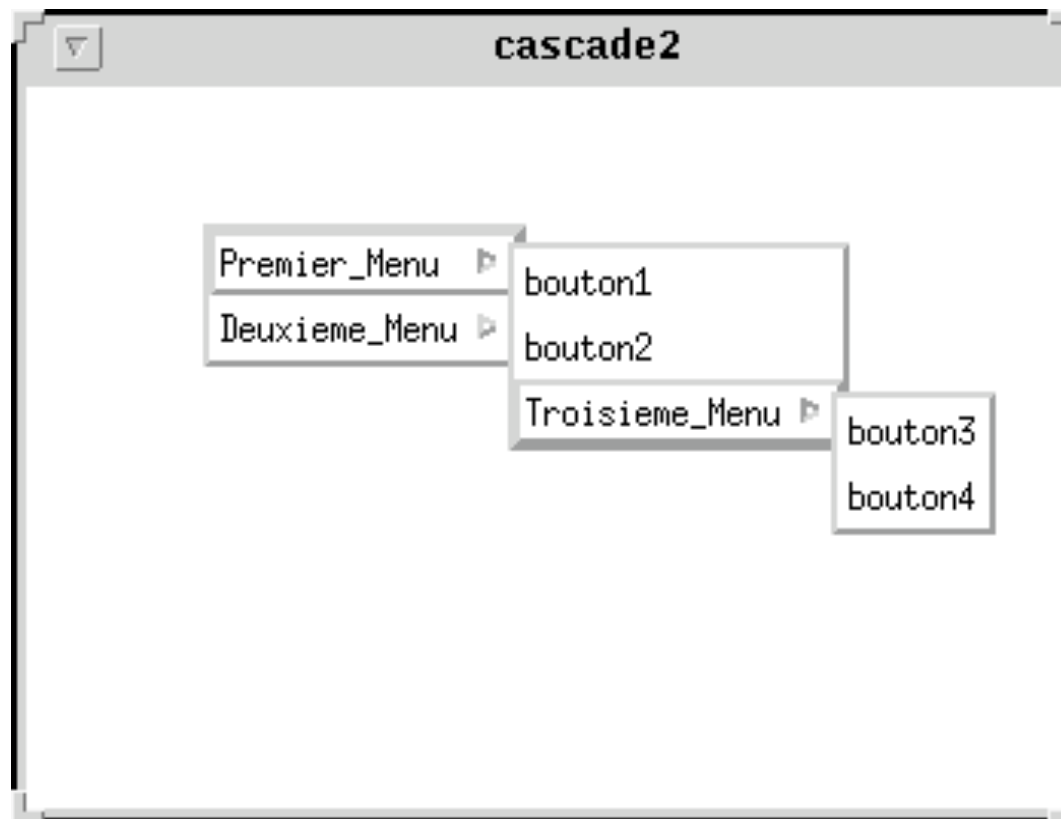
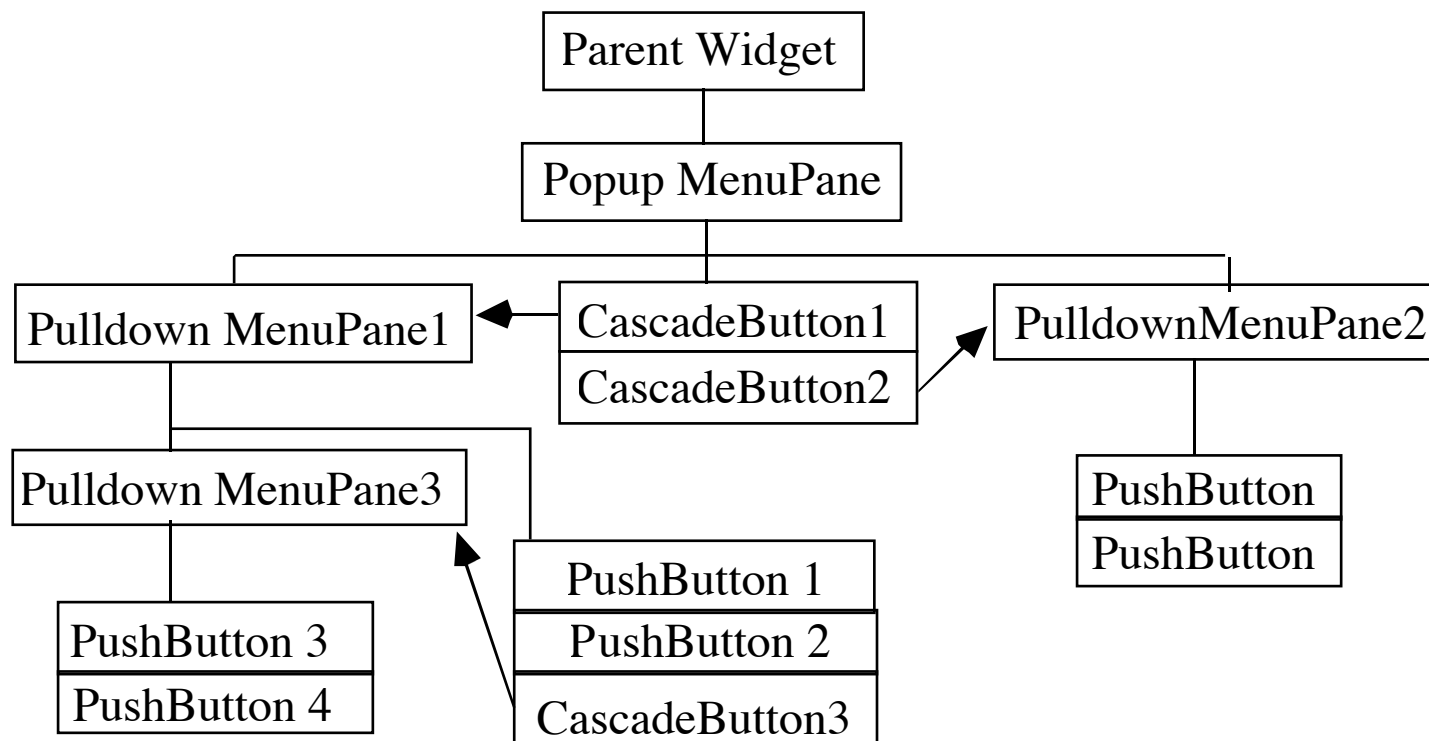


LES MENUS

Les menus popup



Les menus popup: Arborescence



Création de Menus popup

Pour créer un popup menu simple, on peut procéder en trois étapes:

1. Créer le popup menu avec `XmCreatePopupMenu` (parent, name, arglist, argcount)
2. Créer les items du menu fils du menu. Ils peuvent être des `XmLabel`, `XmPushButton`, `XmToggleButton`, `XmSeparator`, `XmCascadeButton` ou les gadgets correspondants.
3. Enregistrer les callbacks pour chaque `XmPushButton` ou `XmToggleButton`.
4. Pour créer un sous-menu, il faut créer séparément le sous-menu avec `XmCreatePulldownMenu` et **l'associer au `XmCascadeButton` du menu qui le fait apparaître via la ressource `XmNsubMenuId` du `CascadeButton`.**

Exemple de menu popup simple

```
#include <Xm/Xm.h>
#include <Xm/PushButton.h>
#include <Xm/RowColumn.h>
#include <Xm/BulletinB.h>

Widget create_popup_menu (parent)
Widget parent;
{
    Widget popup;

    popup = XmCreatePopupMenu (parent, "popup", NULL, 0);
    XtVaSetValues (popup, XmNmenuPost, "<Btn1Down>", NULL);
    XtVaCreateManagedWidget("Choix_1", xmPushButtonGadgetClass, popup, NULL);
    XtVaCreateManagedWidget("Choix_2", xmPushButtonGadgetClass, popup, NULL);
    XtVaCreateManagedWidget("Choix_3", xmPushButtonGadgetClass, popup, NULL);
    return popup;
}
```

Exemple de menu popup simple (suite)

```
void post_menu (Widget w, Widget popup, XButtonEvent * e)
{
    if (e -> button == Button1) {
        XmMenuPosition (popup, e);
        XtManageChild (popup);
    }
}

main (arg, argv) char** argv;
{
    Widget          top, canvas, popup;
    XtAppContext    app;

    top = XtVaAppInitialize (&app, "Test", NULL, 0, &argc, argv, NULL, NULL);
    canvas = XtVaCreateManagedWidget ("bb", xmBulletinBoardWidgetClass, top, NULL);
    popup = create_popup_menu (canvas);
    XtAddEventHandler(canvas, ButtonPressMask, False, post_menu, popup);
    XtRealizeWidget (top);
    XtAppMainLoop (app);
}
```

Exemple de menu popup en cascade

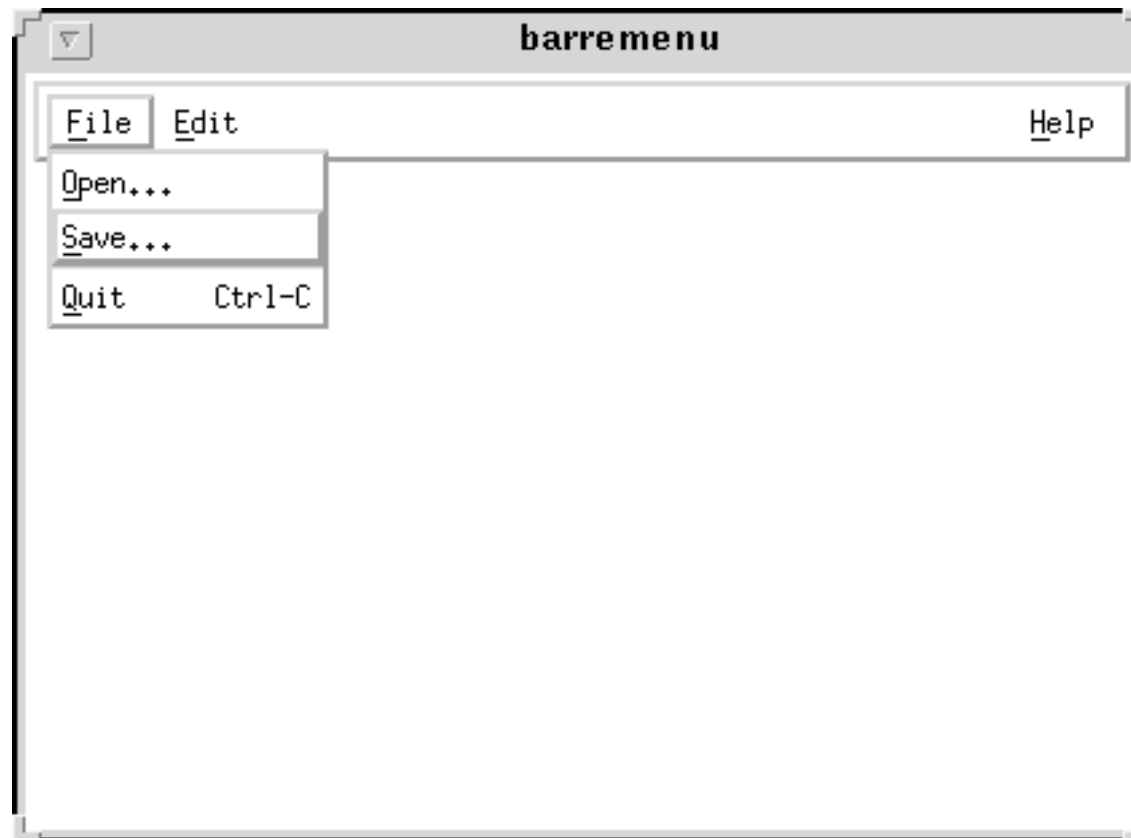
```
void create_sub_menu (Widget parent, char * titre)
{
    Widget      pdown;

    pdown = XmCreatePulldownMenu (parent, "pulldown", NULL, 0);
    XtVaCreateManagedWidget(titre, xmCascadeButtonGadgetClass, parent,
                            XmNsubMenuId, pdown, NULL);
    XtVaCreateManagedWidget("Choix_1", xmPushButtonGadgetClass, pdown, NULL);
    XtVaCreateManagedWidget("Choix_2", xmPushButtonGadgetClass, pdown, NULL);
}

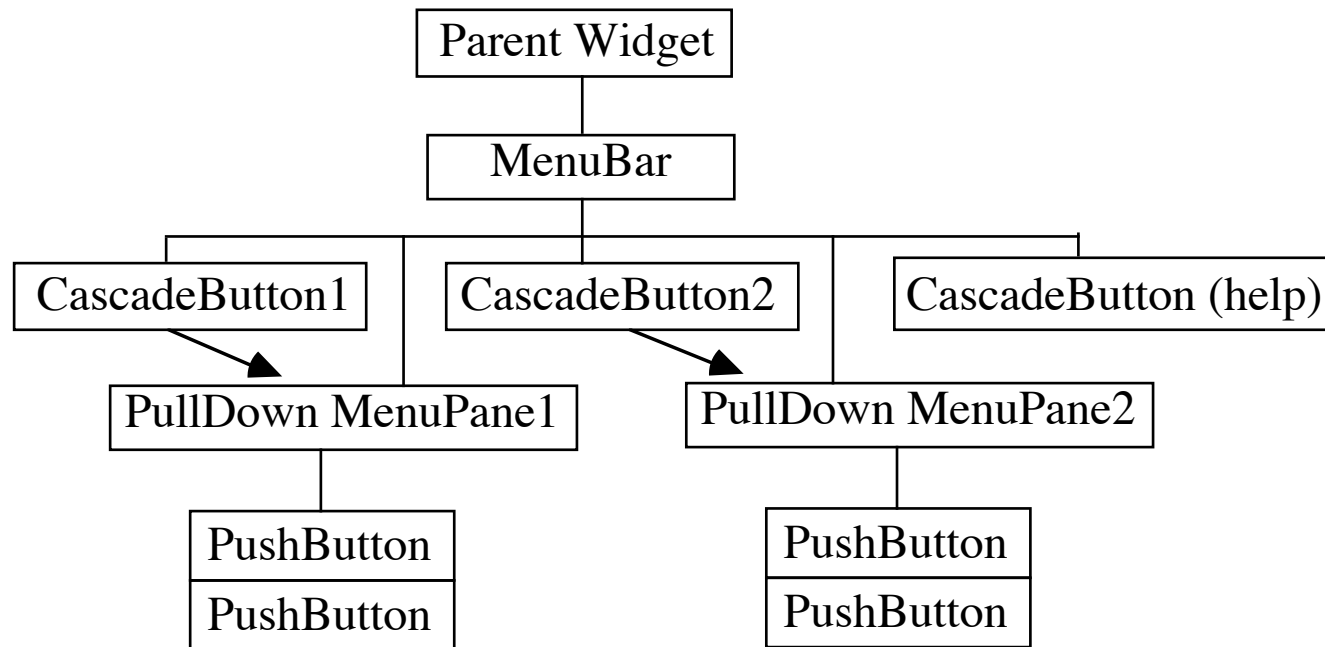
Widget create_popu_menu (Widget parent) {
    Widget  popup;

    popup = XmCreatePopupMenu (parent, "menu_popup", NULL, 0);
    XtVaSetValues (popup, XmNwhichButton, Button1, NULL);
    create_submenu (popup, "Titre1");
    create_submenu (popup, "Titre2");
    return popup;
}
```

Les menus déroulants



Les menus déroulants: arborescence



Les menus déroulants: Exemple

```
create_pulldown_menu (Widget menubar)
{
    Widget  pdown;

    pdown = XmCreatePulldownMenu (menubar, "pulldown", NULL, 0);
    (void) XtVaCreateManagedWidget ("cascade", xmCascadeButtonGadgetClass,
                                     menubar, XmNsubMenuId, pdown, NULL);
    XtManageChild(XmCreatePushButtonGadget(pdown,"choix_1", NULL, 0));
    XtManageChild(XmCreatePushButtonGadget(pdown,"choix_2", NULL, 0));
}
create_pulldown_menubar(Widget parent)
{
    Widget  menubar;
    menubar = XmCreateMenuBar (parent, "menubar", NULL, 0);
    XtManageChild (menubar);
    create_pulldown_menu (menubar);
}
```

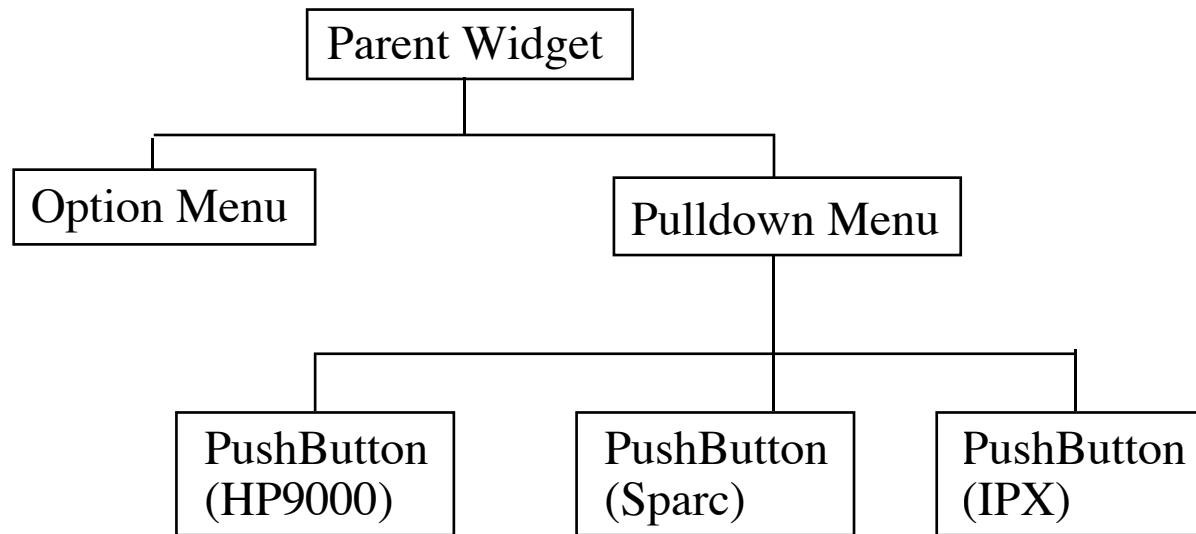
Les menus à options

Machine

Machine

HP9000
Sparc <input type="text"/>
IPX
RS6000

Les menus à options: Arborescence



Exemple de menu à options

```
#include <Xm/PushButton.h>
#include <Xm/CascadeB.h>
#include <Xm/RowColumn.h>
#include <Xm/BulletinB.h>
```

```
Widget create_option_menu (Widget parent) {
    Widget optmenu, pulldmenu;

    optmenu = XmCreateOptionsMenu (parent, "optmenu", NULL, 0);
    pulldmenu = XmCreatePulldownMenu (parent, "pulldmenu", NULL, 0);
    XtVaSetValues (optmenu, XmNsubMenuId, pulldmenu,
                  XmNlabelString, XmStringCreateLocalized ("Machine"), NULL);
    XtVaCreateManagedWidget ("HP9000", xmPushButtonWidgetClass, pulldmenu, NULL);
    XtVaCreateManagedWidget ("Sparc", xmPushButtonWidgetClass, pulldmenu, NULL);
    XtVaCreateManagedWidget ("IPX", xmPushButtonWidgetClass, pulldmenu, NULL);
    XtVaCreateManagedWidget ("RS6000", xmPushButtonWidgetClass, pulldmenu, NULL);
    XtManageChild(optmenu);
    return optmenu;
}
```

Exemple de menu à options (suite)

```
main (argc, argv)
char ** argv;
{
    Widget          top, canvas;
    XtAppContext    app;

    top = XtVaAppInitialize (&app, "Test", NULL, 0, &argc, argv, NULL, NULL);

    canvas = XtVaCreateManagedWidget ("bb", xmBulletinBoardWidgetClass, top, NULL);
    create_option_menu (canvas);

    XtRealizeWidget (top);
    XtAppMainLoop (app);
}
```

Les accélérateurs (raccourcis)

- Un accélérateur (ou raccourci) est une combinaison de touches qui peut être utilisée pour activer un bouton d'un menu sans afficher le menu
- Un raccourci peut être affecté à un bouton d'un menu particulier en utilisant la ressource `XmNaccelerator` supportée par toutes les sous-classes de `XmLabel`. La ressource `XmNaccelerator` est de type `char*` et utilise la même syntaxe que celle utilisée par les translations. Par exemple:

`Test*button1*accelerator: Ctrl<Key>X`

- Quand un accélérateur est installé, il est utile de l'indiquer sur le bouton de menu en utilisant la ressource `XmNacceleratorText`.

Cette ressource est de type `XmString` et peut contenir un texte tout à fait arbitraire, cependant le guide de style Motif recommande un style spécifique, par exemple:

`Test*button1*acceleratorText: Ctrl+X`

Les mnémoniques

- Les mnémoniques permettent aussi de sélectionner un bouton de menu sans utiliser la souris
- Un mnémonique est un caractère qui doit être l'un des caractères du labelString d'un bouton. En général, c'est le premier caractère qui est utilisé, mais en cas de conflit avec un autre bouton, un autre caractère peut être utilisé. Le caractère utilisé comme mnémonique apparaît automatiquement comme souligné
- Pour afficher le menu déroulant d'une barre de menu, on peut alors taper la clé <Alt> et le caractère mnémonique associé avec le cascade bouton de ce menu déroulant
- Lorsque le menu est affiché, pour activer un bouton, il suffira de taper le mnémonique associé à ce bouton s'il en possède un
- Un mnémonique est affecté en utilisant la ressource XmNmemonic de XmLabel, par exemple:

Test*button1*mnemonic: F

Menus détachables

- Depuis MOTIF 1.2, un menu déroulant peut être détaché (“tear off”) du cascade bouton auquel il est lié.
- La ressource XmNtearOffModel permet d’indiquer si un menu est détachable. Cette ressource peut prendre l’une des valeurs suivantes:

XmTEAR_OFF_ENABLE:	le menu est détachable
XmTEAR_OFF_DISABLE:	le menu n’est pas détachable (le défaut)

- Lorsqu’un menu est détachable, une ligne pointillée apparaît au dessus du premier item du menu. La sélection de cette ligne pointillée permet de détacher le menu.
- Une fois détaché, le menu reste à l’écran, ce qui permet de sélectionner des commandes sans avoir à faire apparaître le menu. Le menu peut être déplacé en utilisant les fonctionnalités du window manager. Il peut être rattaché en fermant sa fenêtre (à l’aide du window manager).

Fonctions de création de menus MOTIF

Il existe des fonctions MOTIF qui permettent la création rapide de menus complexes :

- `XmVaCreateSimpleMenuBar (parent, name, . . . , NULL)`
- `XmVaCreateSimpleOptionMenu (parent, name, option_label, option_mnemonic, button_set, callback, . . . , NULL)`
- `XmVaCreateSimplePopupMenu (parent, name, callback, . . . , NULL)`
- `XmVaCreateSimplePulldownMenu (parent, name, post_from_button, callback, . . . , NULL)`

XmVaCreateSimpleMenuBar: exemple

XmString file, edit, help;
Widget menubar, parent, widget;

```
file = XmStringCreateLocalized (“File”);
edit = XmStringCreateLocalized (“Edit”);
help = XmStringCreateLocalized (“Help”);
menubar = XmVaCreateSimpleMenuBar(parent, “menubar”,
                                   XmVaCASCADEBUTTON, file, ‘F’,
                                   XmVaCASCADEBUTTON, edit, ‘E’,
                                   XmVaCASCADEBUTTON, help, ‘H’,
                                   NULL);

XmStringFree (file);
XmStringFree (edit);
XmStringFree (help);

if (widget = XtNameToWidget (menubar, “button_2”))
    XtVaSetValues (menubar, XmNmenuHelpWidget, widget, NULL);
XtManageChild(menubar);
```

XmVaCreateSimplePulldownMenu

(parent, name, post_from_button, callback, ... , NULL)

XmString open, save, quit, quit_acc;
Widget menu;

```
open = XmStringCreateLocalized ("Open ..."); save = XmStringCreateLocalized ("Save ...");  
quit = XmStringCreateLocalized ("Quit"); quit_acc = XmStringCreateLocalized ("Ctrl-C");  
menu = XmVaCreateSimplePulldownMenu (menubar, "file_menu",
```

```
    0 /* 1er bouton de la barre de menu */ , file_cb,  
    XmVaPUSHBUTTON, open, 'O', NULL, NULL,  
    XmVaPUSHBUTTON, save, 'S', NULL, NULL,  
    XmVaSEPARATOR,  
    XmVaPUSHBUTTON, quit, 'Q', "Ctrl<Key>C", quit_acc,  
    NULL);
```

```
XmStringFree (open); XmStringFree (save); XmStringFree (quit); XmStringFree (quit_acc);
```

```
void file_cb (Widget w, XtPointer button_number, XtPointer call_data)  
{    /* activateCallback des boutons et valueChangedCallback pour les toggles */  
    printf("On a declenche le %d eme bouton du menu File\n", (int) button_number);  
}
```