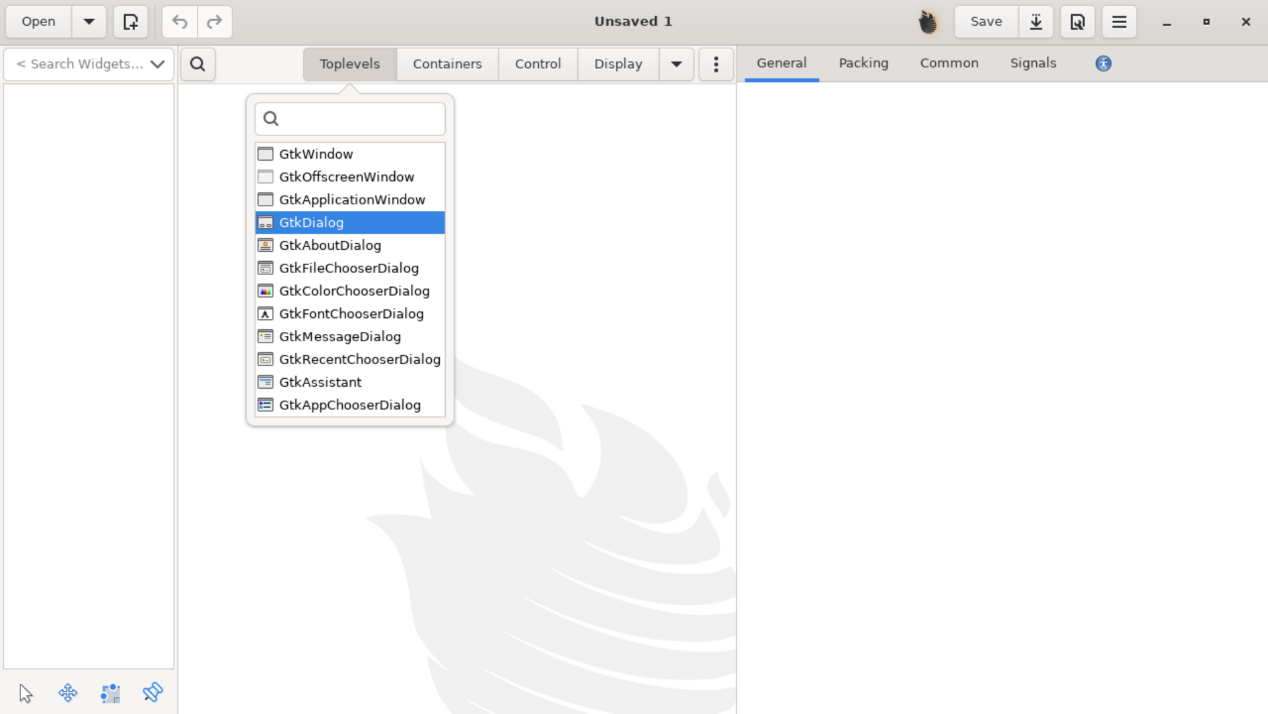
A updated and incomplete article of [Development/Create new dialog in Impress - The Document Foundation Wiki](https://wiki.documentfoundation.org/Development/Create_new_dialog_in_Impress) (2013) - by Jingze Xu 2023

## create and place .ui file

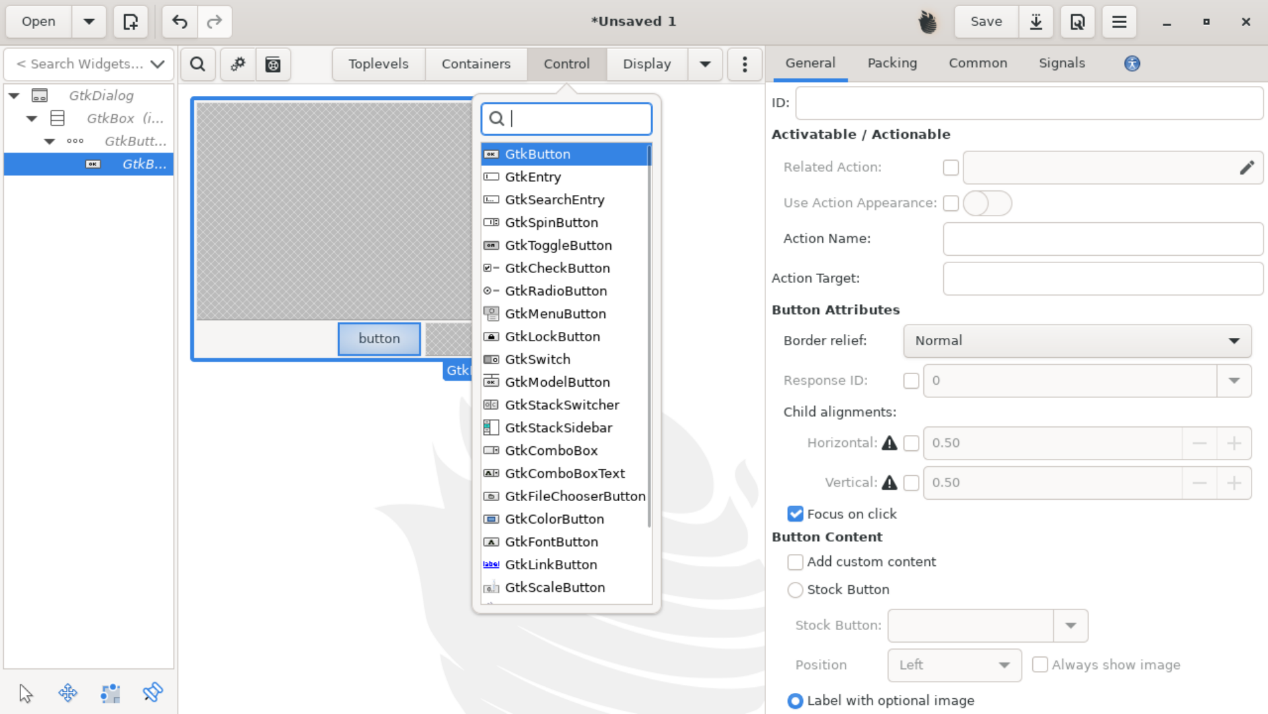
Open the glade tool.

|  |
| --- |
|  |
| glade |

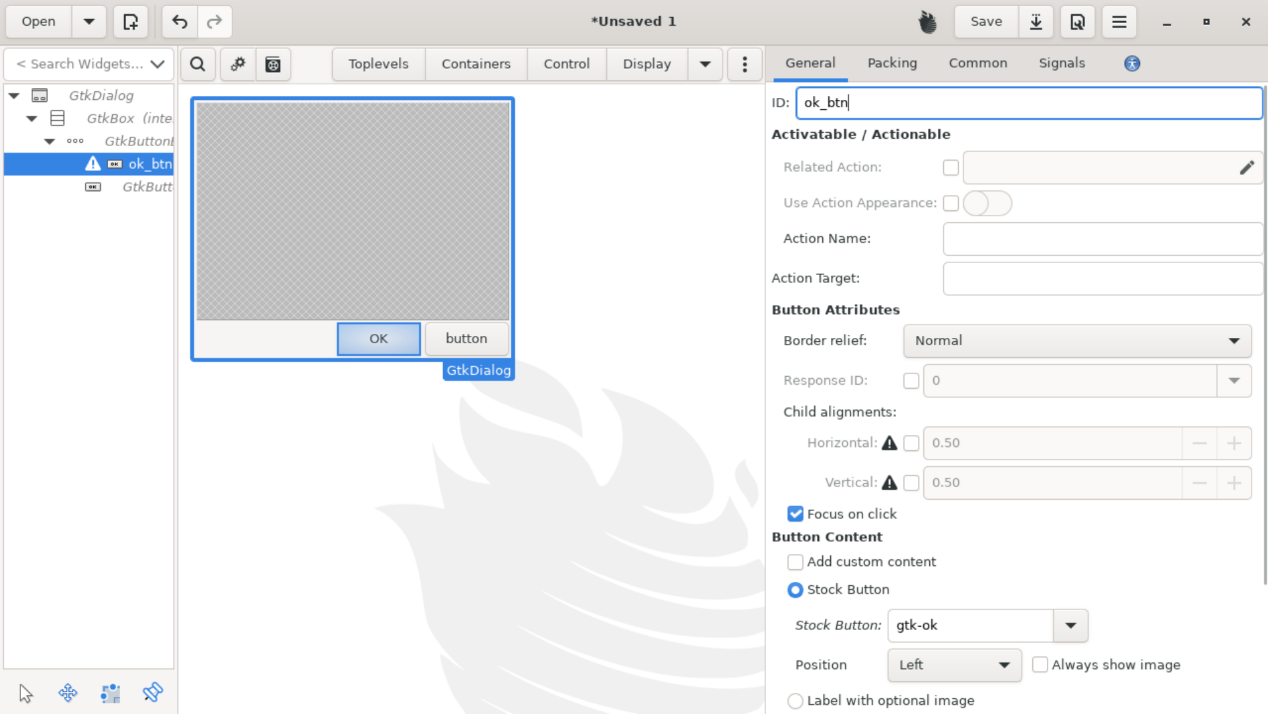
Create a new file，click GtkDialog in Toplevels column to create a dialog.



On the left panel, click GtkButtonBox，the corresponding area will be highlighted。Click GtkButton in Control column，then click GtkButtonBox area and place the button into it。



Set ID for the buttons. We set the left button’ID to “ok\_btn”, and the right button’s ID to “cancel\_btn”. We will use them later. Next, check stock button and choose ok/cancel in the menu.





Name other components. Set GtkDialog’s ID to HelloDialog, we will use it later.

图形用户界面

描述已自动生成

## code for the dialog

HelloDialog.hxx

#pragma once

#include "tools/link.hxx"

#include <vcl/weld.hxx>

namespace sd

{

class SdHelloDialog: public weld::GenericDialogController

{

public:

    SdHelloDialog(weld::Window\* pWindow);

    virtual ~SdHelloDialog() override;

private:

    std::unique\_ptr<weld::Button> m\_xCancelBtn;

    std::unique\_ptr<weld::Button> m\_xOKBtn;

    DECL\_LINK(CancelHdl, weld::Button&, void);

    DECL\_LINK(OKHdl, weld::Button&, void);

};

} // end of namespace sd

HelloDialog.cxx

#include "HelloDialog.hxx"

#include "strings.hrc"

#include "sdresid.hxx"

#include <vcl/weld.hxx>

namespace sd

{

SdHelloDialog::SdHelloDialog(weld::Window\* pWindow)

    : GenericDialogController(pWindow, "modules/simpress/ui/hellodialog.ui", "HelloDialog")

    , m\_xCancelBtn(m\_xBuilder->weld\_button("cancel\_btn"))

    , m\_xOKBtn(m\_xBuilder->weld\_button("ok\_btn"))

{

    m\_xCancelBtn->connect\_clicked(LINK(this, SdHelloDialog, CancelHdl));

    m\_xOKBtn->connect\_clicked(LINK(this, SdHelloDialog, OKHdl));

}

SdHelloDialog::~SdHelloDialog()

{

}

IMPL\_LINK\_NOARG(SdHelloDialog, CancelHdl, weld::Button&, void)

{

    m\_xDialog->response(RET\_CANCEL);

}

IMPL\_LINK\_NOARG(SdHelloDialog, OKHdl, weld::Button&, void)

{

    m\_xDialog->response(RET\_OK);

}

} // end of namespace sd

Place the .hxx and .cxx file in sd/source/ui/dlg/.

## modify a series of files

1. sd/uiconfig/simpress/menubar/menubar.xml

          <menu:menuseparator/>

          <menu:menuitem menu:id=".uno:Grid"/>

          <menu:menuitem menu:id=".uno:NavigationBar"/>



        </menu:menupopup>

      </menu:menu>

      <menu:menuseparator/>

      <menu:menuitem menu:id=".uno:HelloDialog"/>



    </menu:menupopup>

  </menu:menu>

  <menu:menu menu:id=".uno:FormatMenu">

    <menu:menupopup>

      <menu:menu menu:id=".uno:FormatTextMenu">

        <menu:menupopup>

          <menu:menuitem menu:id=".uno:Bold"/>

          <menu:menuitem menu:id=".uno:Italic"/>

1. sd/inc/app.hrc

#define SID\_PREVIEW\_STATE (SID\_SD\_START+329)

// FREE

#define SID\_SLIDE\_TRANSITIONS\_PANEL (SID\_SD\_START+334)

// test

#define SID\_HELLODIALOG (SID\_SD\_START+335)



#define SID\_PHOTOALBUM (SID\_SD\_START+336)

#define SID\_REMOTE\_DLG (SID\_SD\_START+337)

1. sd/sdi/drviewsh.sdi

SID\_PHOTOALBUM

[

ExecMethod = FuTemporary ;

StateMethod = GetMenuState ;

]

SID\_HELLODIALOG



[

ExecMethod = FuTemporary ;



StateMethod = GetMenuState ;

]

}

shell DrawViewShell

{

import ImpressEditView;

}

1. sd/sdi/sdraw.sdi

SfxVoidItem PhotoAlbumDialog SID\_PHOTOALBUM

()

[

AutoUpdate = FALSE,

FastCall = TRUE,

ReadOnlyDoc = FALSE,

Toggle = FALSE,

Container = FALSE,

RecordAbsolute = FALSE,

RecordPerItem;

AccelConfig = TRUE,

MenuConfig = TRUE,

ToolBoxConfig = TRUE,

GroupId = SfxGroupId::Options;

]

// testing

SfxVoidItem HelloDialog SID\_HELLODIALOG



()

[

AutoUpdate = FALSE,

FastCall = TRUE,

ReadOnlyDoc = FALSE,

Toggle = FALSE,

Container = FALSE,



RecordAbsolute = FALSE,

RecordPerItem;

AccelConfig = TRUE,

MenuConfig = TRUE,

ToolBoxConfig = TRUE,

GroupId = SfxGroupId::Options;

]

1. sd/source/ui/dlg/sddlgfact.hxx

    // test

    virtual VclPtr<VclAbstractDialog>          CreateSdHelloDialog(weld::Window\* pWindow) override;



    // For TabPage

    virtual CreateTabPage               GetSdOptionsContentsTabPageCreatorFunc() override;

    virtual CreateTabPage               GetSdPrintOptionsTabPageCreatorFunc() override;

    virtual CreateTabPage               GetSdOptionsMiscTabPageCreatorFunc() override;

    virtual CreateTabPage               GetSdOptionsSnapTabPageCreatorFunc() override;

};

1. sd/source/ui/dlg/sddlgfact.cxx

// test

#include "HelloDialog.hxx"

VclPtr<VclAbstractDialog> SdAbstractDialogFactory\_Impl::CreateSdPhotoAlbumDialog(weld::Window\* pParent, SdDrawDocument\* pDoc)

{

    return VclPtr<SdAbstractGenericDialog\_Impl>::Create(std::make\_unique<sd::SdPhotoAlbumDialog>(pParent, pDoc));

}

// test

VclPtr<VclAbstractDialog> SdAbstractDialogFactory\_Impl::CreateSdHelloDialog(weld::Window\* pWindow)



{

    return VclPtr<SdAbstractGenericDialog\_Impl>::Create(std::make\_unique<sd::SdHelloDialog>(pWindow));

}

1. sd/inc/sdabstdlg.hxx

    virtual VclPtr<VclAbstractDialog> CreateSdPhotoAlbumDialog(weld::Window\* pWindow, SdDrawDocument\* pDoc) = 0;

    // test

    virtual VclPtr<VclAbstractDialog> CreateSdHelloDialog(weld::Window\* pWindow) = 0;



protected:

    ~SdAbstractDialogFactory() {}

};

1. sd/source/ui/view/drviews2.cxx

        case SID\_PHOTOALBUM:

        {

            SdAbstractDialogFactory\* pFact = SdAbstractDialogFactory::Create();

            vcl::Window\* pWin = GetActiveWindow();

            ScopedVclPtr<VclAbstractDialog> pDlg(pFact->CreateSdPhotoAlbumDialog(

                pWin ? pWin->GetFrameWeld() : nullptr,

                GetDoc()));

            pDlg->Execute();

            Cancel();

            rReq.Ignore ();

        }

        break;

        // test

        case SID\_HELLODIALOG:



        {

            SdAbstractDialogFactory\* pFact = SdAbstractDialogFactory::Create();

            vcl::Window\* pWin = GetActiveWindow();

            ScopedVclPtr<VclAbstractDialog> pDlg(pFact->CreateSdHelloDialog(

                pWin ? pWin->GetFrameWeld() : nullptr));

            pDlg->Execute();

            Cancel();

            rReq.Ignore ();

        }

        break;

1. sd/Library\_sdui.mk

add “sd/source/ui/dlg/HelloDialog \” under “$(eval $(call gb\_Library\_add\_exception\_objects,sdui,\”.

1. sd/UIConfig\_simpress.mk

add “sd/uiconfig/simpress/ui/hellodialog \” under “$(eval $(call gb\_UIConfig\_add\_uifiles,modules/simpress,\”.

1. officecfg/registry/data/org/openoffice/Office/UI/DrawImpressCommands.xcu

      <node oor:name=".uno:PhotoAlbumDialog" oor:op="replace">

        <prop oor:name="Label" oor:type="xs:string">

          <value xml:lang="en-US">Photo Album</value>

        </prop>

        <prop oor:name="Properties" oor:type="xs:int">

          <value>1</value>

        </prop>

      </node>

      <node oor:name=".uno:HelloDialog" oor:op="replace">



        <prop oor:name="Label" oor:type="xs:string">

          <value xml:lang="en-US">Open Hello Dialog</value>

        </prop>

        <prop oor:name="Properties" oor:type="xs:int">



          <value>1</value>

        </prop>

      </node>

## rebuild and run

|  |
| --- |
|  |
| make sd |

|  |
| --- |
|  |
| make |

|  |
| --- |
|  |
| instdir/program/soffice --impress |

