
Subject: Re: Japanese IME test code
Posted by [mobilehunter](#) on Fri, 09 May 2008 00:16:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Mirek,
Sorry about virtual functions. Maybe we can remove some.
Please also consider the X11 implementations

Here my codes for X11:
at CtrlCore.cpp

```
#ifdef PLATFORM_X11
static int ImePreEditStartCB(XIC xic, XPointer clientData, XPointer callData);
static void ImePreEditDoneCB(XIC xic, XPointer clientData, XPointer callData);
static void ImePreEditDrawCB(XIC xic, XPointer clientData, XIMPreeditDrawCallbackStruct
*callData);
static void ImePreEditCaretCB(XIC xic, XPointer clientData, XIMPreeditCaretCallbackStruct
*callData);
#endif
```

at Ctrl.cpp

```
#ifdef PLATFORM_X11
int Ctrl::ImePreEditStartCB(XIC xic, XPointer clientData, XPointer callData)
{
    Ctrl *ctrl = (Ctrl*)clientData;

    Ptr<Ctrl> pfocusCtrl = ctrl->GetFocusCtrl();
    if(!pfocusCtrl)
        return -1;

    pfocusCtrl->ImePreEditStart();
    return -1;
}

void Ctrl::ImePreEditDoneCB(XIC xic, XPointer clientData, XPointer callData)
{
    Ctrl *ctrl = (Ctrl*)clientData;

    Ptr<Ctrl> pfocusCtrl = ctrl->GetFocusCtrl();
    if(!pfocusCtrl)
        return;
    pfocusCtrl->ImePreEditEnd();
}

void Ctrl::ImePreEditDrawCB(XIC xic, XPointer clientData, XIMPreeditDrawCallbackStruct
*callData)
```

```

{
    Ctrl *ctrl = (Ctrl*)clientData;
    Ptr<Ctrl> pfocusCtrl = ctrl->GetFocusCtrl();

    if(callData)
    {
        int caretPos = callData->caret;
        int changeFirst = callData->chg_first;
        int changeLength = callData->chg_length;

        if(callData->text)
        {
            int length = callData->text->length;

            if(!callData->text->encoding_is_wchar)
            {
                String newString = callData->text->string.multi_byte;
                WString newWString = ToUnicode(newString, newString.GetLength(), CHARSET_UTF8);

                pfocusCtrl->ImePreEditSetText(newWString);
            }
        }
    }
}

```

```

void Ctrl::ImePreEditCaretCB(XIC xic, XPointer clientData, XIMPreeditCaretCallbackStruct
*callData)
{
    Ctrl *ctrl = (Ctrl*)clientData;
    Ptr<Ctrl> pfocusCtrl = ctrl->GetFocusCtrl();

    if(!pfocusCtrl)
        return;

    pfocusCtrl->imePreEditCursor = callData->position;
    pfocusCtrl->ImePreEditSyncCaret();
    pfocusCtrl->SyncCaret();
}
#endif

```

At X11Wnd.cpp

```

/*cw.xic = xim ? XCreateIC(xim,
    XNInputStyle, XIMPreeditNothing|XIMStatusNothing,
    XNClientWindow, w,
    XNFocusWindow, w,
    NULL)

```

```

        : NULL;*/
//Added for CJK
//Using on the spot way
XIMCallback PreEditStartCallback;
XIMCallback PreEditDoneCallback;
XIMCallback PreEditDrawCallback;
XIMCallback PreEditCaretCallback;

PreEditStartCallback.client_data = (XPointer)this;
PreEditStartCallback.callback = (XIMProc)ImePreEditStartCB;

PreEditDoneCallback.client_data = (XPointer)this;
PreEditDoneCallback.callback = (XIMProc)ImePreEditDoneCB;

PreEditDrawCallback.client_data = (XPointer)this;
PreEditDrawCallback.callback = (XIMProc)ImePreEditDrawCB;

PreEditCaretCallback.client_data = (XPointer)this;
PreEditCaretCallback.callback = (XIMProc)ImePreEditCaretCB;

XVaNestedList preEditAttribute;
XPoint imePosition;
XFontSet fs;
XIMStyle overSpot=XIMPreeditPosition | XIMStatusNothing;
XIMStyle onSpot=XIMPreeditCallbacks | XIMStatusNothing;
XIMStyle uppStyle = onSpot;
char **missing_list;
int missing_count;
char *def_string;
fs = XCreateFontSet(Xdisplay, "-*-*-R-Normal--14-130-75-75-*-*", &missing_list,
&missing_count, &def_string);

imePosition.x=imePosition.y=0;
preEditAttribute = XVaCreateNestedList(NULL,
XNSpotLocation, &imePosition,
XNFontSet, fs,
XNPreeditStartCallback, &PreEditStartCallback,
XNPreeditDoneCallback, &PreEditDoneCallback,
XNPreeditDrawCallback, &PreEditDrawCallback,
XNPreeditCaretCallback, &PreEditCaretCallback,
NULL);

cw.xic = xim ? XCreateIC(xim,
XNInputStyle, uppStyle,
XNClientWindow, w,
XNFocusWindow, w,
XNPreeditAttributes, preEditAttribute,
NULL)

```

```
        : NULL;  
XFree(preEditAttribute);  
//End of Added for CJK
```
