

Anexo XI

Como Inserir ícone na barra de título de um programa executável compilado em Clean

Este método para inserir um ícone na barra de títulos de um programa compilado em Clean, consiste em utilizar uma DLL (gentilmente criada pelo aluno Daniel Vaz do 6º período da graduação em Eng. Elétrica, em 2006) para realizar esta tarefa. Portanto são necessárias duas etapas bem distintas:

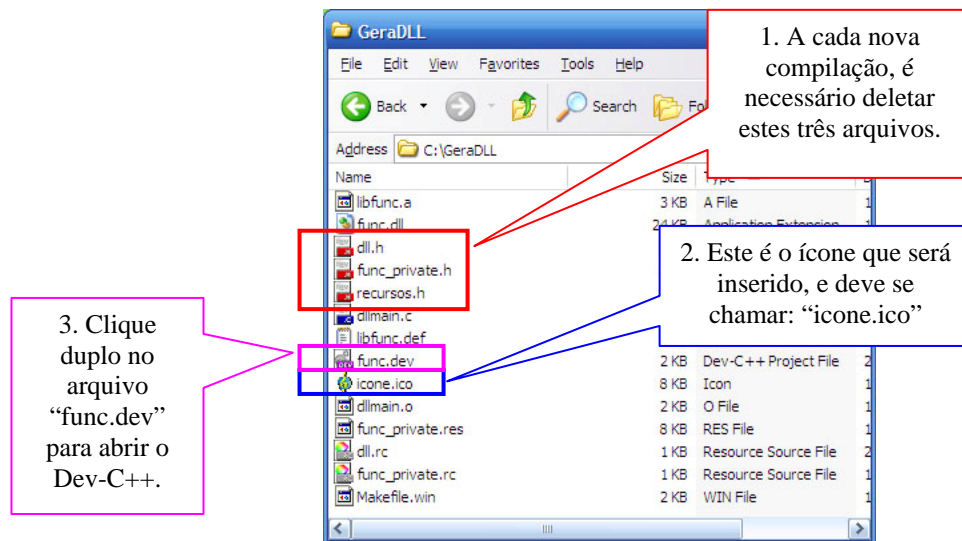
- Compilar a DLL a cada mudança de ícone;
- Alterar o arquivo **.icl** que será compilado pelo Clean.

Neste exemplo, foi utilizado o programa compilado em Clean “ToolBar”, cuja pasta encontra-se no CD anexo.

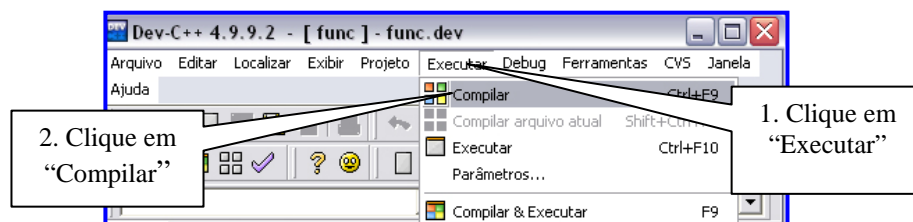
Obs: nos testes realizados, somente o Clean 2.1.1 conseguiu compilar sem problemas, utilizando a DLL “*func.dll*”.

Compilar a dll a cada mudança de ícone:

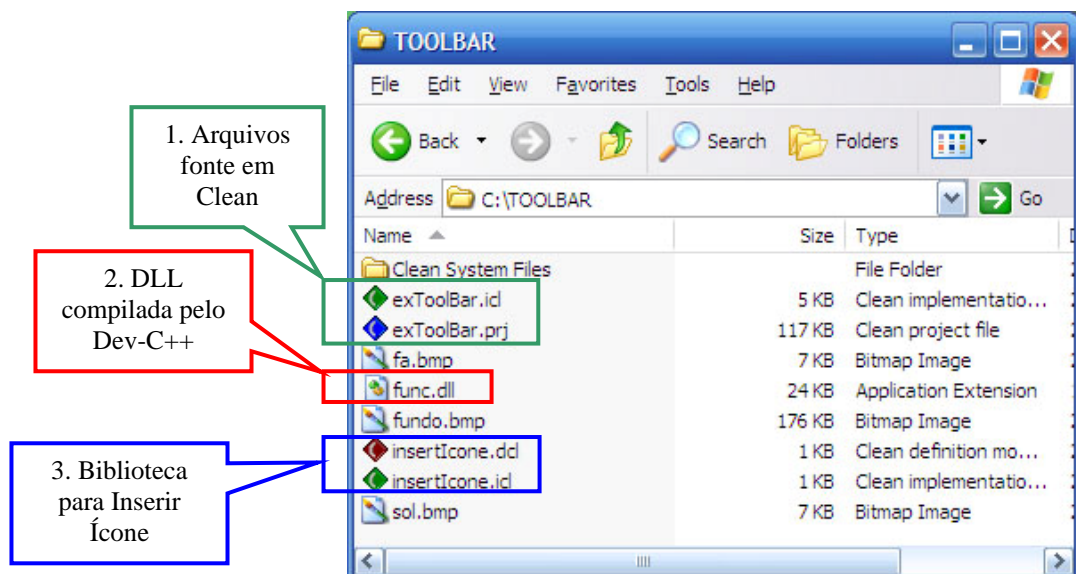
- Toda vez que alterar o ícone, uma nova DLL deve ser compilada. Para esta tarefa, utilizaremos o software freeware Dev-C++, cujo instalador se encontra anexo no CD;
- Copie a pasta “GeraDLL” que esta no CD para o HD. Copie para esta pasta, o ícone a ser inserido, o qual deve ter obrigatoriamente o nome: “ícone.ico” .



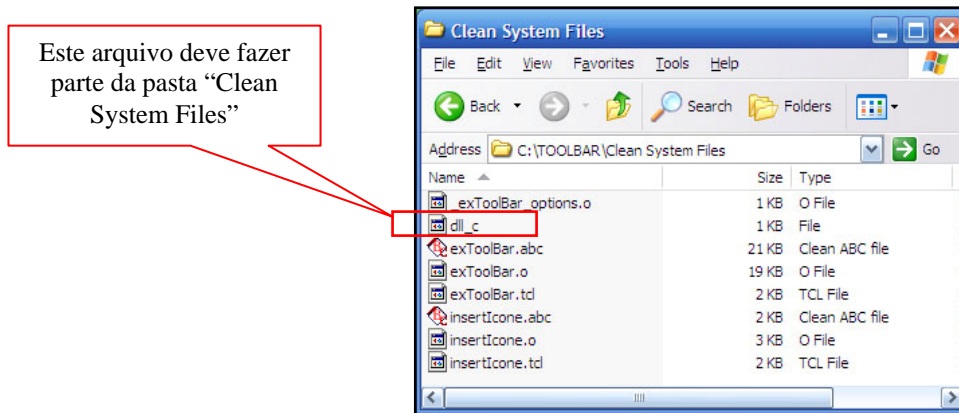
- Duplo clique no arquivo "func.dev" o Dev-C++ será aberto.



- Ao término da compilação será gerada a dll com o nome "**func.dll**", bastando agora, somente copiá-la para a pasta onde está o programa fonte do Clean.
- Os arquivos "**insertÍcone.dcl**" e "**insertÍcone.icl**", são as bibliotecas utilizadas pelo Clean.



- Outro arquivo obrigatório, que deve sempre estar dentro da pasta “Clean System Files” é o arquivo “**dll_c**”, que se encontra no CD anexo.



- Estando todos os arquivos em suas respectivas pastas, abordaremos a seguir a 2ª etapa, que é alterar o programa fonte do Clean.

Alterar o programa .icl – fonte em Clean.

- A primeira alteração, é importar a biblioteca “**insertIcone**” e o arquivo “**dll_c**”, como mostra a figura abaixo:

```
1 module exToolBar
2 import StdEnv, StdIO, insertIcone
3
4 import code from library "dll_c"
5
```

- Na estrutura onde abre o bitmap, deverá ser inserida a linha de instruções destacada na figura abaixo, onde a função “**clean_icone**” faz a chamada da **DLL** que contém o ícone:

```
44 = startIO SDI 0 (initialize ids) (attributes++[ProcessClose closeProcess]) world
45
46 initialize ids proc
47   # (maybeBitmap, proc) = openBitmap "fundo.bmp" proc
48   # bitmapFundo = fromJust maybeBitmap
49   # bitmapsize = getBitmapSize bitmapFundo
50   # (erro, proc) = openWindow Void (janela ids bitmapFundo bitmapsize) proc
51   | erro <> NoError = proc
52   # (erro, proc) = openMenu Void (menu) proc // ativa a toolbar -> SEM O OPENMENU AS T
53   | erro <> NoError = proc
54   # proc = clean_icone proc "Exemplo de ToolBar"
55   | otherwise = proc
56 where
57   menu = Menu "Arquivo"
58         (MenuItem "Sair" (MenuFunction ((noLS close
59
60 nada proc = proc
61
62
63
64
65
66 janela ids bitmap bitmapsize
67 = Window "Exemplo de ToolBar"
68   (
69     EditControl "" (PixelWidth 180) 2 [ControlPos (Center, zero), ControlHide]
70     ::
```

Linha adicionada ao programa.

Observe que o nome "Exemplo de ToolBar" deve ser o mesmo para o título da janela.

- Feito estas alterações, basta finalmente compilar o programa utilizando a versão Clean 2.1.1



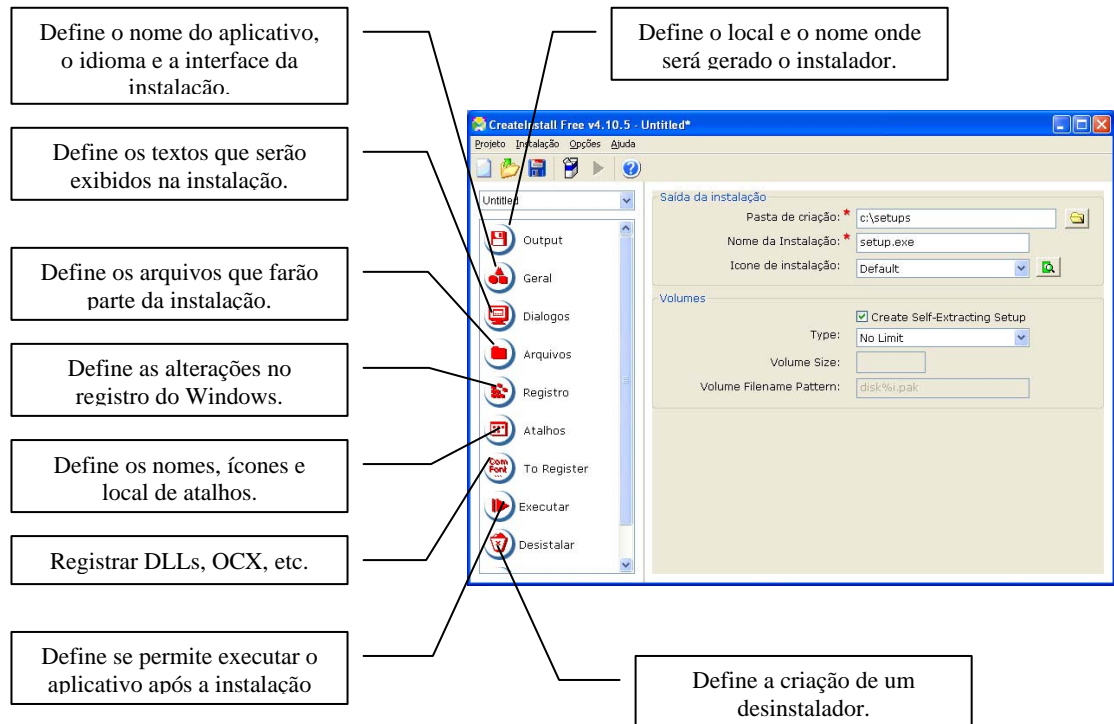
Criando um instalador com o Create Install Free

O Create Install Free, é de fácil utilização, é auto-explicativo e ótimos benefícios, e encontra-se anexo neste CD ou pode ser baixado do site: <http://www.createinstall.com>

No caso dos programas compilados em Clean, serão necessários para se criar um instalador, além do programa executável, todos os arquivos utilizados pelo programa executável como: bmps, mid, executáveis disparados pelo programa principal, ícones, fonte, etc., onde todos deverão ficar em uma pasta separada, para facilitar a criação da instalação. No exemplo da figura abaixo, foi criado um instalador do programa “Editor Didático F1”, cujo projeto encontra-se no CD anexo.

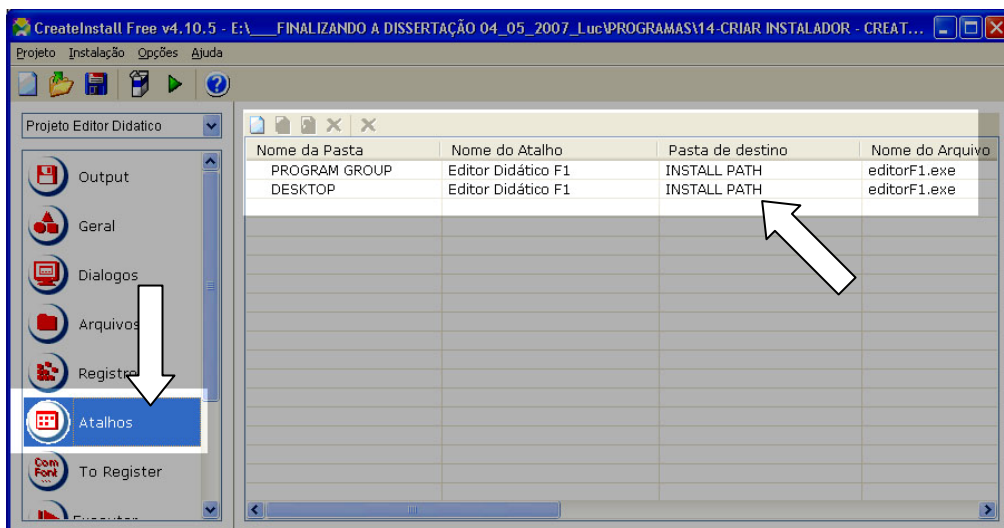
Por ser um programa auto-explicativo, serão comentados somente alguns itens considerados relevantes para uma instalação básica.

Interface principal do programa.

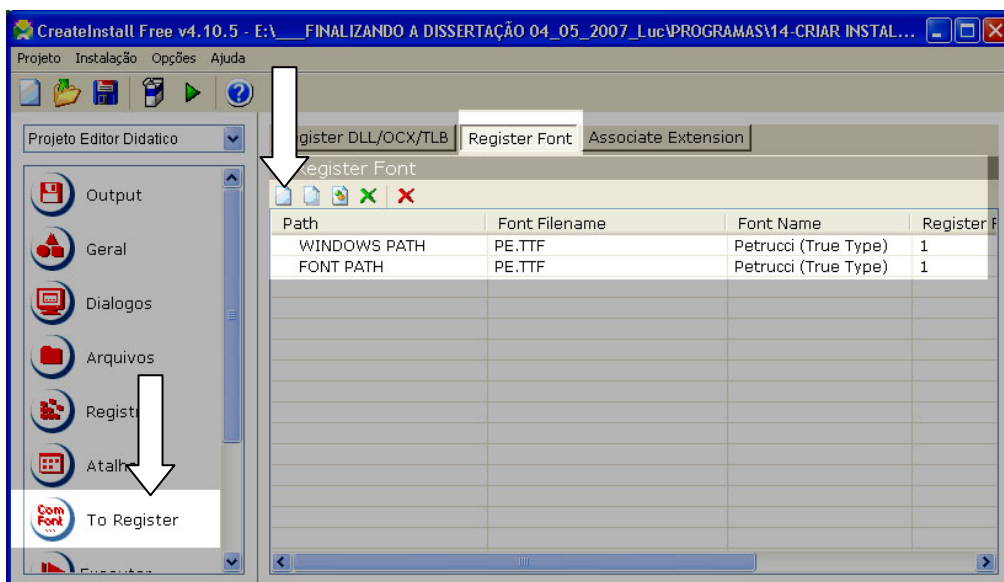


Procedimentos

- Crie um novo projeto;
- Configure os itens detalhados anteriormente;
- Para criação de Atalhos, configure a pasta de trabalho como “INSTALL PATH”;



- Para instalar uma fonte, adicione a mesma como na figura abaixo:



- 2.5 Salve o projeto;
- 2.6 Compile o projeto.

Índice Remissivo

—B—

| | |
|---|-----|
| Biblioteca Gera MIDI formato 0..... | 119 |
| Biblioteca Gera MIDI formato 1..... | 125 |
| Biblioteca <i>separaEventosF0</i> | 109 |

—C—

| | |
|-------------------------------|-----|
| Cabeçalho Principal | 54 |
| cabo MIDI | 8 |
| CANAL MIDI..... | 19 |
| conexões MIDI..... | 9 |
| contador MIDI..... | 28 |
| Conversor didático MIDI | 168 |

—D—

| | |
|--------------------------------|-----|
| Delta Time..... | 39 |
| <i>Dinamic Libraries</i> | 308 |

—E—

| | |
|------------------------------|-----|
| Especificação MIDI 1.0 | 201 |
| Extra Objects | 309 |

—F—

| | |
|------------------------|-----|
| Figuras Musicais | 224 |
| Formato 0 | 50 |
| Formato 1 | 51 |

—H—

| | |
|--------------------|----|
| Header Chunck..... | 54 |
|--------------------|----|

—I—

| | |
|--------------------------------------|-----|
| Instâncias e Sobrecarregamento | 229 |
|--------------------------------------|-----|

—M—

| | |
|----------------------------|--------|
| máquina MIDI | 33 |
| Mensagens MIDI..... | 21 |
| Metrônomo | 40 |
| MIDI..... | 6, 113 |
| MIDI em um computador..... | 11 |

—P—

| | |
|----------------------|--------|
| Polimorfismo | 226 |
| Ppq..... | 44, 48 |
| Protocolo MIDI | 14 |

—S—

| | |
|-----------------|----|
| Set Tempo | 46 |
|-----------------|----|

—T—

| | |
|----------------------------|-----|
| Tokens | 258 |
| Tonalidades | 219 |
| Trabalhos futuros..... | 195 |
| Track Chunck | 56 |
| Trilhas de mensagens | 56 |

—Z—

| | |
|------------------------|----|
| Zermelo-Fraenkel | 88 |
|------------------------|----|