

MMIX-Edit

Felix Albrecht

Inhalt

- Einleitung
- Was ist Scintilla
- Wie baut man eine Win32 Anwendung
- Wie integriert man Scintilla
- Wie integriert man MMIX

Einleitung

- Bachelorarbeit
- Erstellung einer MMIX IDE für Windows Systeme mit Hilfe von Scintilla

Einleitung

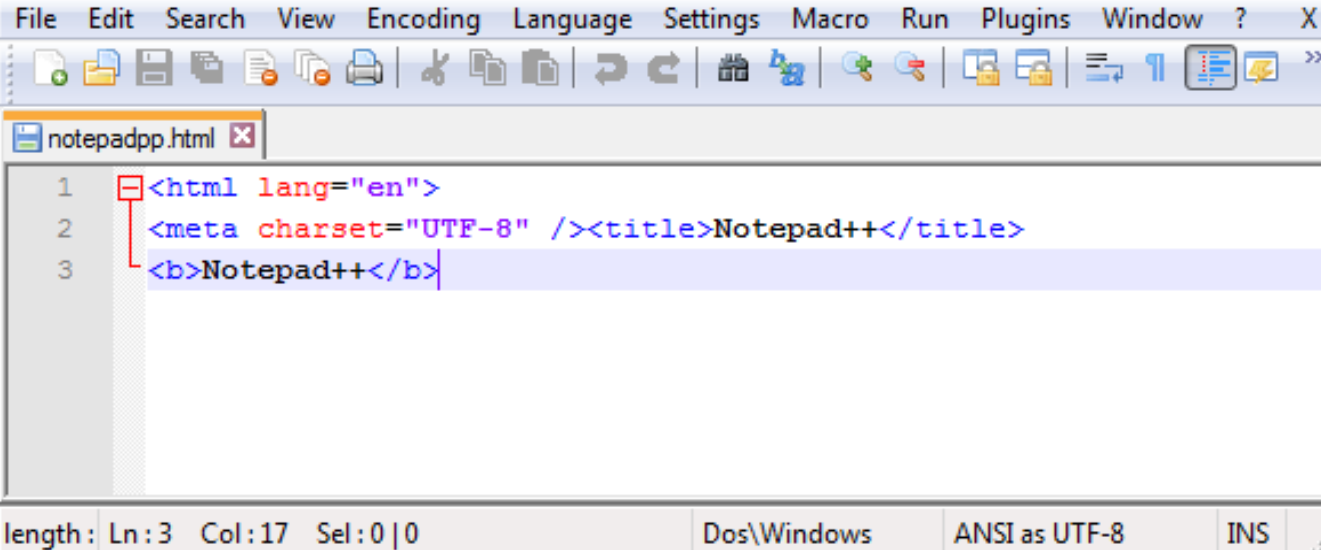
Was war benötigt:

- Win32 Anwendung
- Scintilla als Editorkomponente
- MMIX im Hintergrund

Scintilla

- Texteditorkomponente
- Open-Source

Scintilla



```
1 <html lang="en">
2 <meta charset="UTF-8" /><title>Notepad++</title>
3 <b>Notepad++</b>
```

length: Ln:3 Col:17 Sel:0|0 Dos\Windows ANSI as UTF-8 INS

Scintilla

- Enthalten in vielen Editoren wie
Notepad++, SciTE, TeXnicCenter, usw.
- Mehrere Ports:
Win32, MUI (AmigaOS),
Gtk (GIMP-Toolkit), Qt (mit Gtk in X11),
usw.

Scintilla

- Wichtige Funktionen von Scintilla:
 - Code-Highlighting
 - Marker-Setzung
 - Zeilennummern

Win32

- Win32 Funktionen arbeiten mit `wchar_t`
- `char*/String` müssen in `wchar_t` umgewandelt werden
- `"String"` => `L"String"`

Win32

```
int WINAPI WinMain(...)
```

- Einstiegspunkt in die Anwendung
- Enthält die Nachrichtenschleife

Win32

WNDCLASSEX windowClass

- speichert Informationen der Anwendung
- benötigt zum Erstellen des Hauptfensters

Win32

Atom WINAPI RegisterClassEx(...)

- Registrieren des Hauptfensters
- benötigt die WNDCLASSEX

Win32

HWND WINAPI CreateWindow([...])

- erstellt registriertes Hauptfenster
- gibt Handle des Fensters zurück

Win32

BOOL WINAPI ShowWindow(...)

- ändert Sichtbarkeit des angegebenen Fensters
- Minimierung oder Maximierung möglich

Win32

Nachrichtenschleife

- while mit GetMessage
- leitet Nachrichten weiter

Win32

BOOL WINAPI GetMessage([...])

- holt oberste Nachricht von Messagequeue
- filtert Nachrichten nach Kriterien

Win32

LRESULT WINAPI DispatchMessage([...])

- leitet Nachricht an richtiges Fenster weiter
- bei Hauptfenster an WndProc

Win32

LRESULT CALLBACK WndProc(...)

- verarbeitet Nachrichten an Fenster
- muss WNDCLASSEX zugeordnet werden

Scintilla Integration

- Scintilla.h
muss inkludiert werden
- Scintilla.dll
braucht nicht eingebunden werden
muss nur in Ordner sein

Scintilla Integration

- Neuer Editor mit CreateWindow
- lpClassName = L"Scintilla"
lpWindowName = L"Source"

Scintilla Integration

- SciLexer.h muss eingebunden sein
- SciLexer.DLL muss mit LoadLibrary geladen werden

Scintilla Integration

- MMIX Lexer existiert bereits
- SendMessage zum Ändern des Lexers
- SCI_SETLEXER und SCLEX_MMIXAL

MMIX Integration

- MMIX Quellcode in .w Dateien
- Change files in .ch Dateien
- Mit Ctangle zu .c Dateien kompilieren

MMIX Integration

@x

original Text

@y

neuer Text

@z

MMIX Integration

- Änderungen müssen in richtiger Reihenfolge sein

MMIX Integration

- Exit und Return müssen geändert werden
- Sourcecode muss direkt übergeben werden können

MMIX Integration

- Einstiegsfunktionen müssen Prototyp in cpp haben
- Einbindung als EXTERN_C Funktion

Fragen!