

# Criando Aplicativos Multimídia com **Director 7**

V. 1.0

Uma coletânea de informações indispensáveis e dicas quentes sobre o software de autoria em multimídia mais utilizado no mundo!

**Engloba parcialmente as versões 5, 6 e 6.5**

**Fabrizio Guedes Bissoli**  
<http://www.phasis.net>

**Júlio Francisco Planella Gonzalez**  
[jfpg@ez-poa.com.br](mailto:jfpg@ez-poa.com.br)

Nível



Básico Intermed.

*Director is a trademark of Macromedia Inc.*

# PREFÁCIO :

Esta obra foi desenvolvida com o objetivo de oferecer apoio àqueles que estão iniciando suas atividades no setor de desenvolvimento de multimídia. A ferramenta de autoria utilizada é o software da Macromedia, Director 7. Como a produção de multimídia não é restrita somente a um único software, eventualmente outros nomes de softwares serão citados a nível complementar (editores de imagens, sons, vídeo, etc).

A versão do Director abordada é a 7.0 para Windows, logo alguma diferença entre as imagens da interface capturadas e a interface da versão para Macintosh pode ser observada (apesar de pouco significativa), mas o conteúdo referente à programação Lingo (linguagem utilizada pelo Director) é quase idêntico em ambas plataformas.

O material apresentado foi desenvolvido à partir da compilação de perguntas frequentes (FAQs) em fóruns e listas de discussão referentes ao tema proposto, uma extensa bibliografia e, principalmente, da experiência dos autores. Tentou-se apresentar certas questões que não são abordadas diretamente em livros e/ou manuais existentes até então. O objetivo deste trabalho não foi apresentar um manual de Director e/ou Lingo, mas oferecer base para os iniciantes para um início sem traumas e dicas importantes mesmo para aqueles de nível intermediário.

O livro se inicia com uma introdução genérica à multimídia, comentando não diretamente sobre o referido software, mas sobre a estrutura geral necessária à produção de multimídia. Hardware e software necessários, dicas sobre vídeo e áudio digitais e outros assuntos úteis ao iniciante são abordados no chamado “capítulo zero”. Em seguida, no capítulo 1 encontra-se uma descrição da estrutura de funcionamento do Director, sua interface e como ele interpreta eventos com um dinamismo particular. São abordados casos simples de animação que não necessitam praticamente de códigos de programação Lingo no decorrer do capítulo 2. Multimídia acessada via browsers em Intranets ou Internet é sucintamente abordada no capítulo 3, sendo que os assuntos tratados no mesmo são Shockwave e Java. No capítulo 4 inicia-se a programação com Lingo, a linguagem própria do Director. Esse capítulo é bastante extenso devido a alta complexidade do assunto, não que Lingo seja uma linguagem difícil, mas sim pelo seu potencial. Obviamente o assunto "programação Lingo" não é completamente abordado nesse material, todavia procurou-se abordar alguns dos comandos mais usados e, posteriormente citar-se exemplos práticos nos anexos. No capítulo 5 são apresentados algumas estruturas de linguagem "pré-fabricadas" presentes no Director 7. O restante do material visa trazer a praticidade abordada através de exemplos bastante úteis e usuais.

Multimídia na Internet é a grande tendência, e como o Director oferece recursos interessantes para a Web, esses tópicos são também comentados: trata-se da tecnologia Shockwave e Java, incorporadas já há algum tempo nos pacotes do Director Multimedia Studio. Em seguida vem o Lingo, a poderosa linguagem de programação multimídia da Macromedia, que alia relativa simplicidade a impressionantes resultados. O Lingo é abordado de forma superficial, pois é impossível tratar de um assunto tão extenso em tão pouco espaço. Alguns comandos mais utilizados são apresentados e algumas aplicações práticas são mostradas posteriormente.

Não é pretensão deste material substituir bons livros existentes no mercado, mas com certeza o conteúdo aqui apresentado é indispensável a qualquer iniciante e mesmo aqueles de nível intermediário poderão encontrar novos conhecimentos entre as diversas dicas práticas aqui apresentadas.

Esperamos atender às expectativas do leitor.

\*\*\*

### **Direitos de uso do Arquivo :**

#### **Para a Versão Trial :**

É liberado o uso livremente do conteúdo deste arquivo, pois trata-se de uma versão reduzida do livro.

#### **Para a Versão Completa do Livro :**

É expressamente proibido que o conteúdo aqui apresentado seja cedido a terceiros sem a devida permissão dos autores. É permitida a impressão deste material de modo a oferecer maior comodidade na leitura para o leitor licenciado, mas o mesmo material impresso obedece às mesmas normas de direitos autorais de seu formato digital.

# Índice :

(para o Livro Completo)

Introdução	7
Capítulo zero - O que é necessário para se criar um cd-rom	9
Capítulo 1 - A estrutura	20
Capítulo 2 - Animação básica	41
Capítulo 3 - Shockwave e Java	49
Capítulo 4 - Lingo	59
Capítulo 5 - Behaviors pré-definidos	103
Anexo 1 - Preparando um cd para distribuição	123
Anexo 2 - Alterações no Lingo do Director 7	127
Anexo 3 - Listas	129
Anexo 4 - Programação orientada a objetos & parent scripts	137
Anexo 5 - Definições básicas	143
Anexo 6 - Strings - conceitos fundamentais	146
Complemento - Lingo, dicas úteis	149

# Introdução :

O software de autoria Macromedia Director 7 disponibiliza uma enorme gama de controles necessários para a criação de aplicativos que podem variar de simples apresentações a complexos games, programas interativos (inclusive acessados via browser) e aplicativos Multiusuários via Internet.

Apesar de ser o mais poderoso software de autoria em multimídia existente, eventualmente pode-se fazer necessário o uso de recursos adicionais do mesmo. Isso pode ser resolvido geralmente adquirindo-se “Xtras” (complementos) de outros desenvolvedores e que divulgam seus produtos pela Internet. Alguns Xtras são gratuitos (freeware), outros solicitam uma doação a instituições (donationware) e outros, comercializados. É importante observar que a necessidade de adquirir-se um Xtra externo dependerá do objetivo do software proposto, os Xtras expandem as possibilidades do Director de forma ilimitada. A Macromedia oferece todas as informações necessárias sobre a arquitetura para o desenvolvimento de Xtras (MOA - Arquitetura aberta da Macromedia) de forma que os interessados possam desenvolver seus próprios Xtras em C++.

Neste material serão mostrados os recursos mais usados do Director 7 de forma a possibilitar o desenvolvimento de softwares de considerável complexidade. Serão mostradas técnicas de animações e controle de interatividade através de exemplos práticos. Ao final deste curso, você estará apto a desenvolver seu próprio software utilizando não somente os comandos aqui abordados, pois o objetivo principal do mesmo é capacitar aos iniciantes a possibilidade de um auto-desenvolvimento a nível avançado, seja consultando o Help do Director, através de manuais e/ou livros e listas de discussão na Internet.

# Principais Codecs :

## **Radius Cinepak Codec :**

Apesar de a compressão com este codec ser mais lenta, a descompressão ocorre bem mais rapidamente, o que torna esse codec um dos mais indicados como compressor de Vídeo for Windows, trabalhando muito bem na criação de títulos em CD-ROM. Trabalha melhor em 24 bits.

## **Intel Indeo 3.2 :**

Também funciona melhor em 24 bits e é o mais versátil codec, sendo também muito recomendado para obras em CD-ROM. Apesar de muito semelhante ao Cinepak quanto à taxa de compressão e qualidade, parece que exige um pouco mais da capacidade do processador durante a descompressão.

## **Microsoft Video 1 :**

Suporta 8, 16 ou 32 bits de cor. A única vantagem deste codec é a rapidez durante a compressão.

## **Microsoft RLE :**

Foi criado originalmente para o uso em animações, não sendo recomendável seu uso com imagens capturadas a partir de câmeras de vídeo. Uma animação 2D com cor sólida pode trabalhar bem com este codec, lembrando que este suporta apenas 8 bits de cor.

## **Restrições de uso do padrão MPeg :**

Necessidade de software ou hardware específicos: O padrão MPeg é uma forma compactada de vídeo digital, por isso consegue exibir vídeos de excelente qualidade (e áudio hi-fi) mesmo com uma baixa taxa de transferência do CD-ROM, porém, torna-se necessário uma descompactação em "real-time" para que os quadros sejam exibidos convenientemente. Algumas placas de vídeo já possuem esse descompactador incorporado, existem também placas descompressoras MPeg vendidas separadamente; existem ainda softwares que fazem esta descompactação utilizando a capacidade de processamento do processador principal do micro (e não da placa de vídeo), mas em qualquer desses casos, um gasto adicional se faz necessário, seja em hardware ou em software.

**Obs:** O Padrão MPeg 2 é o mesmo utilizado em DVDs.

## A Estrutura :

A seguir serão abordados os conceitos básicos fundamentais, necessários para se entender o mecanismo do Director e, após essa teoria, será abordada a programação em Lingo simultaneamente a exemplos práticos. Esses fundamentos são realmente **indispensáveis**, e não há como seguir adiante na programação sem que se entenda perfeitamente sua estrutura. Lembre-se de que o motivo do Director ser a ferramenta adotada por 75% dos desenvolvedores de multimídia no mundo, não deve-se somente à sua incontestável superioridade de recursos, mas também à interface que trata de uma forma inovadora a programação multimídia.

O Director usa metáforas do meio teatral como “Stage” (Palco) (onde são mostradas as apresentações), “Cast” (Elenco), onde é armazenado o elenco (castMembers ou membros do elenco) e “Score”, (onde estão contidas as instruções que organizam o elenco), Behavior (comportamento), geralmente contém funções de Lingo, que ao ser aplicado a uma sprite fazem com que ela tenha um comportamento próprio. Tudo isso e muito mais será apresentado com detalhes mais adiante.

O *score* está dividido em quadros para facilitar o controle de tempo e *layer* de cada objeto usado no projeto. Se observarmos os quadros na horizontal, estaremos observando os *frames*. Um frame é como um quadro de um filme, exibido durante determinado período de tempo. O *default* desse tempo é 1/15 do segundo, ou seja, em 1 segundo são exibidos 15 quadros. Esse valor é aproximadamente a percepção retiniana e uma animação mostrada nessa velocidade geralmente é suficiente para bons resultados. No Director 7 essa velocidade pode ser aumentada até 1000 fps (frames por segundo), porém o desempenho real dependerá de uma máquina (usuário final) que tenha capacidade de mostrar uma animação nessa velocidade. Atenção a isso!

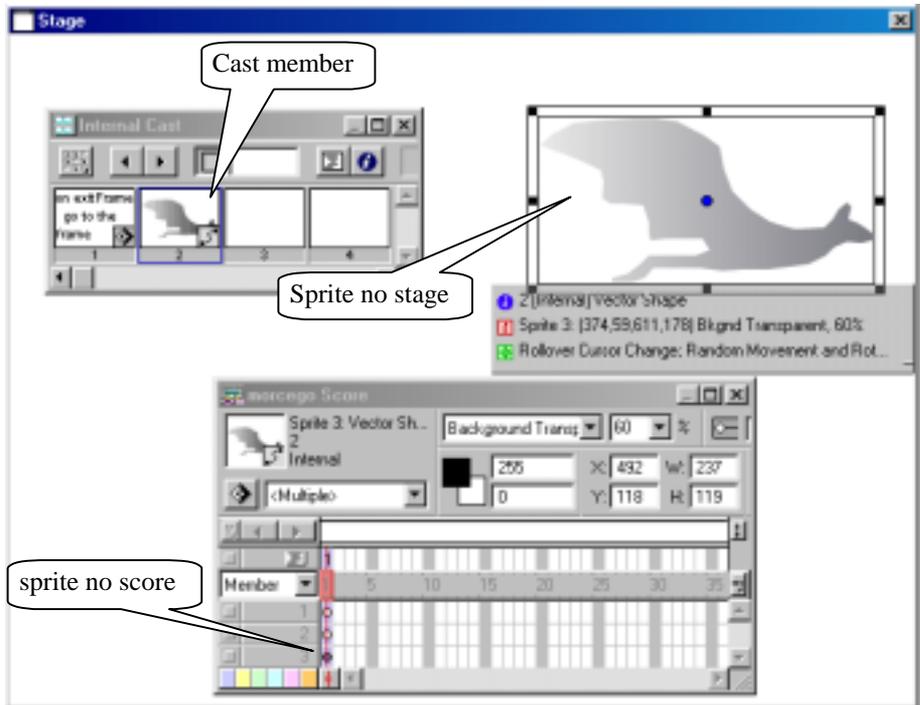
Observando-se esses mesmos quadros na vertical, verifica-se o *layer*, ou como é chamado no Director, o *channel* (canal) de cada objeto. Se um *cast member* (membro do elenco) for solto sobre o canal 1 e um outro estiver ocupando o canal 2, o objeto no canal de nível superior aparecerá sobre o antecessor. Esses objetos são chamados de Sprites no “linguajar macromediano”. Sprites são membros do elenco em cena, ou seja, o que o usuário final observará.

# 1 - Como funciona o Director :

Como o próprio nome sugere, você é o “diretor” de um filme. Nesse filme estão os membros do elenco (*Cast members*) e você comanda o espetáculo dizendo para cada membro o que ele deve fazer. Só para entender a diferença entre ele e seu concorrente mais próximo (o ToolBook), basta lembrar que neste você associa eventos a objetos e cada objeto está associado a outros. No Director é diferente, o filme vai “correndo”, o *script* (código contendo comandos da linguagem *Lingo*) referente a cada objeto (Esse “objeto” no Director chama-se *sprite* e refere-se ao *Cast member* em ação, conforme citado anteriormente) comanda a interatividade e tudo mais... de forma que o filme quase nunca para na realidade. Essa visão dinâmica ajuda em muito a estrutura de um projeto que envolva sobretudo animações.

Com a figura a seguir, pode-se entender melhor os diversos componentes de um filme do Director. O programa executável (compilado) que o Director gera é chamado de *projector*. Pode-se criar projetores de 16 bits (funciona em Windows 3.x e Windows 9x/NT) e 32 bits (apenas Windows 9x/NT), mas não é recomendável criar-se projetores 16 bits para serem executados em sistemas 32 bits. Com a mesma facilidade com que se cria um projetor de 16 ou 32 bits, pode-se criar um projetor para Machintosh, bastando para tal compilar o programa fonte em um Mac com o respectivo Director.

## Cast members e sprites :



Cast members são os componentes do filme, mesmo que eles não estejam em cena. Quando importamos algum objeto para dentro de um filme do Director, ele torna-se um Cast member. Também quando definimos um script para um objeto, ele torna-se um cast member. Os cast members são organizados e exibidos em casts, conforme a figura abaixo.

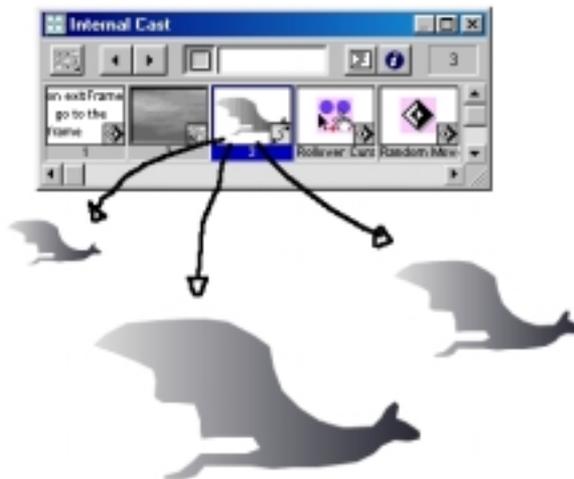
Exemplos de mídias: Imagens (BMP, JPG, GIF), Animações (FLI, FLC e Film Loop), Vídeos (MOV, AVI e MPEG) e áudio (WAV, AIF, SWA, AU e MP3).



Sprites são objetos em cena. Quando se arrasta um cast member para dentro do stage, ele torna-se um sprite. É importante observar que um cast member pode existir sem que exista na forma de sprite, mas a recíproca não é verdadeira.

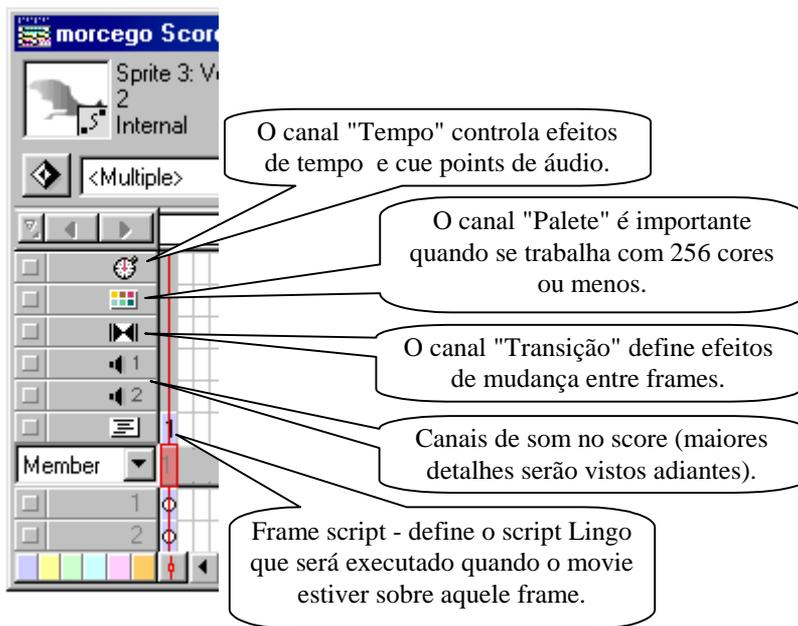
### **Importante :**

O mesmo cast member pode originar diversos sprites, sendo cada um com suas propriedades de tamanho, posição no stage, opacidade, tinta, etc...



## Os canais do score :

Como representado na figura abaixo, existem alguns canais para controlar sprites no score. O primeiro é relativo ao tempo, o seguinte às palletes de cor, o outro às transições de frames. Tem-se ainda o controle individual dos 2 canais de áudio e o script de cada frame, executado cada vez que a execução do filme chega neste.



Com a tabela abaixo, pode-se referenciar vários tipos de mídias através da visualização dos indicativos das mesmas.



Cast Member	Descrição da Mídia
1	Bitmap,
2	Pallete de cor,
3	Efeito da Intel,
4	Fonte,
5	Cursor,
6	Gif animada,
7	Vector shape,
8	Flash movie,
9	Botão,
10	Field,
11	Text,
12	Movie script,
13	Parent script,
14	Behavior,
15	Vídeo digital AVI,
16	Áudio interno,
17	Áudio "linkado",
18	SWA

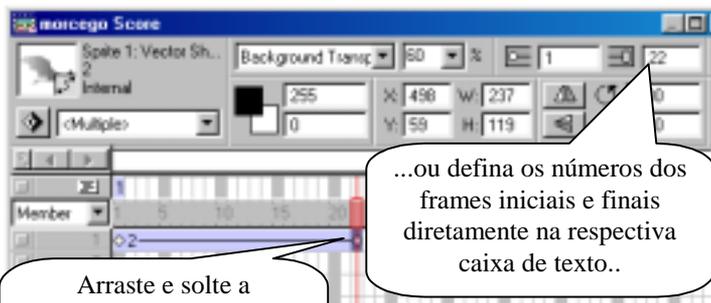
### 3 - Trabalhando com Sprites :

Neste capítulo inicia-se a parte da organização do material. Na verdade uma animação é tão somente a definição de instruções para cada elemento componente do filme e um complexo aplicativo de multimídia é apenas a integração de várias mídias.

A princípio, um sprite precisa ser devidamente organizado em um lugar do score. Deverá também estar associado a ele (eventualmente) um código de instruções; esse código, que na verdade é uma linguagem própria do Director chama-se **Lingo**. Veremos que algumas animações podem ser feitas sem que seja necessário escrever-se uma única linha de código.

#### Criando sprites :

Quando um cast member é arrastado e solto sobre um canal do score, é criado um sprite com duração de 28 frames (isso é o default, mas pode ser modificado para o número desejado). ou então, caso o mesmo já exista, pode-se arrastar a barra em uma das extremidades, colocando-a do tamanho em Frames que se desejar.



Arraste e solte a extremidade posterior do sprite para definir até em que frame ele estará presente...

...ou defina os números dos frames iniciais e finais diretamente na respectiva caixa de texto..

## Film loops :

Muito usado em animações, geralmente em conjunto com o Tweening. Imagine que se queira fazer um boneco caminhando de um ponto a outro da tela, ou mesmo a rotação da terra em conjunto com uma translação. O Film loop é o que o próprio nome indica. Após definida uma seqüência de imagens, pode-se criar um loop de filme e trabalhar com ele como se fosse uma única imagem, mas ao criar-se um movimento com este, ele executará além da animação (que pode ser o Tweening) o loop determinado. Um exemplo disso pode ser o movimento de um pássaro: suas asas batem enquanto ele se move sobre o stage, o "bater de asas" pode ser um film loop e ele será tratado pelo Director como um bitmap comum, com a diferença que ao invés de mostrar uma imagem estática , mostrará uma animação.



### **Criando um film loop :**

Selecione a seqüência de frames desejada. Escolha Insert ⇒ Film Loop. Defina um nome para ele e clique ok. Após criado o Film Loop, o mesmo aparecerá como um cast member independente, apesar de não o ser. Pode-se apagar a seqüência de frames que o gerou no Score ou no Stage, mas não pode-se apagar os cast members que lhe deram origem. Uma vez criado um frame loop, pode-se fazer qualquer animação como se ele fosse uma imagem comum.

---

## Lingo

- **Onde se localiza o Lingo**
- **Caminhos para se acessar os scripts**
- **Alguns eventos em Lingo**
- **A janela "Watcher"**
- **Eventos**
- **Sprite Script**
- **CastMember Script**
- **Localização e Propriedades da Janela de Scripts**
- **Eventos do Mouse**
- **Funções relacionais ao Ponteiro do Mouse**
- **Comando IF**
- **Comando Case**
- **Uso e Controle de um MIAW**
- **Controlando áudio e vídeo digital com Lingo**

## Onde se localiza o Lingo :

Esse capítulo é de extrema importância e seu completo entendimento é fundamental para um bom desenvolvedor.

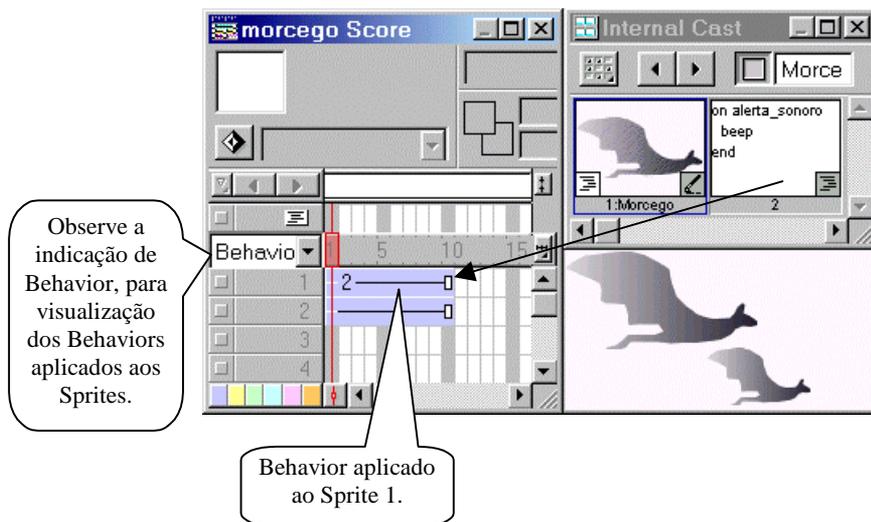
Os scripts de Lingo podem variar-se quanto a sua localização e tipo das seguintes maneiras:

<b>Sprite Script</b>	Como scripts de sprites, sendo este o método mais usado para sprites, podemos aplicar múltiplos Behaviors (comportamentos) a uma sprite.
<b>Cast members Scripts</b>	Como script de um cast member, sendo que este castMember ao virar uma sprite terá um mesmo comportamento durante toda sua vida útil no Score, e não só nos frames determinados (sprite Script).
<b>Frame Script</b>	Definido na barra de script do score, é ele que determina o comportamento daquele Frame, ou define o fluxo da apresentação.
<b>Movie Script</b>	Este é o lugar aconselhável para criações de funções, definição de variáveis, comportamento inicial, frame a frame, e final do programa, ou seja, a programação em Lingo.

# Diferenças entre Sprite Script e CastMember Script :

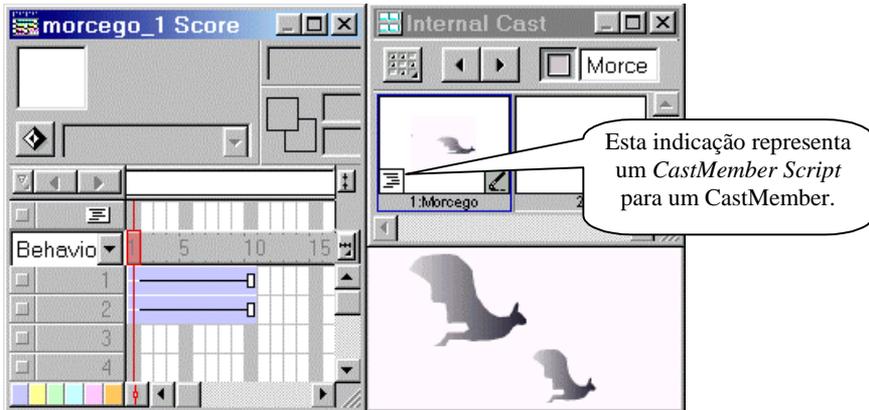
## Sprite Script :

Quando aplicamos um script a uma sprite como *Sprite Script*, este terá efeito quando algo ocorrer na sprite e apenas na sprite que foi aplicado. Como no exemplo foi aplicado o Behavior “click\_sprite” na sprite 1.



## CastMember Script :

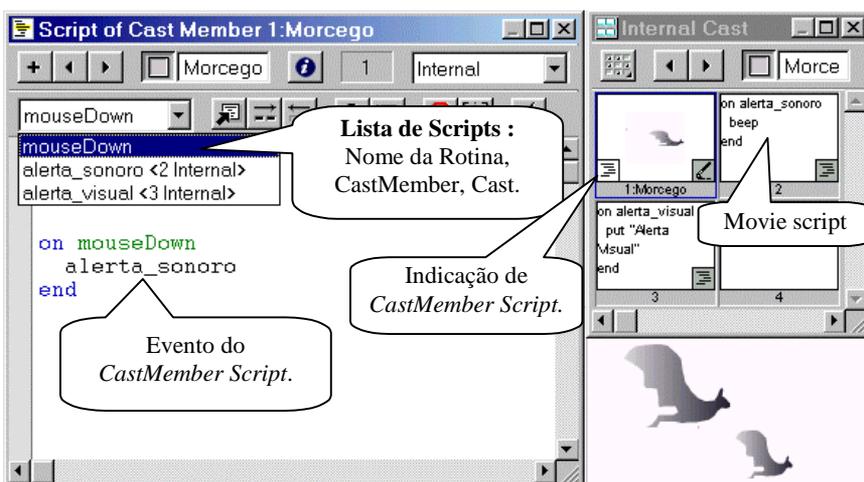
Quando definimos um *CastMember Script* a um CastMember, o script estará ativo quando este castMember for arrastado para o Stage ou para o Score, virando uma sprite com um *CastMember Script* vinculado.

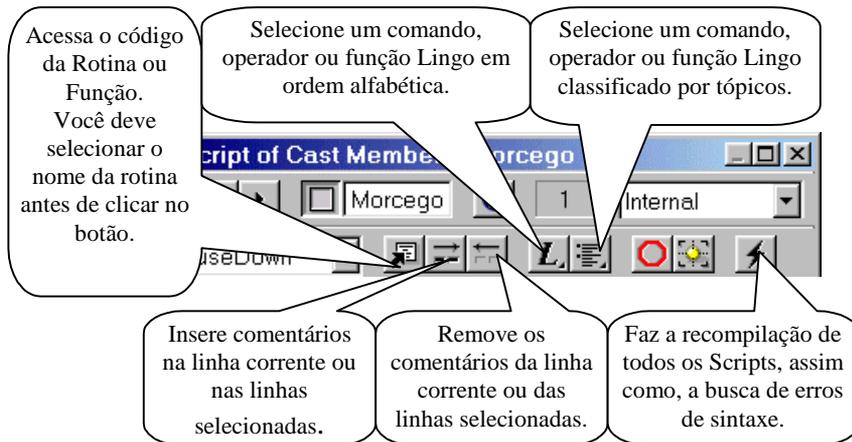
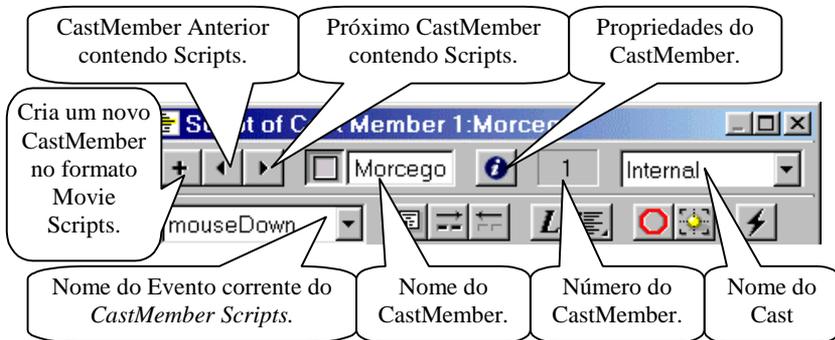


# Localização e Propriedades da Janela de Scripts :

O título da janela “Script of Cast Member 1:Morcego”, está nos informando que o script é do tipo *CastMember Script* e este esta vinculado ao CastMember 1 ⇒ “Morcego” do Cast Internal.

A Lista de Scripts, apresenta o Evento corrente do CastMember Script, assim como uma relação de Rotinas e Funções de outros CastMembers contendo Scripts.





# Eventos e Funções relacionais ao Ponteiro do Mouse :

## Eventos do Mouse :

...

...

## Funções relacionais ao Ponteiro do Mouse :

...

...

## Cursor do Mouse :

...

...

## Comando IF :

IF, talvez seja um dos comandos mais usados em um programa, seja ela em que linguagem for, é ele que determina se o fluxo do programa vai seguir por um lado ou pelo outro.

### Convenções preliminares :

Não importa o tão complexa seja a expressão lógica, ou quantas variáveis ela tenha, vamos apenas obter 2 valores : **TRUE** ou **FALSE**, podemos traduzir como Verdadeiro ou Falso, mas o mais certo é dizer : 1 ou 0.

Nas expressões lógicas podemos usar o > (maior), < (menor), = (igual), <> (diferente), NOT (negação), **AND**, **OR**, etc...

O NOT significa que ele vai inverter o estado lógico da expressão, por ex. : NOT **TRUE**, significa **FALSE**.

### Sintaxe do IF :

```
if expressão_lógica then
  Comando(s)
else
  Comando(s)
end if
```

ou

```
if expressao_logica then comando
```

...

...

## Comando Case :

O comando **case** do Director, é muito semelhante ao das outras linguagens, ele testa a expressão e a compara com as expressões seguidas.

### Sintaxe :

```
case expressão of  
  expressao1 : comando(s)  
  expressao2 : comando(s)  
  expressao3, expressao4 : comando(s)  
  expressaoN : comando(s)  
  otherwise: comando(s)  
end case
```

Seu funcionamento é bem simples, caso a expressão seja igual a uma das expressões a baixo do case, ele executa o devido comando(s), caso as expressões não coincidam com a expressão, ele executa o comando(s) do otherwise.

### Alguns exemplos :

...  
...

## Uso e Controle de um MIAW :

Movie In a Window, mais conhecido como MIAW, geralmente é usado para rodar outro movie, enquanto o movie principal continua sendo executado, por exemplo, o movie principal poderia ser um programa de tratamento de imagens e o MIAW poderia ser a barra de ferramentas para o programa.

Temos completo controle sobre um MIAW, desde sua visibilidade, posição na tela, tamanho da janela, tipo de janela, título da janela, troca de informações entre o movie principal e o MIAW, entre outros comandos via Lingo.

### **Exemplo de abertura de um MIAW :**

No exemplo a baixo, abrimos uma janela chamada ferramentas, a partir do arquivo “janela\_ferramentas.dir” localizado no raiz do drive D:, setamos seu tamanho e a posição através do comando “the rect of window”, esta janela será apresentada sem barra de título.

```
on abrir_janela_ferramentas
  set the fileName of window ferramentas to the pathName &
  “janela_ferramentas.dir”
  set the rect of window ferramentas to rect(0, 0, 250, 200)
  set the titleVisible of window ferramentas to FALSE
  open window ferramentas
end
...
...
```

### **Descrição dos comandos Lingo para abertura e controle de um MIAW :**

```
...
...
```

# Controlando áudio e vídeo digital com Lingo :

Um dos elementos mais importantes da multimídia é o áudio. Costuma-se dizer que ele chega a representar 50% da importância das mídias em um CD-ROM. Realmente, quando o objetivo é criar multimídia, fica difícil imaginar esta sem bons recursos de som. Seja em CD-ROM, em Shockwave ou mesmo Java, o uso adequado de áudio pode melhorar em muito uma apresentação interativa.

Falar em áudio e vídeo digital é tema para muitas e muitas páginas, por isso esse assunto será tratado de forma sucinta, mas abrindo porta para que o leitor busque maiores informações à medida que precisar.

## **Sobre o áudio :**

Inicialmente, devemos levar em conta o que desejamos fazer. Se é um CD-ROM, em qual plataforma ele será executado (PC ou Mac)? que qualidade de áudio deverá ser adotada? haverá necessidade de sincronismo com animações? existirá algum arquivo baixado via Internet?

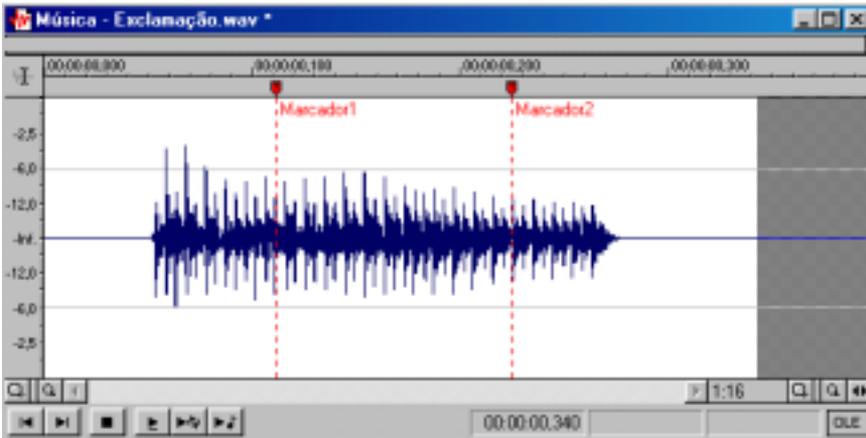
Essas questões já devem estar definidas antes que as mídias sejam digitalizadas. Devemos escolher o melhor formato de áudio que atenda às necessidades do projeto, e não apenas o melhor formato de áudio! Por exemplo, para que você usaria uma wav de 44Khz, 16 bits estéreo para uma simples narrativa se a voz humana está numa faixa de frequências bem menor e é monofônica? ou para que usar uma wav em uma apresentação Shockwave para tocar uma trilha musical se podemos usar o swa?

Vamos definir os principais tipos de áudio usados com o Director 7, isso servirá para esclarecer algumas dúvidas freqüentemente presentes em fóruns de discussão e listas pela Internet:

<b>WAV</b>	Padrão do Windows, normalmente é usada em softwares que funcionarão sob Windows. A qualidade escolhida dependerá da utilização, lembre-se que o tamanho é diretamente proporcional à qualidade, um arquivo em 22khz, 8 bits, mono, terá 1/4 do tamanho de um arquivo com a mesma duração, a 22khz, 16 bits e estéreo. Suporta Cue Points que tenham sido gerados com o software Sound Forge 4 ou superior.
<b>AIF</b>	Padrão do Mac. Funciona bem mesmo no Windows. Muito semelhante em tamanho e qualidade ao WAV. Suporta Cue Points sendo que é necessário o software SoundEdit 16 (existente somente para Mac).
<b>AU</b>	Padrão da Sun. Essa extensão é usada principalmente para apresentações convertidas para JAVA. Não suporta Cue Points.
<b>SWA</b>	Criado pela Macromedia para ser usado principalmente em aplicações que ocupem pouco espaço (por exemplo, Shockwave). Trata-se de um formato compactado muito semelhante ao MP3 (mesma qualidade e compactação). Sua semelhança chega ao ponto de que, se for renomeado para .mp3, funciona perfeitamente em um player do mesmo. Não aceita Cue Points.
<b>MP3</b>	O Director 7 aceita importar arquivos MP3, a necessidade de sua utilização deverá ser avaliada com cautela, sendo mais recomendável utilizar-se do padrão compactado .swa. Também não aceita Cue Points.

## Obs.:

Cue Points são "marcadores" nas trilhas de áudio que permitem informar ao movie em que ponto exato da trilha se está, permitindo um perfeito sincronismo entre áudio e animações.



Exemplo de locação de Cue Point (marcadores 1 e 2) usando o software Sound Forge 4.0

O gerenciamento de Cue Points é de extrema importância, mais adiante esse assunto será novamente abordado quando se falar do comando "cuePassed".

Um arquivo de som é tratado como uma mídia qualquer quanto à importação, porém existe uma diferença quanto à sua localização no score. Quando importar um áudio de tamanho significativo, é conveniente trazê-lo para um cast externo, importando-o internamente para este. O motivo é simples: se fosse importado para o cast interno, o tamanho do projetor seria demasiadamente grande (é conveniente limitar seu tamanho para poucos megabytes), se ele fosse importado com "link to external file", o arquivo de áudio seria visível e isso é deselegante para um programador em Lingo. Um cast externo é o lugar mais conveniente para abrigar arquivos de áudio! Infelizmente não se pode dizer o mesmo para vídeos digitais, pois estes sempre estarão "linkados" externamente, sendo recomendável por razões estéticas que seja criado uma pasta para abrigar esses vídeos.

## Controlando Áudio em um movie :

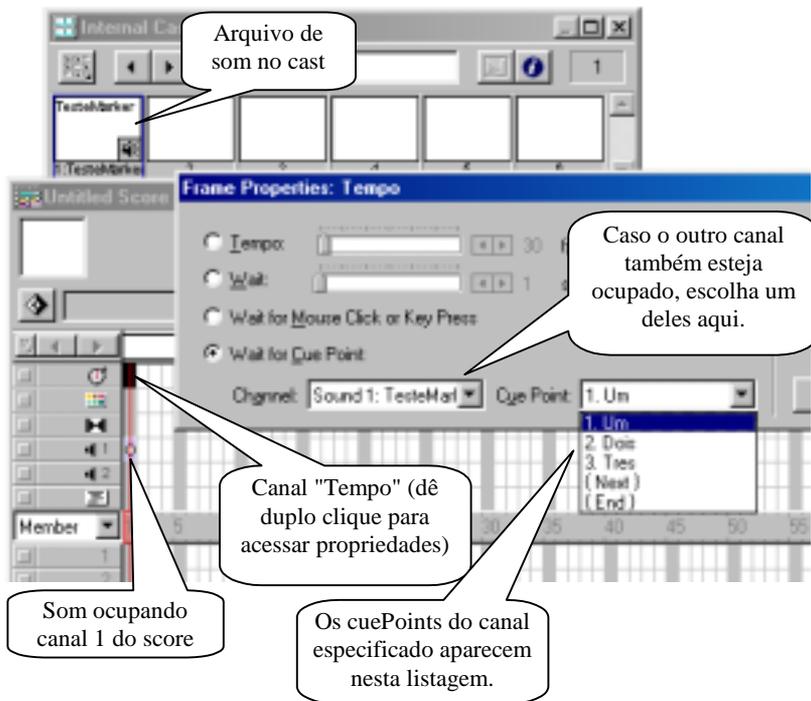
Os arquivos de áudio normalmente estarão dentro de casts. Isso é recomendável e, salvo necessidade específica, recomenda-se que esteja em um cast externo.

Os formatos aif, wav, au e swa são totalmente compatíveis com o Director 7. Até a versão 6.5 os arquivos .swa eram tratados de forma diferente, porém isso já não acontece no D7.

Considerando-se que o arquivo de áudio já esteja em algum cast, observa-se 3 maneiras de se ativar esse som:

1 - Traz-se o cast member (clitando e arrastando) até um dos dois canais de áudio no score. Ele aparecerá como sprite, porém no canal de áudio exclusivamente! Quando o filme for executado, enquanto o *header* (indicador vermelho) estiver em frame adjacente ao áudio, ele será ativado (da mesma forma que um bitmap seria mostrado), a menos que algum comando Lingo o desative propositalmente.

2 - Usando o comando *sound playFile*: é possível controlar-se o áudio diretamente via Lingo. Existem várias vantagens do uso do Lingo sobre o processo citado anteriormente. Imagine que dez botões situados em um mesmo frame emitem sons, cada um diferente do outro. Você simplesmente colocará o código Lingo associado a cada arquivo de áudio em seu respectivo botão. Outra: como fazer mais de dois sons tocarem simultaneamente? Basta acionar cada som em um canal diferente. Os canais existem além dos 2 canais mostrados no score e, pode-se ter, por exemplo, um pássaro cantando no canal 1 enquanto um riacho utiliza o canal 2 e uma vaca muge no canal 3. Cuidado para não confundir canal de som no Director com canais de áudio comuns. Um som estéreo possui dois canais, mas não significa que utilizará dois canais do Director. Um som estereofônico ocupa um único canal do Director, mesmo que soe estéreo nas caixas acústicas.



Observe que, apesar da simplicidade mostrada acima, os recursos são limitados usando-se simplesmente o canal Tempo do score. Para um melhor gerenciamento dos cuePoints é necessário o Lingo. A seguir são mostrados alguns exemplos do uso de cuePoint controlado via Lingo:

O handler abaixo coloca em uma caixa de mensagem as informações referentes aos cuePoints presentes no canal 1 do áudio.

```

on cuePassed channel, number, name
  if (channel = #Sound1) then
    put "CuePoint" && number && "named" && name &&
      "occurred in sound 1"
  end if
end

```

## Controlando vídeo digital :

Após escolher-se um arquivo de vídeo digital para a importação (lembre-se sempre de definir a importação para "link to external file"), uma opção surge com alternativa para AVI ou MOV, lembre-se que um MOV no Windows sempre precisará de QuickTime para visualização. Se você estiver trabalhando em um projeto unicamente para Windows e não vai utilizar nenhum recurso de QTVR (Quick Time Virtual Reality), recomenda-se sempre o uso do tipo AVI com codec Cinepak ou Intel Indeo 3.2.

### **Observações :**

QTVR é uma tecnologia desenvolvida pela Apple para emulação de realidade virtual em vídeo. Atualmente é necessário um software desenvolvido pela Apple chamado "Quick Time VR" e compatível com Mac somente.

Vídeos digitais no padrão MPeg são compatíveis com o Director mas necessitam de Xtras fornecidos por outros fabricantes (consulte a lista de Xtras da Macromedia em seu site).

No QuickTime versão 2.1.2 ou mais recente, é possível ter-se mais de uma trilha de áudio em um mesmo vídeo digital e controlá-la via Lingo. Isso pode ser muito útil quando se trabalha em vários idiomas, por exemplo. Pode-se ativar determinada trilha de áudio, checar qual trilha está ativa, contar o número de trilhas existentes em determinado vídeo, etc. Para maiores detalhes, consulte o help do Director usando as palavras chave *trackCount*, *trackType*, *trackText*, *trackEnabled*, *setTrackEnabled*, *trackStartTime*, *trackStopTime*). Arquivos AVI não suportam múltiplas trilhas até a presente versão!

Os controles de vídeo "play", "pause", "ff", "rew", etc, podem ser definidos em botões usando scripts Lingo como mostrado abaixo:

<b>play</b>	<code>sprite(X).movieRate = 1</code>
<b>pause</b>	<code>sprite(X).movieRate = 0</code>
<b>slow</b>	<code>sprite(X).movieRate = 0.5</code>
<b>ff</b>	<code>sprite(X).movieRate = 2</code>
<b>roda o filme ao reverso</b>	<code>sprite(X).movieRate = -1</code>
<b>volta o filme ao início</b>	<code>sprite(X).movieTime = 0</code>

### Obs.:

Use esses comandos associados a botões e dentro de um handler "`on mouseUp`" para definir controles simples de vídeo digital.

Quando um vídeo digital termina, é comum encontrar-se "resíduos" sobre a tela. Para resolver esse problema use o seguinte script para um perfeito "refresh" sobre a tela:

```
on exitFrame
  sprite(X).movieRate=0
  sprite(X).visible=FALSE
  updateStage
  sprite(X).visible=TRUE
end
```

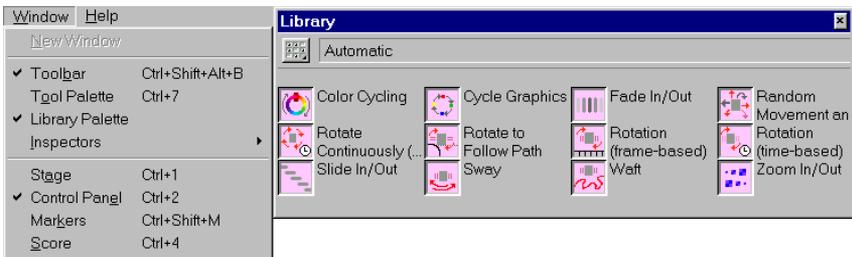
Onde X é o número do sprite referente ao vídeo digital.

---

## Behaviors pré-definidos

- **Behaviors de Formulário (Forms.cst)**
- **Behaviors Multiusuários (Multiuser.cst)**
- **Behaviors Streaming (Streaming.cst)**
- **Multiuser Server**
- **Client-Director Multiuser Server**
- **Peer-to-peer**
- **Client-to-text-based-server**
- **Como criar chats e outros aplicativos multiusuários**
- **Grupo Navigation**

A Macromedia disponibiliza alguns behaviors "pré-fabricados" para facilitar a programação com rotinas que se repetem com frequência. Esses behaviors são muito úteis e dificilmente um filme não usará pelo menos um desses scripts. Devido a facilidade de uso é possível até mesmo que um programador inexperiente use de recursos mais avançados do Lingo. Estes behaviors estão dispostos em Casts externos convenientemente, mas não é necessário abri-los para se ter acesso a seus scripts, O conceito de Behavior "pré-fabricado" começou com a versão 6.0 do Director, trazendo inúmeras rotinas prontas para uso, bastando apenas clicar e arrastar para uma sprite ou para o Stage(frameScript), gerando assim um *Comportamento* a sprite ou ao frameScript. Podemos acessar através do menu Window ⇨ Library Palette, uma janela contendo inúmeros Behavior para uso.



## Obs.:

Os behaviors apresentados aqui compõe o software Director 7, não estando presente em versões anteriores.

...  
...

## Preparando um CD para distribuição :

Considerando-se o término de um projeto, ou seja, que o projetor e todos os seus componentes já tenham sido testados e “re-testados” inclusive em CDRs, passa-se à etapa seguinte. Esta etapa é a implementação final para a distribuição aos usuários. A seguir será apresentado o imprescindível:

### 1 - Editando o ícone de seu projetor :

Sempre que um projetor é gerado, um ícone padrão do Director é inserido no projetor. Esse ícone não aparece nos arquivos protegidos pois os mesmos não terão autonomia em nenhuma máquina (sempre estarão vinculados ao projetor principal).

O ícone inserido no projetor pode ser alterado de algumas maneiras e é altamente recomendável alterá-lo por questão de profissionalismo, embora isso não influencie na execução do software.

A maneira mais usual, recomendada até mesmo pela própria Macromedia em seu site, é o uso do software de edição de ícones chamado Microangelo (homenagem ao artista Michelangelo Buonarotti). Este software pode ser baixado de <http://www.impactsoft.com> (shareware) e alia simplicidade a uma boa funcionalidade. Ele permite uma boa edição de ícones e a troca de ícones de executáveis como os projetores criados no Director.

...

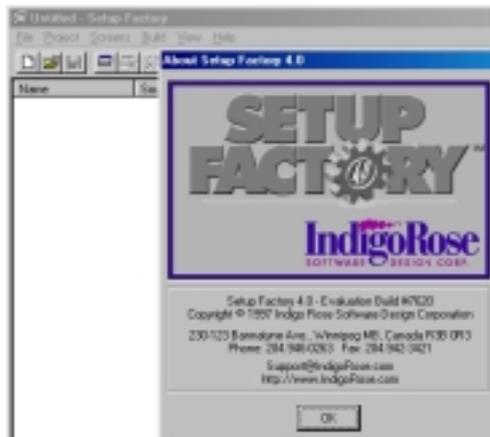
...

## 2 - Instalação do software :

Antes de finalizar um software para a replicação/distribuição, é conveniente prepará-lo de modo a facilitar sua instalação pelo usuário final. O Director possui uma vantagem em relação à maioria dos softwares de autoria que é a possibilidade de ser executado inteiramente de dentro do cd, sem que seja necessário instalar-se qualquer arquivo para o HD do usuário. Embora essa facilidade exista, eventualmente pode-se fazer necessário copiar-se certos arquivos que necessitem de alterações ou mesmo pode-se fazer conveniente a adição de atalhos para os executáveis do CD-ROM.

### Existem algumas opções quanto a isso :

Softwares como InstallShield ou o Setup Factory (<http://www.indigoroze.com>) podem ser usados normalmente, principalmente quando se fizer necessário a edição de arquivos .ini existente ou verificação do hardware e/ou softwares presentes na máquina do usuário.



Tela capturada do instalador Setup Factory:

## Listas

Listas (*lists*) nada mais são do que matrizes (também chamada *arrays* na maioria das linguagens), ou seja, uma “variável” que pode conter diversos elementos. Estes elementos aparecem dentro de colchetes e são separados por vírgulas no Lingo.

Ao contrário de certas linguagens, não é necessário especificar um determinado espaço de memória para uma lista no Lingo, pois isso é feito automaticamente à medida do necessário. Outro diferencial às outras linguagens de programação é que o número referente ao primeiro elemento de uma lista é “1” e não “0” como em outras linguagens.

...

...

### **Listas de Propriedades (property lists) :**

...

...

### **Alguns exemplos de listas de propriedades :**

...

...

### **Criando listas :**

...

...

### **Exemplos de declaração de listas :**

...

...

### **Exemplo de um movie usando lista (rastros de um sprite)**

...

...

### **Uma agenda telefônica simples :**

...

# Programação Orientada a Objetos & Parent Scripts

(conceitos básicos)

Ao contrário do que se pensa, a linguagem Lingo não é orientada a objetos, mas pode-se dizer que a mesma suporta uma programação orientada a objetos.

Orientação a objetos (OO) não vem a ser uma linguagem, e sim um conceito de programação! Trata-se de um complexo sistema dentro de discretas subsecções que são chamadas objetos. A essência da programação orientada a objetos é esse conteúdo unitário que coopera e comunica com cada outro pensamento através de protocolos dizendo o que fazer.

...

...

**Algumas vantagens da OO :**

...

...

**O Lingo possui 3 fatores que facilitam a OO :**

...

...

**The birth function :**

...

...

**Criando um child object proveniente de um parent Script :**

...

...

**Um *parent script* muito simples :**

...

...

## Definições Básicas

### **Lingo :**

...

...

### **Rotina :**

...

...

### **Função :**

...

...

### **Declaração de variáveis :**

...

...

### **Atribuição de Valores a Variáveis :**

...

...

### **Comentários no Lingo :**

...

...

### **Entrada e Saída de Valores de uma função :**

...

...

## Strings, conceitos fundamentais

...

...

### **Comandos e Funções para trabalhar com Strings :**

...

...

## Lingo - Dicas úteis

**1 - criando efeito de fade in e fade out sobre uma imagem :**

...

...

**2 - Encaixando sprites :**

...

...

**3 - Salvando um arquivo .txt com o Xtra FileIO :**

...

...

**4 - Lendo o conteúdo de um arquivo .txt com o Xtra FileIO :**

...

...

**5 - Abrindo um MIAW de 640X480 sobre configuração de 800X600 :**

...

...

**6- Exportando cast members como JPEGs usando os próprios Xtras do Director 7 :**

...