicamente três formatos de arquivos de som: AIF, WAV e “Macintosh System 7”. Embora os dois primeiros sejam multiplataformas, você deve se preocupar particularmente com o formato WAV, porque só mais recentemente ele foi incorporado à plataforma Macintosh. No entanto, se o seu filme for rodar exclusivamente em Windows, o formato WAV é a melhor opção, por ser o tipo de arquivo de som nativo deste sistema operacional, minimizando possíveis problemas na hora de rodar. Já o formato “Macintosh System 7” é exclusivo para Macintosh e na realidade permite gravar arquivos de som do Director. Para fazer isto, você tem que selecionar Insert ⇒ Media Element ⇒ Sound.

### ***Adicionado sons em seu filme***

Uma vez importado o seu arquivo de som, você pode dar mais vida ao seu filme. Adicionar sons a ele é relativamente fácil, basta arrastá-lo da janela Internal Cast para a janela Score:

1 – Comece expandindo a janela Score, para que você possa enxergar o canal de som. Você pode fazer isto, clicando no botão Hide/Show Effects Properties .

2 – A janela Score mostra dois canais de som. Isto significa que você pode tocar dois arquivos de som ao mesmo tempo. Na realidade, você pode até tocar mais de dois, mas você terá que usar o Lingo para fazer isto.

*Embora tecnicamente o Windows não possa tocar dois sons ao mesmo tempo, pois só tem um canal de som, o Director acomoda esta situação misturando os dois arquivos de som para serem tocados neste único canal do som do Windows.*

3 – Clique no som na janela Internal Cast e arraste até o local que você selecionou no Score. Quando você soltar o botão, o som aparece no Score. Se você arrastar o som direto para o stage, o Director coloca o som no Score no primeiro canal de som disponível.

4 – Pressione o botão Play do Painel de Controle para testar.

### ***Alterando a duração dos sprites de som***

Pode ter acontecido de o seu arquivo de som não ter sido tocado por completo. Isto acontece porque todos os sprites são criados com uma duração padrão definida nas Preferências do sprite (File ⇒ Preferences); geralmente 28 frames. Às vezes este valor não é suficiente para que o som seja tocado por completo. Você vai ter que alterar isto direto no score:

1 – Abra a janela Score (Ctrl +4), se ela ainda não estiver aberta.

2 – Clique no último frame do som que você pretende mudar e arraste para a direita, até notar que todo o som vai ser tocado.

3 – Volte o filme e rode novamente. Se ainda não ficou bom, tente novamente até conseguir ajustar convenientemente.

### ***Ancorando o som em um frame***

Se você acha um pouco complicado ter que ficar adivinhando a duração exata de um arquivo de som para ele não ser cortado, há uma maneira um pouco mais simples de eliminar este problema, mas que você só pode usar em algumas ocasiões: ancorar um som em um determinado frame. Isto significa: quando o filme chegar a um determinado frame, você pode “pedir” para o Director parar o filme até que este som termine de tocar. Para isto, você usa o canal Tempo:

1 – Arraste um arquivo de som em um dos canais de áudio do Score.

2 – Clique no último frame do canal de áudio em que este arquivo de som foi colocado e arraste até o ponto em que você deseja que ele seja tocado.

3 – Dê um duplo-clique neste mesmo ponto, só que no canal Tempo. Deverá ser aberta a janela Frame Properties: Tempo.

4 – Nesta janela, selecione o botão Wait for Cue Point.

5 – No campo Channel, selecione o mesmo arquivo de som que você arrastou para o Score.

6 – Clique em OK.

7 – Teste. Você vai perceber que o seu filme vai rodar normalmente até chegar no frame que você configurou para esperar até o som terminar de tocar. O filme vai ficar parado neste frame até terminar de tocar o arquivo de som. Depois ele volta a rodar normalmente.

### ***Som em looping***

Sons em loopings são sons que ficam tocando ininterruptamente; até que você peça explicitamente para ele parar ou você navegar para outro frame que não contenha este som. No Director é muito fácil fazer um arquivo de som ficar tocando em looping:

- 1 – Na janela Internal Cast, selecione o arquivo de som que você quer colocar em looping.
- 2 – Selecione Modify ⇒ Cast Member ⇒ Properties. Será aberta a janela correspondente.
- 3 – Na janela aberta, selecione o botão de seleção Loop.
- 4 – Pressione Ok.
- 5 – Arraste o arquivo de som da janela Internal Casta para um dos canais de áudio do Score.
- 6 – No Score, clique no último frame onde ele foi colocado e arraste para a direita até ele ocupar mais frames do que a sua duração.
- 5 – Teste. Você verá que agora o arquivo de som vai ficar tocando em looping.

### ***Interatividade***

Até agora você colocou som no seu filme simplesmente arrastando um arquivo de som da janela Internal Cast para o Score. Mas até agora você não adicionou nenhum efeito sonoro em sprites. Por exemplo, quando você clica em um botão do filme, ele toca um pequeno sinal sonoro, uma música, dispara uma narração. Para fazer isto é necessário conhecer um pouco da linguagem Lingo. Felizmente, o Director introduziu alguns behaviors que fazem isto para você. Enquanto não entrarmos definitivamente no estudo do Lingo, vamos utilizar deste recurso para criar este tipo de interatividade:

- 1 – Crie um botão direto no stage.
- 2 – Selecione Xtras ⇒ Behavior Library.
- 3 – Na janela aberta, selecione o behavior “Sound Play Member” e arraste para o botão no stage ou no score.
- 4 – Vai ser aberta uma nova janela para que você escolha os parâmetros necessários para se tocar o som. No campo Sound, escolha um dentre os arquivos de som que fazem parte de seu filme. Mantenha Channel como está e escolha em Initializing Event o tipo de evento que você quer estabelecer com o botão. Por exemplo, mouseUp, que significa quando você pressionar o botão esquerdo do mouse.
- 5 – Clique em OK e teste. Quando você pressionar o botão que você colocou no stage, o som que você escolheu como parâmetro vai ser tocado assim que você clicar com o botão esquerdo do mouse em cima dele.

## **O Vídeo**

O Director fornece suporte para que seus filmes rodem arquivos de vídeo digitais. Basicamente são dois os formatos aceitos: o Video for Windows (AVI), preferencialmente para plataformas Windows, e o Quick Times para Macintoshs. No entanto, os dois formatos já possuem suporte nas duas plataformas. O Quick Time for Windows é a versão deste formato para Windows e utiliza a extensão MOV. O formato AVI também já pode ser rodado em Macintoshs mais recentes.

Você importa um vídeo digital da mesma maneira que qualquer outro elemento de seu filme. A diferença em relação aos outros, é que arquivos de vídeo não podem ser embutidos no arquivo final de seu filme. Eles são necessariamente vinculados.

Quando você pede para um vídeo ser rodado, o Director vai procurar o arquivo de vídeo exatamente no caminho onde ele se encontrava quando você o importou. Caso você mova o seu arquivo de vídeo para outro caminho que não aquele especificado quando você o importou, o Director não poderá encontrá-lo. Uma medida de segurança é copiar o seu arquivo de vídeo exatamente para o mesmo diretório onde se encontra o seu filme, antes de importá-lo.

Para importar um vídeo, siga estes passos:

- 1 – Selecione File ⇒ Import para mostrar a janela Import Files.
- 2 – Em File of types, escolha Video Clip (se estiver trabalhando na plataforma Windows) ou Quick Time (se estiver no Mac).
- 3 – Em Media escolha uma entre duas opções: Include Original Data for Editing, para que rapidamente você possa editar o seu arquivo de vídeo se der um duplo-clique no cast member correspondente; caso contrário escolha Link to External File.  
*Se você quiser atribuir um outro programa para editar os seus arquivos de vídeo quando der um duplo-clique no cast member correspondente, siga os mesmos passos descritos anteriormente para arquivos de som. Selecione File ⇒ Preferences ⇒ Editors.*
- 4 – Selecione o vídeo digital na lista de arquivos e pressione Import.

### ***Colocando um vídeo digital no stage***

A maneira de colocar um vídeo no seu filme é tão elementar quanto qualquer outro elemento:

- 1 – No Score, clique no frame onde você deseja que o vídeo comece a rodar.
- 2 – Clique no vídeo e arraste da janela Cast ao Score. O vídeo vai aparecer centralizado no stage. Você pode também arrastar o vídeo direto para o stage no ponto onde deseja que ele apareça. Note que o vídeo, por padrão, sempre apareça por cima de todos os outros elementos de seu filme. Isto acontece porque, por padrão, o Director ativa a opção Direct to Stage para todos os vídeos importados. Veja abaixo a diferença entre ter ou não esta opção ativada.
- 3 – Volte o seu filme e comece a rodá-lo. Quando o filme chegar no frame correspondente ao início do vídeo, este começa a rodar. Se ele não rodar completamente, você pode clicar e arrastar no último frame correspondente ao vídeo no Score e arrastar para a direita, até que ele rode completamente.

### ***O que é Direct to Stage***

Todos os vídeos no Director são, por padrão, importados com a opção Direct to Stage ativada (se o vídeo que você importou for Quick Time for Windows, você não pode alterar esta opção). Quando esta opção está habilitada, você pode sincronizar o vídeo com o canal de som. Outra vantagem, é que você pode fornecer recursos para que o vídeo rode o mais rápido possível de forma a acompanhar a velocidade do filme.

A desvantagem é que o vídeo sempre vai estar à frente de todos os outros elementos do seu filme, mesmo que um outro elemento tenha sido colocado depois, ou que você tenha alternado as suas posições no stage. Outra desvantagem é que efeitos de transição e *ink* não funcionam.

Para habilitar ou desabilitar Direct to Stage:

- 1 – Com o cast member do vídeo selecionado, clique Modify ⇒ Cast Member ⇒ Properties para mostrar a janela Cast Member Properties.

2 – Selecione a opção Direct to Stage para habilitar ou vice-versa.

3 – Clique OK para completar a operação.

### ***As outras propriedades do Vídeo***

| <b>Janela Digital Video Cast Member Properties</b> |   |
|--|---|
| <b><i>Opções</i></b>                               | <b><i>Descrição</i></b>   |
| Playback   | São quatro opções:<br><i>Video</i> : habilita trilha de vídeo, se selecionado. Se não estiver selecionado, a porção do vídeo ficará parada. Se o áudio estiver habilitado, somente este irá tocar.<br><i>Sound</i> : habilita ou desabilita a trilha sonora do vídeo.<br><i>Paused</i> : se habilitado, o vídeo começa em estado de pausa. Caso contrário, começa tocando.<br><i>Loop</i> : faz o vídeo rodar em estado de looping. |
| Framing  | <i>Crop</i> : retém o tamanho original do vídeo se ele for redimensionado. Se opção <i>Center</i> estiver selecionada, o vídeo mostrará as partes dele a partir do centro.<br><i>Scale</i> : o vídeo muda de tamanho de acordo, se for redimensionado.  |
| Options  | <i>Direct to Stage</i> (já visto anteriormente).<br><i>Show Controller</i> : se a opção <i>Direct to Stage</i> estiver selecionada, mostra controles que permitem ao usuário tocar, parar, voltar etc o vídeo.  |
| Vídeo  | Estabelece como o vídeo vai ser sincronizado:<br><i>Sync to Soundtrack</i> : sincroniza o filme com a trilha sonora do vídeo. As imagens do vídeo são mostradas na medida do possível.<br><i>Play Every Frame (No Sound)</i> : todos os frames do vídeo serão mostrados, mas por causa disto o som do vídeo não poderá ser tocado.  |
| Rate   | Determina a velocidade com que o filme será tocado.<br><i>Normal</i> : cada frame do vídeo é mostrado normalmente, sem escapar nenhum.<br><i>Maximum</i> : o filme é rodado o mais rápido possível, enquanto é mostrado cada frame.<br><i>Fixed</i> : toca o filme usando uma velocidade específica. Entre com o número de fps no campo à direita.  |
| Enable Preload                                     | Pré-carrega o vídeo na memória antes de ele começar a ser tocado. Se não for possível carregar todo o vídeo, será carregado o que couber.   |
| Unload   | Controla como o Director remove o cast member da memória se esta for baixa.<br><i>Normal</i> : o cast member selecionado será removido da memória se necessário.<br><i>Next</i> : o cast member selecionado será o próximo a ser removido da memória.<br><i>Last</i> : o cast member selecionado será o último a ser removido da memória.<br><i>Never</i> : o cast member selecionado nunca é removido da memória.                  |

### ***Behaviors para vídeo***

O Director vem com um pacote de behaviors para vídeos:

| <b><i>Nome do Behavior</i></b> | <b><i>Descrição</i></b>   |
|--------------------------------|---|
| Video Rewind                   | Volta um vídeo quando um usuário clica o botão.                   |
| Video Play                     | Toca um vídeo quando um usuário clica um botão.                   |
| Video Stop                     | Pára um vídeo quando um usuário clica um botão.                   |
| Video Control                  | Fornecer controles para voltar, tocar, parar, colocar uma pausa e |

|              |  |
|--------------|--|
|              | saltar trechos de um vídeo.  |
| Video Slider | Inclui um controle de slide que aponta para o progresso de um vídeo. |

Aplicando um behavior a um vídeo

- 1 – Selecione Xtras ⇒ Behavior Library
- 2 – Procure na janela Behavior Library Cast o behavior que você quer usar. Clique e arraste o behavior para o sprite, no stage ou no score. Aparecerá um janela para você incluir parâmetros.
- 3 – Na janela aberta, entre com os valores pedidos.
- 4 – Clique OK para completar a operação.

### ***Ancorando um vídeo em um frame***

No canal Tempo, você pode estabelecer alguns parâmetros que controlam o modo como o seu vídeo interage com o seu filme. Por exemplo, você pode estabelecer uma pausa no filme até que um vídeo termine de tocar:

- 1 – Arraste um cast member de vídeo para um dos canais da janela Score.
- 2 – Na janela Score, dê um duplo-clique, no canal Tempo, no frame que você deseja estabelecer uma pausa. Aparecerá a janela Frame Properties: Tempo.
- 3 – Na janela aberta, clique o botão de seleção Wait for Cue Point para habilitar esta opção.
- 4 – Em Channel, selecione o sprite de vídeo que você criou.
- 5 – Em Cue Point, escolha {End}. Esta opção faz o filme parar até o vídeo escolhido ter terminado de tocar.
- 6 – Clique OK para terminar a operação.

*Você pode estabelecer um Cue Point alternativo (na janela Frame Properties: Tempo, só há em princípio duas opções para Cue Point: {End} e {Next}). Com programas como o Macromedia's SoundEdit 16, somente avaliável para Macintosh, você pode inserir cue points em vídeos Quick Time.*

### ***Exportando filmes no formato AVI***

Você pode tornar o filme que criou em Director em um arquivo AVI:

- 1 – Após testar exaustivamente o seu filme para verificar se não tem nenhum problema, selecione File ⇒ Export (Ctrl + Shift + R). Será aberta a janela Export.
- 2 – Caso você queira exportar todos os frames de seu filme, selecione All Frames em Export. Caso contrário, escolha exatamente o intervalo de frames que deseja exportar em Frame Range.
- 4 – Em Format, escolha Video for Windows (.AVI), se estiver na plataforma Windows. Senão escolha QuickTime Movie, para Mac.
- 5 – Clique o botão Options. Se você estiver trabalhando no Windows, aparecerá a janela Video for Windows Export Options. Nesta janela, escolha um valor em fps (frame per second) no campo Frame Rate. Esta opção determina a velocidade em que seu vídeo vai rodar. Pressione OK.
- 6 – Se estiver num Mac, a janela aberta QuickTime Options tem um pouco mais de opções, como, por exemplo, qual canal de som será considerado, a profundidade de cor (Color Depth) ou o formato compressor. Quando escolher as opções, pressione OK.
- 7 – Se você estiver em Windows, após fechar a janela Video for Windows Export Options, pressione Export. Uma nova janela será aberta, pedindo para você escolher o nome do vídeo que será criado e o local onde será salvo. Pressione Salvar.

8 – Uma última janela ainda será aberta: Compactação de Vídeo. Escolha aqui o codec de preferência em Compactador e clique em OK. Uma janela vai aparecer mostrando o progresso da formação do arquivo.

*Se você estiver na plataforma Windows, todos os sons utilizados no filme não serão exportados. O seu vídeo ficará mudo. Caso queira incluir uma trilha sonora no seu vídeo, você terá que fazer isto em algum outro programa, como o Adobe Premiere.*

## Lingo: criando scripts no seu filme

Até o momento temos usado os behaviors para criar interatividade nos filmes. Embora eles sejam uma excelente ferramenta, principalmente para aqueles desenvolvedores que não pretendem se tornar programadores de última hora, somente aprendendo o Lingo é que você vai realmente ter maior controle sobre aquilo que está produzindo.

*Embora os behaviors cubram grande parte dos comportamentos básicos que você precisa para criar interatividade no seu filme, como clicar em botão para tocar um arquivo de som, navegar entre diferentes frames, ou parar o filme em determinada posição, com o tempo você vai querer ter maior controle nos seus scripts e não ficar preso a programações já prontas. Isto leva tempo, mas um boa dica é atribuir um behavior a um determinado elemento (sprites, cast members, frames, movies), abrir a janela Script para ver como é a programação deste behavior, e com o tempo tentar alterá-lo, personalizando de acordo com suas necessidades.*

É natural que certos desenvolvedores não tenham a mínima aptidão para programar. Alguns preferem se preocupar somente com o trabalho gráfico, outros gostam de produzir animações, alguns se limitam a roteirizar o filme e param por aí. No entanto, com um pouco de boa vontade é possível aprender os princípios do Lingo sem muito esforço, porque esta linguagem é relativamente simples, baseada no inglês e muito bem estruturada. Compreender a terminologia e os componentes que fazem parte da linguagem é, portanto, o primeiro passo.

Em princípio entenda o seguinte: qualquer elemento no Director pode provocar alguma ação ou alguma interatividade: sprites, cast members, frames ou o próprio movie (filme). Ou seja, eles podem ser os responsáveis para que algo aconteça no seu filme – são eles que disparam uma ação. Se você clicar num botão, é a partir dele que uma ação é executada; é ele o responsável para que isto ocorra.

Mas, para que você clique em um botão e ele execute algo, como tocar um som, é necessário que exista algum lugar no Director onde esteja escrita esta ordem: “toque tal arquivo de som”, quando “clicar” no mouse. Este lugar é chamado *Script* (veja mais adiante em **Instruções**). Todos os elementos possuem uma janela de Script para que você programe o que ele deve fazer se alguma coisa acontecer – clicar com o botão esquerdo ou direito do mouse, pressionar um tecla do teclado, entrar com o mouse por cima de determinado elemento etc. Esta alguma coisa que acontece e dispara uma determinada ordem ou ação é chamado de *Evento*.

### **Os eventos**

Um evento é portanto um incidente. É o ato responsável para que o filme execute uma determinada ação. Alguns eventos são fáceis de entender e bastante conhecidos, porque são

disparados pelos usuários: cliques de mouse ou teclas pressionadas. Outros eventos são um pouco mais obscuros, porque não dependem da ação do usuário, como: o início ou fim do filme ou o momento em que o filme entra ou sai de um frame.

*Obs.:* A partir de agora, tudo o que aparecer no texto com a fonte *Courrier*, faz parte da linguagem Lingo e pode descrever um script.

Basicamente os eventos no Director são os seguintes:

| <b>Eventos do Director</b> |   |                               |
|----------------------------|---|-------------------------------|
| <b>Eventos</b>             | <b>Descrição</b>  | <b>Quem recebe</b>            |
| activatedWindow            | Uma janela é selecionada ou se torna ativa  | Movie                         |
| closeWindow                | Uma janela é fechada  | Movie                         |
| cuePassed                  | Um <i>cue point</i> em um sprite ou som é passado   | Todos                         |
| deactivateWindow           | Uma janela perde a seleção ou torna-se inativa.   | Movie                         |
| endSprite                  | <i>Playback head</i> deixa o último frame da duração (span) de um sprite.   | Score                         |
| enterFrame                 | O filme entra em um frame.  | Frame e Movie                 |
| exitFrame                  | Ocorre entre o momento em que o frame é completamente desenhado e em que ele deixa o frame.   | Frame e Movie                 |
| idle                       | O filme está em estado inercial, ou seja, o usuário não está fornecendo interatividade, como clicar um botão do mouse ou pressionar uma tecla.  | Frame e Movie                 |
| keyDown                    | O usuário pressiona uma tecla.  | Editable Field, Frame e Movie |
| keyUp                      | O usuário libera uma tecla.   | Editable Field, Frame e Movie |
| mouseDown                  | O botão do mouse (o esquerdo, para Windows) é pressionado.  | Todos                         |
| mouseEnter                 | O exato momento em que mouse entra (sem clicar) dentro dos limites de um sprite.  | Todos                         |
| mouseLeave                 | O mouse deixa de estar dentro dos limites de um sprite.   | Todos                         |
| mouseUp                    | O botão do mouse é liberado, depois de clicar em cima de um sprite.   | Todos                         |
| mouseUpOutside             | O botão do mouse é liberado, mas fora dos limites do sprite, depois de clicar em cima de um sprite.   | Sprite e Cast                 |
| mouseWithin                | Enquanto o mouse permanece dentro dos limites de um sprite, com o botão do mouse pressionado ou não.  | Todos                         |
| moveWindow                 | Uma janela é movida.  | Movie                         |
| openWindow                 | Uma janela é aberta.  | Movie                         |
| prepareFrame               | Instruções para este evento ocorrem antes que o frame corrente seja desenhado. É útil para alterar propriedades de sprites, antes de eles serem | Frame e Movie                 |

|                |  |               |
|----------------|--|---------------|
|                | desenhados.  |               |
| prepareWindow  | Instruções para este evento ocorrem depois que um filme carrega os cast members e antes que o filme prepare o primeiro frame para rodar. Isto inclui desenhar o frame, tocar algum som e executar alguma efeito de transição ou de palheta de cores. | Movie         |
| resizeWindow   | O usuário redimensiona o tamanho de uma janela.  | Movie         |
| rightMouseDown | O usuário pressiona o botão direito do mouse. (Para usuários Mac, isto corresponde a clicar o botão do mouse ao mesmo tempo que pressiona a tecla Command)   | Todos         |
| rightMouseUp   | O usuário libera o botão direito do mouse, depois de clicar com ele em um sprite.  | Todos         |
| spriteBegin    | <i>Playback head</i> entra em um frame contendo a primeira célula de um sprite.  | Frame         |
| startMovie     | Um filme começa. Útil para inicializar variáveis globais, atribuir propriedades iniciais de sprites etc.   | Movie         |
| stepFrame      | Instruções escritas para este evento ocorrem quando o <i>playback head</i> entra em um frame ou o stage é atualizado. Somente é válido para scripts atribuídos a objetos incluídos em <code>actorList</code> .                                       | Frame e Movie |
| stopMovie      | Um filme termina. Útil para limpar a memória, variáveis globais, fields, ou para atribuir propriedades a sprites.  | Movie         |
| timeOut        | Nada acontece. Instruções escritas para este evento rodam quando o usuário não clica o botão do mouse, nem pressiona uma tecla por um período de tempo estabelecido em <code>timeOutLength</code> .  | Frame, Movie  |
| zoomWindow     | Instruções escritas para este evento ocorrem quando um filme está rodando em uma janela que é redimensionada pelo usuário. Por exemplo, quando a janela é maximizada ou minimizada.  | Movie         |

### ***Instruções***

Uma instrução é a ação que acontece quando um evento ocorre. Por exemplo, você pode pedir para o filme “saltar” para o frame 30 quando você clicar o botão do mouse em cima de uma imagem. Neste caso, “saltar para o frame 30” é a instrução, que vai acontecer quando o usuário clicar o botão do mouse. “Clicar o botão do mouse” é o evento que dispara esta instrução.

No Director as instruções são armazenada dentro de *handlers*. E os handlers são armazenados nos scripts de cada objeto. Handlers sempre começam com a palavra-chave

on e terminam com `end`. `on` indica o início do handler, `end` indica que a programação daquele handler terminou:

```
on <evento>
    <instrução>
end
```

Por exemplo:

```
on mouseUp
    go to the frame + 1
end
```

Este exemplo clássico diz para o filme saltar para o frame seguinte (`frame + 1`), quando o usuário clicar o botão do mouse no objeto que contém este handler no seu script. Um script pode ter mais do que um handler, desde que eles não sejam gerados por eventos iguais. Assim, não é válido o script abaixo, porque possui dois handlers para o evento `mouseUp`:

```
on mouseUp
    go to the frame + 1
end

on mouseUp
    puppetSound 1, "som"
end
```

Isto parece bastante lógico, se considerarmos que o Director ficaria bastante confuso na hora de saber qual dos dois `mouseUp` ele deveria executar. Esta mesma programação poderia ser alterada com o script abaixo:

```
on mouseUp
    go to the frame + 1
    puppetSound 1, "som"
end
```

Ou seja, quando pressionar o botão do mouse, salte para o frame seguinte e toque o arquivo de som "som" no canal de som 1. As instruções são executadas linha a linha, à medida que aparecem. Uma instrução só é realizada no momento em que a instrução da linha anterior tiver sido executada de maneira satisfatória.

Por outro lado a programação abaixo é bastante válida:

```
on mouseDown
    go to the frame + 1
end

on mouseUp
    puppetSound 1, "som"
end
```

Neste caso, há dois handlers no mesmo script. Mas estes handlers são gerados por eventos diferentes (`mouseDown` e `mouseUp`), o que não provoca contradições. Ou seja, uma instrução (`go to the frame + 1`) é executada quando o botão do mouse é pressionado. A outra instrução (`puppetSound 1, "som"`) é executada quando o botão do mouse é liberado.

Sintetizando: os eventos só são respondidos no modo de execução de um filme. Isto é, quando o filme estiver rodando. Os eventos enviam mensagens. Quando o botão do mouse

é clicado, o Director envia uma mensagem dizendo que este evento ocorreu. Os scripts são compostos de um ou mais handlers que respondem a estas mensagens.

### ***Acessando scripts de objetos***

Quando você clica em um botão e ele toca um arquivo de som, a programação deste ato pode estar no script do sprite, do cast member deste botão, do frame onde se encontra este botão ou do filme. Se a programação estiver no cast member, todos os sprites criados a partir dele executarão esta ação. Se, por outro lado, a programação estiver no sprite do botão, somente este sprite executará esta ação. Um outro sprite criado a partir do cast member deste botão não executará a ação, pois simplesmente não há programação nele. Se o handler estiver no script do frame, na realidade esta ação será executada quando se clicar em qualquer lugar do stage e não somente neste botão. Finalmente se a programação estiver no filme (movie), esta ação será executada se você clicar em qualquer lugar do filme.

Para atribuir um script a um cast member, selecione o cast member na janela Cast e pressione Modify ⇒ Cast Member ⇒ Script. Uma segunda maneira é pressionar o botão direito do mouse em cima do sprite ou do cast member na janela Cast e selecionar no menu suspenso Cast Member Script. Scripts de cast members são armazenados junto com eles na janela Cast. Quando um cast member possui um script, na janela Cast, um ícone identificador  aparece no canto inferior direito do respectivo cast member.

Para atribuir um script a um sprite, selecione Modify ⇒ Sprite ⇒ Script. Uma outra opção é pressionar o botão direito em cima do sprite e clicar em Script no menu suspenso aberto.

Quando um script de sprite é criado, ele é armazenado na janela Cast como um cast member de script e pode ser usado posteriormente em qualquer outro sprite ou em frames.

Scripts de frame são acessados selecionando o frame na janela Score – arraste o playback head até o frame desejado – ou selecionando Modify ⇒ Frame ⇒ Script. Uma segunda opção bastante prática é dar um duplo-clique no canal script da janela Score em cima do frame desejado.

Para acessar o script do Movie, pressione Ctrl + Shift + U. Como já dissemos, scripts de movies (filmes) podem ser executados de qualquer lugar do filme. Portanto é uma boa idéia incluir scripts que se repetem várias vezes no filme, em vez de ficar repetindo-os em todos os elementos que o contenham. (Lembre que scripts de movies aceitam todos os eventos do Director). Em geral scripts de movies são excelentes para iniciar variáveis globais e para atribuir propriedades iniciais de objetos no Director. É também excelente para criar seus próprios eventos, conhecidos no Director como *user-defined handlers*, que podem ser chamados como sub-rotinas (veja adiante em “Criando sub-rotina”).

### ***Que script tem prioridade?***

A questão é saber: se um sprite tem um script para determinado evento e o cast member correspondente a este sprite também tem um handler para este mesmo evento, qual dos dois será executado?

O Director tem uma hierarquia em relação a isto.

Em primeiro lugar, é sempre executado o script do sprite. Somente se o sprite não tiver um script para a mensagem enviada por um determinado evento qualquer é que o Director vai chamar o script do cast member correspondente. Se o cast member tiver um script para este

evento, ele será executado. Caso contrário, o Director vai enviar uma mensagem procurando um script no frame correspondente. Se encontrar script para este evento no frame, ele será executado. Senão, uma mensagem é enviada buscando um script no movie.

### ***Criando os seus próprios scripts com um filme interativo***

Para compreender melhor como criar scripts, vamos desenvolver um filme bastante simples, mas que utiliza alguns dos handlers que aparecem mais frequentemente em qualquer projeto interativo.

Este filme vai trabalhar basicamente com quatro telas. A primeira tela de abertura, vai conter três botões de interação. Clicando no primeiro, você vai para o segundo frame. Neste frame, haverá um botão para voltar para a tela inicial. A mesma coisa deve acontecer com os outros dois botões de interação. Só que clicando neles, eles devem nos levar respectivamente para os frames 3 e 4. Neste frames também devem haver o mesmo botão para voltar para a tela 1. Para que o filme tenha alguma graça, você pode importar uma foto em cada frame, para que pareça uma mostra fotográfica.

Vamos começar importando uma foto para cada frame. Para isso vamos antes alterar a duração padrão de cada sprite para 1 frame:

- 1 – Selecione File ⇒ Preferences ⇒ Sprite.
- 2 – Na janela Sprite Preferences, altere Span Duration para 1.
- 3 – Clique OK para completar a operação. Todos os sprites criados a partir de agora ocuparão somente um frame.
- 4 – Selecione agora File ⇒ Import e procure três imagens para importar. Quando encontrá-las, clique em Import. Cada imagem vai se tornar um cast member.
- 5 – Arraste o primeiro cast member, da imagem importada, para o frame 2 do Score, o segundo cast member para o frame 3 e o terceiro cast member para o frame 4. As três imagens podem ocupar o canal 1.

Com as três imagens já aparecendo no stage, vamos agora criar os três botões de navegação que aparecem na tela de abertura:

- 1 – Abra a janela Tool Palette (Ctrl + 7).
- 2 – Selecione Push Button, arraste no stage para formar um botão.
- 3 – Digite “Imagem 1” para o cabeçalho do botão.
- 4 – Selecione o botão criado na janela Cast.
- 5 – Pressione o botão Cast Member Script  na janela Cast ou pressione Ctrl + ‘ .
- 6 – Vamos agora criar a primeira programação. Na janela Script aberta digite:

```
on mouseUp
    go to frame 2
end
```

Você criou um *handler* no script do cast member. Isto significa que qualquer sprite que você criar a partir deste cast member vai poder executar esta programação.

Na programação acima `mouseUp` é o evento que vai disparar a instrução que você está pedindo. `MouseUp` significa clicar o botão do mouse (botão esquerdo para usuários Windows). Já a instrução `go to frame 2` indica que o Director vai saltar para o frame 2 do seu filme.

- 7 – Quando tiver digitado o script acima pressione a tecla Enter do teclado numérico ou fecha a janela do Script.

8 – Vamos criar os outros dois botões a partir deste. Selecione o botão “Imagem 1” na janela Cast.

9 – Pressione Ctrl+C para copiar este botão para a Área de Transferência.

10 – Selecione um cast vazio na janela Cast e pressione Ctrl+V para colar o botão como um novo cast member.

11 – Com o novo cast member ainda selecionado, pressione Ctrl+' para abrir a janela Script. Repare que já há uma programação neste script – é o *handler* que você digitou para o botão “Imagem 1”. Altere a programação para:

```
on mouseUp
    go to frame 3
end
```

12 – Pressione Enter no teclado numérico ou feche a janela Script.

13 – Vamos alterar o cabeçalho do botão. Dê um duplo-clique no novo botão. Será aberta uma janela para que você altere o cabeçalho.

14 – Digite: “Imagem 2”. Feche a janela.

15 – Refaça novamente do passo 8 ao passo 14, só que altere o script para `go to frame 4` e altere o cabeçalho para “Imagem 3”.

16 – Quando os três botões estiverem prontos, arraste-os da janela Cast para o frame 1.

Vamos criar um botão para voltar para o frame 1. Este botão vai aparecer nos frames 2, 3 e 4:

1 – Abra a janela Tool Palette (Ctrl + 7).

2 – Arraste o *playback head* da janela Score para o frame 2.

3 – Selecione Push Button, clique e arraste no stage para formar um botão.

4 – Digite “Voltar” para o cabeçalho do botão.

5 – Selecione o botão criado na janela Cast.

6 – Pressione o botão Cast Member Script  na janela Cast ou pressione Ctrl+' .

7 – Vamos agora criar a primeira programação. Na janela Script aberta digite:

```
on mouseUp
    go to frame 1
end
```

8 – Feche a janela Script. Você pode fazer isto pressionando Enter no teclado numérico.

9 – Na janela Score selecione o sprite do botão criado.

10 – Arraste o *playback head* para o frame 4.

11 – Selecione Modify ⇒ Extend Sprite. Agora este sprite ocupa os três frames (2, 3 e 4).

Vamos testar nosso filme. Volte para o início e pressione o botão Play do Painel de Controle (Control Panel). Repare que temos um problema: não é possível interagir com o filme simplesmente porque ele não pára de rodar. Seria necessário que o filme parasse em cada frame que entrasse para que pudesse haver alguma interação. Temos então um problema, porque o Director foi projetado para rodar como um filme, e por isso ele não consegue parar em nenhum dos frames por onde passa. Mas há um truque para resolver este impasse:

1 – Dê um duplo-clique no primeiro frame do canal Script da janela Score.

2 – Na janela Script aberta digite:

```
on exitFrame
    go to the frame
end
```

3 – Feche a janela.

4 – Volte para o início e teste novamente. O filme pára no frame 1.

A instrução `go to the frame` faz o filme ficar “saltando” para o frame corrente, toda vez que sair dele (`exitFrame`). O filme fica numa espécie de looping. Para sair de lá, é necessário que se crie alguma instrução para isto (no nosso caso ela já existe, porque se você clicar no botão “Imagem 1”, “Imagem 2” ou “Imagem 3”, o filme vai sair do frame 1).

No entanto ainda temos um problema. Se clicarmos um dos três botões do frame 1, o filme voltar a rodar novamente até o fim do filme. Ele não pára no frame 2, 3 ou 4. Se você clicar no botão “Imagem 1”, o filme salta para o frame 2 (`go to frame 2`) e roda até o fim. Para resolver este problema temos que lançar mão do mesmo artifício que usamos no frame 1. Como a programação já existe no frame 1 do canal Script, basta expandirmos este sprite de script até o frame 4:

1 – Clique em cima do frame 1 do canal script da janela Score.

2 – Arraste o playback head até o frame 4.

3 – Selecione `Modify` ⇒ `Extend Sprite`.

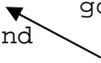
Pronto. Agora todos os quatro frames compartilham do mesmo *handler*. O filme vai parar em todos os frames que entrar.

### ***Algumas observações a respeito da sintaxe***

Você não precisa tabular a programação, como fizemos anteriormente:

```
on exitFrame
  go to the frame
end
```

tabulação



No entanto, é uma boa idéia fazê-la. Mais tarde você vai perceber que existem certos scripts que ocupam muitas linhas, não somente três como fizemos até o momento. A tabulação ajuda a entender a estrutura da programação.

No entanto, o retorno de carro é obrigatório. Você não poderia escrever a programação acima usando somente uma linha. As três linhas, no mínimo, deste exemplo, são compulsórias, caso contrário o Director indicaria uma mensagem de erro. Você poderia, sim, incluir uma ou mais linhas em branco entre as três linhas de programação ou acrescentar uma depois. Isto não afetaria em nada.

Uma boa idéia também é criar comentários sobre a suas programação. Comentários são partes da programação que são desconsideradas na hora de executar o programa. Eles são desprezadas. No entanto, você precisa acrescentar dois hífen (--) antes da frase. Por exemplo:

```
on exitFrame
  go to the frame -- esta instrução faz o filme entrar em loop no
frame 1
end
```

### ***Comandos de navegação***

| <b>Comando go</b>           |                                |   |
|-----------------------------|--------------------------------|---|
| <b><i>Exemplos</i></b>      | <b><i>Ação</i></b>             | <b><i>Observação</i></b>                                |
| <code>go to frame 30</code> | Salta o filme para o frame 30. | A instrução pode ser resumida para <code>go 30</code> . |

|                                    |   |  |
|------------------------------------|---|--|
| <code>go to the frame</code>       | Mantém o filme no mesmo frame.  | Geralmente é utilizada em conjunto com o evento <code>exitFrame</code> .   |
| <code>go to the frame +1</code>    | Salta o filme para o frame seguinte.                                    | O número também pode ser maior do que 1 e negativo. Por exemplo:<br><code>go to the frame -2</code> faz o filme saltar dois frames para traz.  |
| <code>go to frame "Inicial"</code> | Salta o filme para um frame cujo <i>marker</i> chama "Inicial".         | Pode ser abreviada para <code>go "Inicial"</code> .  |
| <code>go loop</code>               | Salta para o primeiro <i>marker</i> anterior ao frame corrente.         | A idéia de loop vem do fato que se o filme salta para o primeiro <i>marker</i> anterior ao frame corrente, o filme vai voltar a rodar até que o filme atinja novamente o frame corrente. |
| <code>go to marker (+1)</code>     | Salta para o próximo <i>marker</i> .                                    | O número especificado pode ser maior do que 1 e positivo ou negativo. O salto vai ser correspondente ao valor especificado.  |
| <code>go next</code>               | Salta para o próximo <i>marker</i> .                                    |  |
| <code>go previous</code>           | Salta para o <i>marker</i> anterior.                                    |  |
| <code>go movie "filme.dir"</code>  | Salta para o primeiro frame do filme indicado pelo arquivo entre aspas. | A especificação da extensão <code>dir</code> é opcional. No entanto, é bom incluí-la se houver outro arquivo com o mesmo nome dentro da mesma pasta.                                     |

Além do comando de navegação `go`, o Director tem um outro: o comando `play`. Ao contrário de `go`, o comando `play` trabalha em duas rotas: uma de ida e outra de volta. A volta é determinada por uma instrução `play done`. Esta instrução faz o filme voltar para o ponto a partir de onde foi disparada a instrução `play`:

| Comando <code>play</code>                         |  |
|---|--|
| <i>Exemplo</i>                                    | <i>Ação</i>  |
| <code>play frame "Imagem"</code>                  | Salta e roda o frame cujo <i>marker</i> é "Imagem". A instrução pode ser resumida por <code>play "Imagem"</code> . |
| <code>play frame 15</code>                        | Salta e roda o frame 20.   |
| <code>play movie "filme"</code>                   | Salta para o filme cujo nome é "filme.dir" e roda a partir do primeiro frame.                                      |
| <code>play frame "Imagem" of movie "filme"</code> | Salta para um frame cujo <i>marker</i> é "Imagem" de um filme cujo nome é "filme.dir"                              |
| <code>play done</code>                            | Retorna o filme para o frame de onde ocorreu uma instrução <code>play</code> anterior.                             |

## Fundamentos para montagem de scripts

### *Scripts*

Qualquer combinação de termos que fazem parte da linguagem Lingo do Director constitui um script. Esta combinação tem que obedecer uma rígida sintaxe, caso contrário o sistema interpretará como um erro e não poderá compilar a programação. Neste caso, surgirá uma mensagem de erro para que o programador possa interpretar a natureza do problema. Os termos do Lingo são classificados de acordo com o tipo de papel que ocupa na linguagem.

### *Janela Script*

A janela Script é o local onde você deve escrever os seus *handlers*. No Director, cada elemento (cast member, movie, sprite ou frame) pode ter a sua programação. Se, por exemplo, você quiser criar um botão que toque um arquivo de som quando for clicado, é na janela Script deste botão (sprite ou cast member) que deve ser escrita a instrução específica para executar esta ação.

Vamos ver algo mais sobre as janelas Script. Quando você abre esta janela, há uma série de botões que auxiliam o trabalho do programador. Os mais importantes são:



Este botão apresenta uma lista alfabética de todos os termos da linguagem Lingo. Para facilitar a digitação (e evitar erros de datilografia), basta escolher o termo que você quer inserir na sua programação – ele vai aparecer automaticamente.



Assim como anterior mostra uma lista dos termos da linguagem Lingo, porém é classificado por categorias.



Coloca um comentário na linha selecionada



Tira um comentário de um linha de programação.

### *Comandos*

Os comandos são basicamente os termos que inserem uma ação em uma instrução. Por exemplo, `go` e `play` (comandos de navegação), `sound playFile` (comando de multimídia), `delete` (comando de manipulação de strings) etc.

### *Funções*

Funções são termos do Lingo que retornam um valor. Usando funções, você pode construir expressões ou fórmulas. As funções podem ou não ter parâmetros. As funções que possuem parâmetros vão retornar um valor que depende do valor do parâmetro. Uma função, sem parâmetro, e que já vimos anteriormente é a `the frame`. Esta função retorna um valor que representa o frame corrente.

### *Variáveis*

Tecnicamente falando uma variável é um espaço na memória onde pode ser armazenada uma informação. O valor desta informação pode variar com o tempo. Para que esta

informação possa ser encontrada na memória é necessário que o local onde ela se encontra tenha um nome – o nome da variável.

No Director, basicamente a variável pode armazenar valores numéricos ou strings. Uma string é escrita entre aspas; um valor numérico não.

Além destes dois tipos, o Director permite armazenar valores como constantes. As constantes são como as variáveis, mas os seus valores não mudam. Duas constantes muito úteis e frequentemente usadas são `True` e `False`.

Antes de atribuir um valor a uma variável, ela pode ter que ser declarada. Quando, você faz isto, na realidade, está declarando para o sistema qual é o nome dela

As variáveis podem ser Globais ou Locais.

As variáveis globais podem ser usadas em todo o filme, porque elas se caracterizam por segurar o seu valor enquanto ele estiver rodando.

Uma variável local não tem esta capacidade. O valor de uma variável local é retido na memória durante a execução de um *handler*. Quando este acabou de ser executado, o valor da variável local é simplesmente perdido – a variável deixa de existir.

Você deve declarar uma variável global desta maneira:

```
global numPessoas
```

```
global cont
```

As variáveis globais devem ser declaradas em todos os objetos que forem usadas e elas podem ser utilizadas em qualquer *handler* de qualquer script em um filme.

Uma variável local, por outro lado, não precisa ser declarada. O próprio Director se encarrega de interpretar o que é ou não é uma variável em um *handler*.

### ***Normalização sobre variáveis***

O nome de uma variável pode ter até 256 caracteres e pode incluir letras, números ou o travessão (`_`). Outros caracteres não são permitidos. No entanto, o nome da variável só pode ser iniciado por uma letra. O Director não diferencia letras maiúsculas de minúsculas, mas não use acentuação nas letras. Uma boa dica é que o nome da variável expresse aquilo que ela vai armazenar. Por exemplo: `totalFrames`. Uma tradição entre os programadores é expressar o nome por mais de uma palavra, dividindo-as com o travessão ou com uma letra maiúscula, deixando a primeira letra como minúscula. Assim: `numeroPessoas` ou `numero_pessoas`.

Formatos corretos para variáveis:

```
numSprites, var2, O_Filme
```

Formatos incorretos para variáveis:

```
num Sprites, 2var, O-Filme
```

### ***Atribuindo valores a variáveis ou strings***

Há três métodos para atribuir valores a variáveis. Com os comandos `put` e `set`:

```
put expressão into variável
```

Por exemplo:

```
put 50 into 50
put "segunda-feira" into field "data"
```

Com `set`:

```
set variável to expressão

ou

set variável = valor
```

Por exemplo:

```
set cont to 50
set x = 1
set the text of member "data" to "segunda-feira"
```

E finalmente:

```
variavel = valor
```

Por exemplo:

```
x = 1
```

### ***Message Window***

Esta janela é um poderoso recurso para você executar instruções sem que o filme esteja rodando. Com ela você pode obter informações sobre o filme ou o sistema, alterar valores, modificar propriedades ou características de objetos etc.

Para abrir a Message Window, selecione Window ⇒ Message ou simplesmente pressione Ctrl+M.

A partir de então você pode executar qualquer instrução, desde que use corretamente a linguagem Lingo. Após digitar a instrução, pressione Enter para executá-la. Por exemplo, digite:

```
put 8 * 8
```

O Director responderá na mesma janela

```
-- 64
```

Outro exemplo:

```
set x = 1
put x
```

E o Director responderá

```
-- 1
```

Mais um exemplo:

```
put the platform
```

Você visualiza a seguinte resposta:

```
-- "Windows,32"
```

Ainda mais um exemplo:

```
alert "Bom dia!"
```

A resposta do Director é mostrar uma janela padrão apresentando a string “Bom Dia!”, com um botão OK.

Repare que na janela Message, você também tem à disposição os botões  e .

### Operadores

Os operadores são palavras-chaves que permitem realizar operações sobre expressões. Só para ficar inicialmente no exemplo mais clássico, o operador + é o operador de adição do Director que permite realizar operações aritméticas entre dois valores numéricos.

Veja na tabela abaixo os operadores aritméticos do Director:

| Operadores Aritméticos |                            |                      |                |  |
|------------------------|----------------------------|----------------------|----------------|--|
| <i>Operador</i>        | <i>Nome</i>                | <i>Exemplo</i>       | <i>Retorna</i> | <i>Operação</i>                                      |
| +                      | Adição                     | set x = 2 + 3        | 5              | Adiciona valores                                     |
| -                      | Subtração                  | set x = 5 - 3        | 2              | Subtrai valores                                      |
| -                      | Negativo                   | set x = -7           | -7             | Inverte o valor                                      |
| *                      | Multiplicação              | set x = 5 * 7        | 35             | Multiplica valores                                   |
| /                      | Divisão inteira            | set x = 9 / 4        | 2              | Divide valores e coloca o resultado como inteiro     |
| /                      | Divisão fracional          | set x = 9.0 / 4.0    | 2.2500         | Divide valores e coloca o resultado como fracionário |
| MOD                    | Resto                      | set x = 7 MOD 5      | 2              | Retorna o resto de uma divisão                       |
| integer                | Conversão para inteiro     | set x = integer(4.2) | 4              | Converte um valor fracionário para inteiro           |
| float                  | Conversão para fracionário | set x = float(4)     | 4.0            | Converte um valor inteiro para flutuante             |

*Operadores de comparação:* a maioria das programações envolvem a comparação de valores para expressões condicionais.

Por exemplo: se x for maior do que 1, vá para o frame 2.

O resultado desta expressão só pode ser verdadeiro ou falso e em função da resposta você pode pedir para executar uma ou outra instrução.

Os operadores de comparação possuem o papel de disponibilizar estas operações.

| Operadores Aritméticos |              |                |                |                                  |
|------------------------|--------------|----------------|----------------|----------------------------------|
| <i>Operador</i>        | <i>Nome</i>  | <i>Exemplo</i> | <i>Retorna</i> | <i>Operação</i>                  |
| =                      | Igualdade    | put 3 = 3      | 1 (true)       | Compara se valores são iguais    |
| <>                     | Desigualdade | put 3 <> 3     | 0 (false)      | Compara se valores são desiguais |

|          |                |                                  |           |   |
|----------|----------------|----------------------------------|-----------|---|
| >        | Maior que      | put 4 > 3                        | 1 (true)  | Compara se o primeiro valor é maior do que o segundo                |
| <        | Menor que      | put 4 < 3                        | 0 (false) | Compara se o primeiro valor é menor do que o segundo                |
| >=       | Maior ou Igual | put 4 >= 4                       | 1 (true)  | Compara se o primeiro valor é maior ou igual ao segundo             |
| <=       | Menor ou igual | put 4 <= 3                       | 0 (false) | Compara se o primeiro valor é menor ou igual ao segundo             |
| CONTAINS | Contém         | put "Director"<br>CONTAINS "Dir" | 1 (true)  | Compara duas strings e verifica se a primeira contém a segunda.     |
| STARTS   | Inicia com     | put "Brasil" STARTS<br>"Bra"     | 1 (true)  | Compara duas strings e verifica se a primeira começa com a segunda. |

*Operadores Lógicos:* Através dos operadores lógicos, você pode inverter ou compor mais de uma expressão comparativa. Eles podem testar mais de uma comparação.

Freqüentemente você usa os operados lógicos junto com estruturas condicionais, como `if ... then ... else` (veja mais adiante sobre estruturas condicionais).

| <b>Operadores Lógicos</b> |             |                                     |                |  |
|---------------------------|-------------|-------------------------------------|----------------|--|
| <i>Operador</i>           | <i>Nome</i> | <i>Sintaxe</i>                      | <i>Exemplo</i> | <i>Descrição</i>   |
| AND                       | Porta E     | if <expressão1><br>and <expressão1> | if x=1 and y=0 | Somente se a expressão 1 e a expressão 2 forem verdadeiras a expressão como um todo será verdadeira. |
| OR                        | Porta OU    | if <expressão1> or<br><expressão1>  | if x=1 or y=0  | Se a expressão 1 ou a 2 for verdadeira, a expressão como um todo será verdadeira.                    |
| NOT                       | Inversora   | if not <expressão>                  | if not (1 > 2) | Inverte o valor da expressão.  |

*Operadores de Concatenação:* São operadores que ligam duas strings

| <b>Operadores de Concatenação</b> |                      |                |                                 |
|-----------------------------------|----------------------|----------------|---------------------------------|
| <i>Operador</i>                   | <i>Exemplo</i>       | <i>Retorna</i> | <i>Descrição</i>                |
| &                                 | put "Versão" & "um"  | "Versãoum"     | Liga duas strings.              |
| &&                                | put "Versão" && "um" | "Versão um"    | Liga duas strings com um espaço |

*Outros operadores:* são operadores com funções diversas:

| <b>Outros Operadores</b> |             |                               |   |  |
|--------------------------|-------------|-------------------------------|---|--|
| <b>Operador</b>          | <b>Nome</b> | <b>Sintaxe</b>                | <b>Exemplo</b>  | <b>Descrição</b>   |
| --                       | Comentário  | -- <comentário>               | -- Isto é apenas<br>-- um comentário.<br>-- Não é executado | Tudo o que vem após o comentário não é executado pelo sistema. |
| ()                       | Parênteses  | (expressão)                   | put (3+2)*3   | Força a porção de uma expressão a ser calculada antes          |
| @                        | Caminho     | @                             | @\pasta   | O operador define o nome do caminho do filme corrente.         |
| #                        | Símbolo     | #NomeDoSímbolo                | #OkCancel   | Define e identifica um símbolo.                                |
| []                       | Listas      | [entrada1,<br>entrada2, ... ] | set x = []  | Usado para entradas de uma lista.                              |

### **Propriedades**

Todo objeto no Director tem propriedades: um cast member, um sprite, um frame ou o próprio movie (filme). Propriedades nada mais são do que atributos de objetos. Por exemplo, um sprite tem como atributo a sua cor de fundo: vermelha, amarela, azul etc. Um cast member tem como atributo o seu nome. E assim por diante.

Em geral, alteramos as propriedades de um objeto usando a própria interface do Director. Por exemplo: quando mudamos o nome de um cast member usando a janela Cast Member Properties. Mas você também pode alterar a propriedade de um objeto usando Lingo.

Há duas formas de atribuir novos valores para propriedades de objetos no Director. A primeira usando o comando `set`. Neste caso, a sintaxe básica é:

```
set the <propriedade> of <objeto> <referência> to <valor>
```

Por exemplo:

```
set the cursor of sprite 1 to 4 - muda o cursor para ampulheta
set the locH of sprite 1 to 200 - reposiciona o sprite na horizontal
```

A segunda maneira de mudar propriedades de objetos é conhecida como notação por ponto:

```
<objeto>.<propriedade> = <valor>
```

Por exemplo:

```
Member("campo").text = "Bom dia!" -- altera o texto do member "campo"
member(3).backcolor = 4 -- muda a cor de fundo do member 3
sprite(5).forecolor = 56 -- muda a cor do sprite que está na camada 5
```

**Obs:** Para saber o valor atual da propriedade de um objeto, use o comando na janela Message (Ctrl+M). Por exemplo: `put the locV of sprite 5 --200`

### ***Controlando sprites por Lingo***

Em geral, um sprite é controlado pelo Score. Quando ele se movimenta ou é redimensionado no stage, ele o faz segundo os valores especificados no Score. No entanto há uma maneira de tirar este controle do Score e colocar o controle completamente sob responsabilidade do Lingo: usando o comando `puppetSprite`:

O comando `puppetSprite` possui dois parâmetros. O primeiro indica o canal que receberá o controle através do Lingo. O segundo parâmetro indica se o controle vai ser acionado (TRUE ou 1) ou não (FALSE ou 0).

```
puppetSprite 3, TRUE - coloca o controle do sprite 3 para o Lingo
```

```
puppetSprite 2, 0 -- tira o controle do sprite 2 do Lingo
```

Se o controle estiver sob mando do Score, isto não significa que se você mudar uma propriedade de um sprite por Lingo, esta alteração não vai ocorrer. Mas se o sprite sumir momentaneamente e voltar a aparecer mais para frente, ele volta segundo os valores iniciais, fornecidos ao Score. Assim, para garantir que uma mudança feita no sprite persista, o melhor a fazer é tirar o controle do Score, usando o comando `puppetSprite`.

Mas cuidado: antes de usar comandos de navegação (`go` ou `play`) coloque o controle do sprite novamente para o Score (`puppetSprite 2, FALSE`).

### ***Estruturas condicionais***

No Lingo, assim como em qualquer outra linguagem de programação, existe a estrutura condicional `if ... then ... else`:

```
if <expressãoLógica> then <declaração>
```

ou

```
if <expressãoLógica> then <declaração1>
else <declaração2>
end if
```

ou

```
if <expressãoLógica> then
    <declaração>
end if
```

ou

```
if <expressãoLógica> then
    <declaração1>
else
    <declaração2>
end if
```

ou

```
if <expressãoLógica1> then
    <declaração1>
else if <expressãoLógica2> then
    <declaração2>
else if <expressãoLógica3> then
    <declaração3>
end if
```

A estrutura avalia o resultado da expressão lógica, especificada por <expressãoLógica>, e:

- Se a expressão for verdadeira, o Director executa a declaração que vem a seguir de `then`.
- Se a expressão for falsa, o Director executa a declaração que vem depois de `else`. Se não houver `else` na estrutura, o Director não faz nada.

Por exemplo:

```
if x = 1 then
    go to frame 30
else
    go to frame 60
end if
```

Neste exemplo, se o valor armazenado na variável `x` for igual a 1, o Director move o filme para o frame 30. Se isto não for verdade, ou seja, se `x` for diferente de 1, o Director avança para o frame 60.

Repare que a estrutura `if ... then ... else` tem algumas variações. Basicamente ela pode ou não ser representada com `else`. Quando `else` não aparece, significa que só há uma declaração se a expressão for verdadeira. Nada acontece se esta for falsa.

*Em tempo: else em inglês quer dizer senão.*

Repare também na última variação de estrutura:

```
if <expressãoLógica1> then
    <declaração1>
else if <expressãoLógica2> then
    <declaração2>
else if <expressãoLógica3> then
    <declaração3>
end if
```

Aqui a estrutura apresenta diversos níveis. Na realidade, estes níveis podem ser aumentados indiscriminadamente. Costuma-se chamar estas estruturas de *ifs aninhados*. Uma alternativa aos *ifs aninhados* é a estrutura `case`, que apresenta a seguinte sintaxe:

```
case <expressão> of
    expressão1 : Declaração(ões)
    expressão2 :
        multiplasDeclarações
        .
        .
    expressão3, expressão4 :
        Declaração(ões)(s)
    {otherwise Declaração(ões)}
end case
```

### ***Estruturas de repetição***

As estruturas de repetição repetem determinada declaração enquanto uma condição for verdadeira. Uma destas estruturas é a `repeat`:

```
repeat while <condição>
    <declaração>
end repeat
```

A <declaração> pode incluir qualquer instrução Lingo que você queira executar. Em <condição> você pode incluir qualquer condição booleana. Inclusive uma função que verifique se um evento reconhecido pelo Lingo foi ou não enviado.

Por exemplo:

```
repeat while mouseDown
    beep 1
end repeat
```

No exemplo acima, enquanto o botão do mouse estiver pressionado, será emitido um sinal sonoro padrão. Outras funções também testam determinados eventos:

| <b>Funções que verificam eventos</b> |  |
|--------------------------------------|--|
| <b><i>Função</i></b>                 | <b><i>O que ela determina</i></b>                |
| the controlDown                      | Se a tecla Ctrl está sendo pressionada           |
| the mouseUp                          | Se o botão do mouse não está sendo pressionado   |
| the pauseState                       | Se o filme está parado                           |
| the shiftDown                        | Se a tecla Shift está sendo pressionada          |
| the stillDown                        | Se o botão do mouse ainda está sendo pressionado |

Uma estrutura de repetição é representada por repeat with:

```
repeat with <variável> = <início> to <fim>
    [declarações...]
end repeat
```

Neste caso uma determinada declaração vai se repetir um certo número de vezes, especificado por <início>, até completar número, determinado por <fim>.

Por exemplo:

```
repeat with x = 1 to 20
    puppetSprite x, TRUE
end repeat
```

Se você quiser que o valor da variável assuma valores decrescentes, inclua o parâmetro down da estrutura:

```
repeat with <variável> = <início> down to <fim>
    [declarações...]
end repeat
```

Por exemplo:

```
repeat with i = 30 down to 15
    sprite(6).memberNum = (10 + i)
end repeat
```

### ***Compatilhando scripts***

Como dissemos anteriormente (no item *Que script tem prioridade?*) no Director existe uma hierarquia entre os seus objetos, que define qual deles vai receber em primeiro lugar a mensagem de um determinado evento. O último objeto da hierarquia é o movie, portanto é o último objeto a receber a mensagem. Por outro lado, quando um *handler* está no movie, ele pode ser executado de qualquer lugar de seu filme, desde que um objeto de hierarquia

inferior não tenha um *handler* para o evento correspondente. Por exemplo, vamos dizer que este *handler* está no script do movie:

```
on exitFrame
  go to the frame
end
```

Por estar no movie, este *handler* vai ser executado todas as vezes que você entrar em qualquer um dos frames de seu filme. Embora, nenhum dos frames possua programação, o *handler* acima vai funcionar para todos eles, porque está no script de um objeto de hierarquia superior – no movie.

### ***Criando sub-rotinas***

Graças à existência desta hierarquia, você pode criar sub-rotinas, colocá-las no script do movie e chamá-las a partir de qualquer objeto de seu filme. Por exemplo:

```
on <subrotina>
  <declarações...>
end
```

<subrotina> é o nome da rotina que você criou no script do movie.

Para executá-la é necessário que você mande uma mensagem para ela a partir do script de qualquer objeto do seu filme. Por exemplo:

```
on mouseUp
  subrotina
end
```

Neste caso, quando você clicar o botão do mouse em cima do objeto que possui este *handler* as declarações colocadas em `on <subrotina>` serão executadas.

A grande vantagem disto é poder compartilhar este *handler* entre diversos objetos. Basta fazer uma chamada a ele.

Este *handler*, no entanto, tem que ser suficientemente universal a ponto de servir a todos os objetos. Se você tiver uma programação deste tipo:

```
on nomedoevento
  go to frame 30
end
```

todas as chamadas a `nomedoevento` farão o filme se deslocar para o frame 30. Se você quiser que em um determinado momento o filme vá para o frame 31, você não poderá usar este *handler*; você terá que criar um novo que faça isto. Por outro lado, dê uma olhada no *handler* abaixo:

```
on nomedoevento
  go to the frame +1
end
```

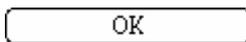
Neste caso, há uma diferença, o filme não vai se deslocar sempre para o mesmo frame. Ele vai se deslocar para um frame que depende do frame em que o filme está. Na medida do possível você deve trabalhar com instruções que não explicitem parâmetros absolutos, como `frame 30`. A não ser que você queira exatamente isto para todos os objetos que executarem esta ação. Quando você criar *handlers* em sub-rotinas, provavelmente você estará trabalhando com parâmetros que relativize o seu valor. Como `go to the frame +1`.

### ***Alterando o estado do botão***

Os botões do tipo *Push Button*, que podem ser traduzidos literalmente como *botões de pressionar* e que são o formato mais convencional de botões na informática, possuem usualmente quatro estados:

- estado normal – é quando o botão está em seu estado inercial.
- estado pressionado ou rebaixado – é quando o botão está rebaixado, em função de ter sido pressionado.
- estado aceso – é quando o ponteiro do mouse entra nos limites do botão, e este muda a sua cor ou é iluminado.
- estado desabilitado – é quando o botão não pode ser pressionado, pois não está habilitado. Em geral, o botão adquire um coloração acinzentada, com seu ícone representado em baixo relevo.

Um botão pode ser criado no Director selecionando `Insert ⇒ Control ⇒ Push Button`. O Director vai criar um objeto deste tipo:



Se você pressioná-lo quando o filme estiver rodando, ele vai mudar de cor, mostrando um efeito de clique.

Embora este botão possa resolver o seu problema, em geral, os softwares hiperídia costumam embutir em suas interfaces botões com desenhos mais personalizados. Se você também tiver esta intenção, você mesmo vai ter que construí-los. Programas como o Adobe PhotoShop ou o CorelDraw permitem que você desenvolva imagens que simulem um botão. Posteriormente, você pode importá-los para o seu filme.

Para que o botão apresente mudanças de estado – como descrito acima – é necessário que você crie uma imagem para cada um: um para o estado normal, uma para o rebaixado, um para o aceso e um para o desabilitado. E depois, você precisa criar uma programação em Lingo para que ele funcione tal qual um botão.

Na realidade, tudo o que você vai fazer é criar um script que esconda ou mostre cada uma das imagens, em função do estado na qual você queira que o botão funcione no momento. Se você quiser que o botão esteja no estado normal, basta mostrar a imagem que você criou para o estado normal do botão e esconder as outras imagens referentes a ele. Caso você queira que o botão alterne para o estado rebaixado, basta que no momento em que o usuário pressionar o botão, você esconda a imagem para o estado normal e mostre a imagem para o estado rebaixado. E assim por diante.

Para criar esta programação você deve usar a propriedade `visible`. Esta propriedade pode assumir dois valores: `TRUE` ou `FALSE`. Quando assumir o valor `TRUE`, você estará mostrando o sprite no stage. Quando for `FALSE`, você estará escondendo o sprite. A sintaxe básica para isto é:

```
sprite(<qualSprite>).visible
```

onde `<qualSprite>` indica o número do canal ocupado pelo sprite.

Por exemplo:

```
sprite(3).visible = TRUE
```

Uma outra maneira de criar este mesmo efeito é usar a propriedade `memberNum`. Esta propriedade indica qual o número do cast member associado a um sprite. Esta propriedade é extremamente útil para alternar o cast member de um sprite.

No nosso exemplo, podemos dizer que o sprite que ocupa o canal 1 vai mudar do cast member 1 (estado normal) para o cast member 1 (estado rebaixado), quando o botão for pressionado e voltar do cast member 2 para o 1 quando o botão do mouse for liberado. Ou seja, o cast member 1 armazena a imagem correspondente ao botão no estado normal e o cast member 2 armazena a imagem correspondente ao botão no estado rebaixado.

Por exemplo, veja esta programação do sprite que está no canal 1. Inicialmente, o sprite 1 é uma cópia do cast member 1:

```
on mouseUp
  go to the frame+1
end

on mouseLeave
  sprite(1).memberNum = 1
end

on mouseDown
  sprite(1).memberNum = 2
end

on mouseEnter
  sprite(1).memberNum = 3
end
```

Onde, o cast member 1 é a imagem correspondente ao botão no estado normal, o cast member 2 é a imagem do botão no estado rebaixado e o cast member 3 é a imagem do botão no estado aceso.

### ***Manipulando sprites através do Lingo***

Usar a sintaxe básica do Lingo para alterar propriedades de objetos é a maneira mais básica para manipular sprites por programação:

```
<sprite>(<referência>).<propriedade> = <valor>
```

Já tínhamos visto isto anteriormente ao alterar a propriedade `visible` de um sprite:

```
sprite(3).visible = TRUE -- mostra o sprite 3
sprite(2).loc = point(100,50) -- posiciona sprite 2 em 100,50
sprite(5).moveableSprite = TRUE -- permite movimentar sprite 5
```

No entanto você deve tomar muito cuidado ao alterar a propriedade de sprites, porque eles são por padrão controlados pelo Score. Se, por exemplo, você alterar a cor de um sprite por Lingo, mas provisoriamente ele deixar de aparecer no stage, quando retornar à tela, ele volta a ter a sua cor original – aquela que ele tinha antes de ser alterado. O Score sempre mostra um sprite no stage segundo as suas propriedades originais. Se você quiser que o Score deixe de controlar o sprite, você deve solicitar. E isto é feito por Lingo, através de um comando chamado `puppetSprite`:

```
puppetSprite numCanal, estado
```

Onde:

`numCanal` número do canal no Score afetado pelo comando `puppetSprite`.  
`estado` dois valores: `TRUE` ou `FALSE`. `TRUE` fornece o controle do canal especificado para o Lingo. `FALSE` dá o controle para o Score.

Por exemplo:

```
puppetSprite 5, TRUE
```

### ***Posição do mouse***

As funções `mouseH` e `mouseV` fornecem respectivamente as posições das coordenadas `x` e `y` do cursor. O sistema de coordenadas do Director é dado em função do stage. A posição 0,0 é o canto superior esquerdo do stage. A coordenada `x` – horizontal – aumenta da esquerda para a direita. A coordenada `y` – vertical – aumenta de cima para baixo. O valor retornado é um número em pixels. Estas funções podem ser testadas, mas você não pode atribuir um valor para elas. A sintaxe delas é a seguinte:

```
the mouseH -- posição do cursor na coordenada x
the mouseV -- posição do cursor na coordenada y
```

Por exemplo:

```
sprite(1).locH = the mouseH -- posiciona o sprite 1 na
                        coordenada x do cursor
```

### ***Limitando movimentos de sprite***

Algumas vezes você pode querer limitar o movimento de um sprite dentro de certos limites do stage. Um modo bastante simples de realizar este tipo de manipulação é usando a propriedade `the constraint of sprite`. Ela limita o movimento de um sprite dentro dos limites de um outro:

```
sprite(qualSprite).constraint
```

Por exemplo:

```
on startMovie
  puppetSprite 2, True
  sprite(2).moveableSprite = True
  sprite(2).constraint = 1 -- faz o sprite 2 só se movimentar
                          -- dentro dos limites do sprite 1
end
```

As funções `constrainH` e `constrainV` são semelhantes. No entanto elas avaliam se um sprite especificado está dentro de um limite dado e retorna um valor seguindo os seguintes critérios. Para a função `constrainH`:

```
constrainH (qualSprite , valor)
```

- Quando o valor estiver entre as coordenadas esquerda e direita, o valor retornado é o valor especificado.
- Quando o valor é menor do que a coordenada esquerda horizontal em que o sprite se encontra, o valor retornado é o valor da coordenada esquerda.

- Quando o valor é maior do que a coordenada direita horizontal em que o sprite se encontra, o valor retornado é o valor da coordenada direita.coordinate.

Por exemplo, o sprite tem suas coordenadas esquerda e direita respectivamente colocadas nas posições 30 e 70.

```
put constrainH(1, 20)
-- 30
put constrainH(1, 55)
-- 55
put constrainH(1, 100)
70
```

A declaração abaixo limita a posição da coordenada horizontal do sprite 1 pela do sprite 2:

```
sprite(1).locH = constrainH(2, the mouseH)
```

A função `constrainV` funciona da mesma maneira do que a função `constrainH`; só o que foi dito para a coordenada  $x$  deve ser traduzido para a coordenada  $y$ .

Finalmente há ainda a função `intersect`. Ela avalia se um sprite está entrando dentro de outro:

```
sprite sprite1 intersects sprite2
```

A função compara a posição dos dois sprites . Ela retorna `TRUE` se os limites do sprite 1 tocou nos limites do sprite 2. Caso contrário retorna `FALSE`.

Por exemplo:

```
if sprite 1 intersects sprite 2 then alert "Você me tocou!"
```

### ***Manipulando o formato do cursor***

Nos sistemas operacionais de hoje, você encontra uma série de cursores com formato padrão: A flechinha em geral é o formato usual, mas em alguns momentos, ele se altera para desenhos que indiquem uma determinada situação, como a ampulheta do Windows ou relógio do Mac, que representam um estado de espera.

No Director você pode atribuir estes formatos ao cursor no momento que desejar com a propriedade `the cursor of sprite` ou o comando `cursor`. Os valores abaixo especificam os seguintes cursores:

| <b>Código de cursores</b> |                         |
|---------------------------|-------------------------|
| <b><i>Código</i></b>      | <b><i>Descrição</i></b> |
| 0                         | Cursor padrão           |
| -1                        | Flechinha ou apontador  |
| 1                         | I-beam                  |
| 2                         | Crosshair               |
| 3                         | Crossbar                |
| 4                         | Ampulheta ou relógio    |
| 200                       | Cursor fica invisível   |

Alternativamente você pode atribuir uma imagem, colocada na janela Cast, como cursor.

A propriedade `cursor of sprite` tem a seguinte sintaxe:

```
sprite(qualSprite).cursor = queCursor
ou
sprite(qualSprite).cursor = [númeroCast, númeroMáscaraCast]
```

Onde:

|                   |   |
|-------------------|---|
| qualSprite        | número do sprite que altera o cursor                        |
| queCursor         | código correspondente ao formato de cursor                  |
| númeroCast        | número do cast que vai funcionar como cursor                |
| númeroMáscaraCast | número do cast que vai funcionar como máscara para o cursor |

Esta propriedade de sprite determina o cursor que é usado quando o ponto quente dele é apontado para o sprite especificado pelo número inteiro representado em `qualSprite`. O cursor permanece até você alterar novamente o código do cursor, como por exemplo para 0 – cursor padrão.

Por exemplo:

```
sprite(20).cursor = 4 -- muda cursor para o formato do relógio
```

Use a sintaxe `[númeroCast, númeroMáscaraCast]` para especificar o número de um cast member para usar como um cursor alternativo e sua máscara opcional.

Para usar cursores alternativos, o cast member que você especificou deve ser padrão 1-bit e no máximo deve ter 16 x 16 pixels. Se for maior, o Director corta a parcela além dos 16 x 16 pixels, a partir do canto superior esquerdo da imagem. O ponto de registro (registration point) da imagem é o ponto quente (hot spot) do cursor.

Já o comando `cursor` muda formato do cursor para um código padrão ou cast member que é especificado para o cursor. O comando `cursor` mantém o formato atribuído até você pedir para alterá-lo novamente. Sua sintaxe é a seguinte

```
cursor qualCursor
ou
cursor [númeroCast, númeroMáscaraCast]
```

Por exemplo:

```
cursor 1 -- cursor muda para o formato I-beam
cursor [15,16} -- cursor altera o formato para a imagem correspondente ao
                cast member 15

on startMovie
    cursor 4 -- começa o filme com o cursor do relógio ou ampulheta
end
```

### ***Manipulando transições através do Lingo***

Se você deseja controlar efeitos de transição através de Lingo e tirar o controle do Score, você deve usar o comando `puppetTransition`. Este comando executa a transição especificada entre o frame corrente e o próximo:

```
puppetTransition código [, tempo] [, tamanho] [, áreaMudança]
```

Onde:

código            indica o código da transição. Veja tabela abaixo

**tempo** é o tempo de duração da transição. O valor 1 corresponde a ¼ segundo.  
**tamanho** indica a grade de transição. O mínimo é 1 (pixel). O máximo, 128.  
**áreaMudança** indica se a transição será executada somente nas áreas que mudaram entre um frame ou outro (**FALSE**) ou no stage inteiro (**TRUE**). É o padrão.

A velocidade da transição não é consequência somente do tempo, mas de uma relação entre este e o tamanho. Se a transição for efetuada em 1 pixel, a velocidade da transição será naturalmente mais lenta do que se a transição fosse executada em uma área maior.

| <b>Códigos de Transições</b> |                           |               |                               |
|------------------------------|---------------------------|---------------|-------------------------------|
| <b>Código</b>                | <b>Transição</b>          | <b>Código</b> | <b>Transição</b>              |
| 1                            | Wipe right                | 27            | Random rows                   |
| 2                            | Wipe left                 | 28            | Random columns                |
| 3                            | Wipe down                 | 29            | Cover down                    |
| 4                            | Wipe up                   | 30            | Cover down, left              |
| 5                            | Center out, horizontal    | 31            | Cover down, right             |
| 6                            | Edges in, horizontal      | 32            | Cover left                    |
| 7                            | Center out, vertical      | 33            | Cover right                   |
| 8                            | Edges in, vertical        | 34            | Cover up                      |
| 9                            | Center out, square        | 35            | Cover up, left                |
| 10                           | Edges in, square          | 36            | Cover up, right               |
| 11                           | Push left                 | 37            | Venetian blinds               |
| 12                           | Push right                | 38            | Checkerboard                  |
| 13                           | Push down                 | 39            | Strips on bottom, build left  |
| 14                           | Push up                   | 40            | Strips on bottom, build right |
| 15                           | Reveal up                 | 41            | Strips on left, build down    |
| 16                           | Reveal up, right          | 42            | Strips on left, build up      |
| 17                           | Reveal right              | 43            | Strips on right, build down   |
| 18                           | Reveal down, right        | 44            | Strips on right, build up     |
| 19                           | Reveal down               | 45            | Strips on top, build left     |
| 20                           | Reveal down, left         | 46            | Strips on top, build right    |
| 21                           | Reveal left               | 47            | Zoom open                     |
| 22                           | Reveal up, left           | 48            | Zoom close                    |
| 23                           | Dissolve, pixels fast *   | 49            | Vertical blinds               |
| 24                           | Dissolve, boxy rectangles | 50            | Dissolve, bits fast *         |
| 25                           | Dissolve, boxy squares    | 51            | Dissolve, pixels *            |
| 26                           | Dissolve, patterns        | 52            | Dissolve, bits *              |

(\*) Indica transições que não funcionam em monitores configurados em 32 bits.

Por exemplo:

```
puppetTransition 1          -- executa uma transição Wipe Right
puppetTransition 51, 4, 30, TRUE - executa transição "Dissolve, pixels",
em 1 segundo, com tamanho 30 pixels, em todo stage (TRUE).

on exitFrame
  puppetTransition 23, 1, 20
end
```

### ***Manipulando textos com Lingo***

Quando falamos em manipular textos com Lingo, estamos nos referindo aos fields. Como dissemos anteriormente os textos criados com a ferramenta Text e os textos criados pela janela Paint não podem ser alterados no modo de execução do filme.

Há uma série de propriedades que permitem manipular certas características dos fields, como cor, fonte, estilo, tamanho etc. Vamos a elas:

| <b>Propriedades de fields</b> |   |  |
|-------------------------------|---|--|
| <b><i>Propriedade</i></b>     | <b><i>Valores</i></b>                     | <b><i>Descrição</i></b>  |
| the fontStyle of member       | "plain", "bold",<br>"italic", "underline" | Fornece o estilo do field, como normal (plain), bold ( <b>negrito</b> ), itálico ( <i>italic</i> ), sublinhado ( <u>underline</u> ). Para atribuir mais de um estilo ao mesmo tempo, você deve usar uma vírgula:<br>set the fontStyle of field 1 to "bold, italic" |
| the fontSize of member        | 1 a 1000                                  | Estabelece o tamanho da fonte em pontos.   |
| the font of member            | Uma fonte corrente do sistema.            | Especifica a fonte usada no field.   |
| the foreColor of member       | 0 a 255.                                  | Especifica a cor usada no texto do field.  |
| the backColor of member       | 0 a 255.                                  | Especifica a cor usada no field.   |
| the alignment of member       | "left", "right",<br>"center"              | Especifica o alinhamento do texto no field.  |
| the border of member          | 0 a 127                                   | Largura da borda do field.   |
| the boxDropShadow of member   | 0 a 127                                   | Tamanho em pixels da sombra que aparece no canto inferior direito do quadrado que delimita o field.  |
| the DropShadow of member      | 0 a 127                                   | Tamanho em pixels da sombra que aparece no canto inferior direito do texto do field.   |
| the margin of member          | 0 a 127                                   | Tamanho da margem entre o texto e o quadrado que limita o field.   |

Para atribuir uma destas propriedades ao field, use a seguinte sintaxe, com o comando `set`:

```
member(<referência ao member>).<propriedade> = <valor>
```

Você pode usar alternativamente a palavra `field` em vez de `member` e o operador `=` em vez de `to`.

Por exemplo:

```
member(1).font = "Arial"
member("texto").fontStyle = "bold, italic, underline"
set the foreColor of field 3 to 100
member(5).fonte = "Times New Roman"
set the fontSize of field "Nome" to 14
member(1).alignment = "center"
```

### ***Atribuindo textos a fields por Lingo***

Há basicamente duas maneiras de colocar textos em fields através de Lingo:

- Usando o comando `put`
- Usando o comando `set` em fields já existentes

Sintaxe para `put`:

```
put <string> into member <referência ao member>
```

Por exemplo:

```
put "texto" into field 1
```

Neste exemplo, a palavra "texto" vai aparecer no field 1. Se este field não existir, ele será criado.

Sintaxe por ponto:

```
Member(<referência ao member>).text = <string>
```

Por exemplo:

```
member("nome").text = "texto"
```

Sintaxe para `set`:

```
set the text of member <referência ao member> to <string>
```

Por exemplo:

```
set the text of member "nome" to "texto"
```

Neste exemplo, a palavra "texto" vai aparecer no member "nome". Este field tem que existir previamente. Você não pode tentar alterar o texto de um field que ainda não existe.

### ***Manipulando partes do texto***

Você não precisa necessariamente alterar todo o conteúdo de um field. Você pode alterar, acrescentar ou eliminar partes de um texto, usando as seguintes palavras-chaves:

| <b>Mudando partes do texto</b> |   |
|--------------------------------|---|
| <b><i>Termo</i></b>            | <b><i>Descrição</i></b>   |
| char                           | Altera, adiciona o apaga um caracter de um texto.   |
| word                           | Altera, adiciona o apaga uma palavra de um texto.   |
| item                           | Altera, adiciona o apaga um item de um texto. O Director considera um item aquilo que está entre uma vírgula e outra de uma string. |
| line                           | Altera, adiciona o apaga uma linha de um texto.   |

Você deve usar o comando `put` para fazer estas alterações:

```
put <valor> into <termo> <número> of member <referência ao member>
```

Por exemplo:

```
put "z" into char 2 of field "nome"
put "segunda linha" into line 2 of member 3
put "palavra" into word 4 of field 1
put "azul" into item 5 of member "listaCores"
```

Os termos `AFTER` e `BEFORE` permitem acrescentar strings sem eliminar textos já existentes. O termo `after` acrescenta uma string depois de uma posição especificada pelos termos acima (`char`, `word`, `item`, `line`). O termo `before` acrescenta uma string antes de uma posição determinada.

Por exemplo:

```
put "esqueci este texto" after word 5 of field 2
put "a" before char 15 of member "texto"
put "vermelho" after item 6 of member "listaCores"
put "Mais uma linha" after line 1 of field 5
```

Quando você acrescenta uma string entre um texto de um field, a ordem dos termos necessariamente se altera. Por exemplo: se você insere uma nova linha, após a quarta linha de um texto que possui 5, a quarta linha passa a ser a quinta e a quinta passa a ser a sexta.

Para eliminar um texto ou parte de textos use os termos `NULL` ou `EMPTY` ou o comando `delete`:

```
put NULL into field 1
put EMPTY into member "Texto"
delete field 2
delete char 3 of member "Nome"
delete word 3 to 5 of member 6
put EMPTY into char 3 to 5 of field "Texto"
```

Você também pode usar os termos `char`, `word`, `item` e `line`, juntamente com o comando `set`, para mudar propriedades de partes do texto:

```
set the foreColor of word 6 of field "Texto" to 6
set the font of char 6 of member 1 to "Times"
```

Para saber o número de termos de um texto use a função `the number`. Neste exemplo, as cores de todas as palavras de um texto são mudadas aleatoriamente pela função `random`:

```
on mouseUp
  repeat with i = 1 to the number of words in field 1
    set the forecolor of word i of field 1 to random(255)
  end repeat
end
```

***Juntando textos***

Você juntar (ou concatenar, no jargão de programação) duas strings. Basicamente você deve usar o operador &:

```
put "Juntar" & "Palavras" into field 1 -- retorna 'JuntarPalavras'
```

Se você usar dois & em vez de um, a concatenação é feita com um espaço:

```
put "Juntar" && "Palavras" into field 1 -- retorna 'Juntar Palavras'
```

Se você usar o operador & juntamente com o termo RETURN, a concatenação é feita em duas linhas:

```
put "Juntar" &RETURN& "Palavras" into field 1 -- retorna:
Juntar
palavras
```

Finalmente, se você usar o operador & juntamente com o termo QUOTE, a concatenação é feita acrescentando aspas:

```
put "Juntar" &QUOTE& "Palavras" &QUOTE into field 1 -- retorna:
Juntar" Palavras"
```

***Simulando hipertextos no Director***

Uma característica bastante comum em projetos hiperídia é a existência de hipertextos. A World Wide Web (WWW), por exemplo, é toda baseada na existência de hipertextos. No Director criar hipertextos não é tão simples. Porém, é possível que você simule hipertextos através de Lingo.

Os passos requeridos para adicionar hipertextos no seu filme são os seguintes:

1. Criar em fields o texto que vai abrigar a palavra que funcionará como hotword (vínculo ou *link*).
2. Colocar o field no stage.
3. Identificar a palavra, caracter, item ou linha que vai servir como hotword (vínculo ou *link*) para o hipertexto.
4. Adicionar o script no field.
5. Preparar o elemento que vai receber a ação do hipertexto. Por exemplo, se você vai mostrar uma imagem quando clicar na hotword, não se esqueça de preparar a imagem.

***Como identificar um caracter, palavra, linha ou item no corpo do texto***

A melhor maneira para identificar caracteres, palavras, linhas ou itens em um corpo de texto é usar uma das quatro funções que permitem a você identificar a string em razão da posição do mouse. Uma vez que o texto esteja identificado, você pode executar a ação com o mouse, quando o usuário clicar em cima de um pedaço de texto.

Cada uma das quatro funções retornam um número que identifica um valor numérico representando a posição (ordem em que se encontra) do caracter, palavra, linha ou item em relação ao início do field. Quando o mouse está apontado para um local que não há texto, o valor retornado é -1.

As quatro funções são:

|  |
|--|
| <b>Funções que identificam a posição do texto no field</b> |
|--|

| <i>Função</i> | <i>Descrição</i>  |
|---------------|---|
| the mouseChar | Conta os caracteres de um field e retorna um número que representa a ordem do caracter apontado pelo mouse. |
| the mouseWord | Conta as palavras de um field e retorna um número que representa a ordem da palavras apontada pelo mouse.   |
| the mouseItem | Conta os itens de um field e retorna um número que representa a ordem do item apontado pelo mouse.          |
| the mouseLine | Conta as linhas de um field e retorna um número que representa a ordem da linha apontada pelo mouse.        |

Para entender melhor o funcionamento destas funções, crie um field com um texto qualquer. Arraste o field para o stage e abra a Janela de Mensagens (Ctrl+M). Digite:

```
put the mouseWord
```

Deixe o mouse parado em cima de uma das palavras do field que você criou. Pressione Enter na Janela de Mensagens. Ela vai retornar o número correspondente à palavra por cima de onde você apontou o mouse.

Para retornar o valor da palavra em vez do número, digite o seguinte na Janela de Mensagens:

```
put word(the mouseWord)of field 1 -- no caso do field ser o member 1
```

A janela vai retornar a palavra por cima de onde você deixou o mouse.

Por exemplo, vamos criar o seguinte caso. No frame 1, vamos colocar um field com o seguinte texto: “As plantas são muito importantes para a nossa vida.”. No frame 2 vamos colocar a imagem de uma floresta.

No frame 1, entre o script do cast member correspondente ao field e digite:

```
on mouseUp
  if word(the mouseWord) of field 1 = "plantas" then
    go to frame 2
  end if
end mouseUp
```

Digite este script nos frames 1 e 2:

```
on exitFrame
  go to the frame
end exitFrame
```

Quando você clicar na palavra “plantas”, você vai navegar para o frame 2 e poderá ver a imagem da floresta.

Só faltou um pequeno detalhe: seria interessante que a palavra tivesse um destaque diferente para que o usuário soubesse que tem que interagir sobre ela. Isto é bastante simples: basta que a palavra tenha uma formatação diferente como a cor ou uma linha sobreposta (underline). Isto pode ser facilmente executado, selecionando a palavra e mudando a formatação.

Um outro fato que pode ajudar bastante é dar um efeito de clique à palavra, quando pressionada. Felizmente o Director tem um comando bastante conveniente para isto: o comando `hilite`:

```
hilite trechoDeTexto
```

No nosso caso acima, poderíamos alterar o script do field 1 para:

```
on mouseUp
  if word(the mouseWord) of field 1 = "plantas" then
    hilite word(the mouseWord) of field 1
    go to frame 2
  end if
end
```

Nesta programação o comando `hilite` vai estar funcionando sobre a palavra “plantas”.

Para terminar, seria interessante também que o cursor alterasse para o formato do “dedinho” quando o mouse apontasse para a palavra “plantas”. Esta é mais uma maneira de ajudar o usuário a enxergar que a palavra “plantas” é uma hotword. Aliás, é padrão, pelo menos na Web, fazer este formato de cursor aparecer em todo o tipo de vínculo. Então altere a programação para:

```
on mouseEnter
  if word(the mouseWord) of field 1 = "plantas" then
    set the cursor of sprite 1 to [4] -- o member 4 é imagem de "dedinho"
  end if
end mouseEnter

on mouseLeave
  set the cursor of sprite 1 to -1
end mouseLeave
```

### ***Controlando som através de Lingo***

O método mais comum para controlar arquivos de som internos por Lingo é pelo comando `puppetSound`:

```
puppetSound númeroCanal, castSom
```

O parâmetro `númeroCanal` especifica em qual canal o som será tocado. As opções são os canais de áudio 1 e 2, além de canais virtuais que você pode estabelecer. O Windows aceita até quatro canais virtuais de áudio. A plataforma Mac aceita até oito.

O parâmetro `castSom` especifica o nome ou número correspondente ao cast member que representa o som.

Por exemplo:

```
puppetSound 1, "som" -- toca o áudio "som" no canal 1
```

```
puppetSound 3, 2 -- toca o áudio representado pelo member 2 no canal 3
```

*Observação:* o comando `puppetSound` só é válido para arquivos de som internos. (Mais tarde veremos como controlar arquivos de som externo por Lingo). Quando você usa este comando, a janela Score perde o controle sobre os seus canais de áudio. Portanto, os arquivos de som destes canais deixarão de ser tocados pelo Score.

### ***Parando um arquivo de som interno***

Há dois métodos para parar um arquivo de som interno por Lingo. Primeiro, você pode parar um arquivo de som usando o valor 0 como parâmetro para `puppetSound`:

```
puppetSound 0
```

Este comando tira o controle dos canais de áudio do Lingo e retorna o controle para o Score.

Uma maneira mais conveniente de parar um arquivo de som é através do comando `sound stop`:

```
sound stop númeroCanal
```

Assim:

```
sound stop 2 - interrompe o áudio que está sendo tocado no canal 2.
```

Uma outra maneira de interromper um áudio é através do comando `sound close`. Ao contrário do comando `sound stop`, o comando `sound close` descarrega o som da memória, além de pará-lo. Sua sintaxe é semelhante à do `sound stop`.

```
sound close 1 -- para e descarrega um som da memória
```

### ***Pré-carregando um arquivo de som***

Você deve ter notado que, algumas vezes, um som demora para ser tocado. Isto acontece porque sons internos têm que ser carregados completamente na memória antes de serem tocados. Uma maneira conveniente de resolver este problema é carregar o som momentos antes de dar o comando para tocá-lo. Você pode fazer isto com o comando `preLoad`.

```
preLoad 10 -- carrega o som representado no cast member 10
```

```
preLoad 10, 15 -- carrega os sons representados entre os members 10 e 15
```

### ***A função soundBusy***

A função `soundBusy` verifica se um determinado som está ocupado (em outras palavras, tocando) ou não. É uma função bastante útil para que você possa enviar notificações para quando o som acabar de tocar. Ou para fazer algo enquanto o som estiver tocando.

```
soundBusy(númeroCanal)
```

Veja este exemplo:

```
on exitFrame
    if soundBusy(1) then -- se o som do canal 1 estiver tocando
        set the text of member "botao" to "Tocar"
    end if
end exitFrame
```

Neste script, pedimos para que o cabeçalho de um botão mudar o seu texto para "Tocar" assim que o som que é disparado por este botão parar de tocar. Assim, o usuário saberá que poderá clicá-lo novamente.

### ***Tocando arquivos externos***

O comando que toca arquivos externos é diferente:

```
sound playFile númeroCanal, <arquivo de som>
```

Veja o exemplo:

```
sound playFile 1, "c:\sons\sirene.wav"
```

Neste exemplo, o arquivo c:\sons\sirene.wav será tocado no canal de áudio 1.

### ***Controlando o volume do som***

Use a propriedade the soundLevel para controlar o volume do seu filme

```
set the soundLevel to n
```

Onde n é nível de volume que varia de 0 (mudo) a 7.

Assim:

```
set the soundLevel to 6 - altera o volume de som para o nível 6.
```

Esta apostila é de uso exclusivo para as disciplinas do curso de Design Digital da Universidade Anhembi Morumbi. É proibida a reprodução deste material sem a autorização do autor.