# C/C++ API für Einsteiger

Ulrich Krause

08. – 10.03.2010, Maritim Hotel, Gelsenkirchen

# Über mich

- Ulrich Krause
- Lotus Notes / Domino seit 1993
- Entwickler
- Administrator
- OpenNTF Projekte  !!HELP!! , Trigger Happy
- Moderator atnotes.de
- Blog www.eknori.de

# Agenda

- Grundlagen
- Setup
  - Windows
  - Linux
- Erstellen von Anwendungen
- Debug von Anwendungen
- Samples
  - Konsolenanwendung
  - DLL
  - Rich Text
- LotusScript eXtension toolkit (a.k.a LSX Toolkit)
- LS2CAPI

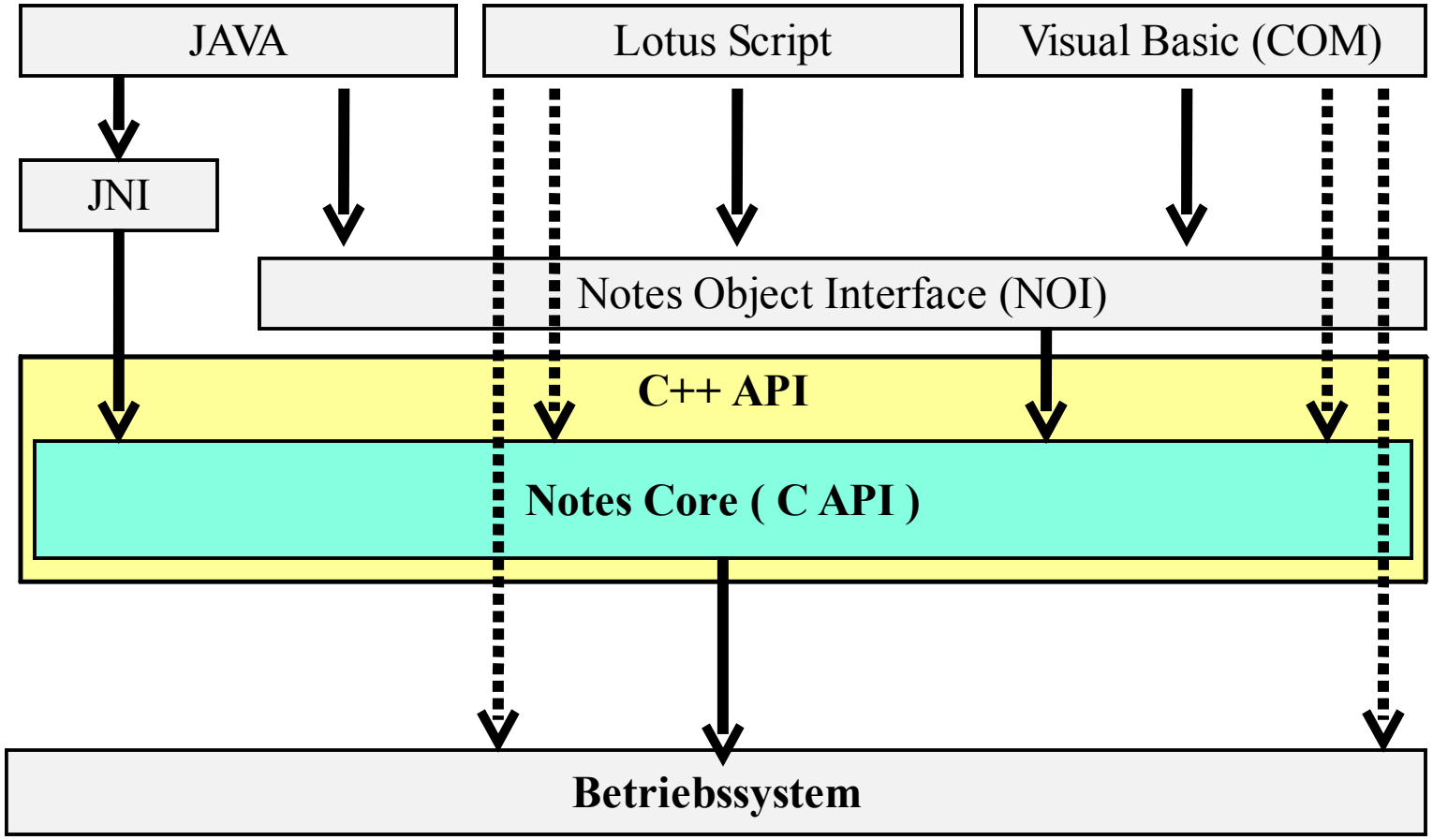
# Grundlagen

# Playing with Napalm and RazorBlades

*"A word of warning. This is difficult, horrible, tedious frustrating error prone work that WILL take far more time than you think, and WILL come back and bite you in the .. leg .."*

( Bill Buchan )

# Zugriffe


JAVA
Lotus Script
Visual Basic (COM)
JNI
Notes Object Interface (NOI)
C++ API
Notes Core ( C API )
Betriebssystem

# Welche Möglichkeiten gibt es

Lotus Notes / Domino C API

Lotus Notes / Domino C++ API

- C API Wrapper
- C / C++ kann zusammen verwendet werden

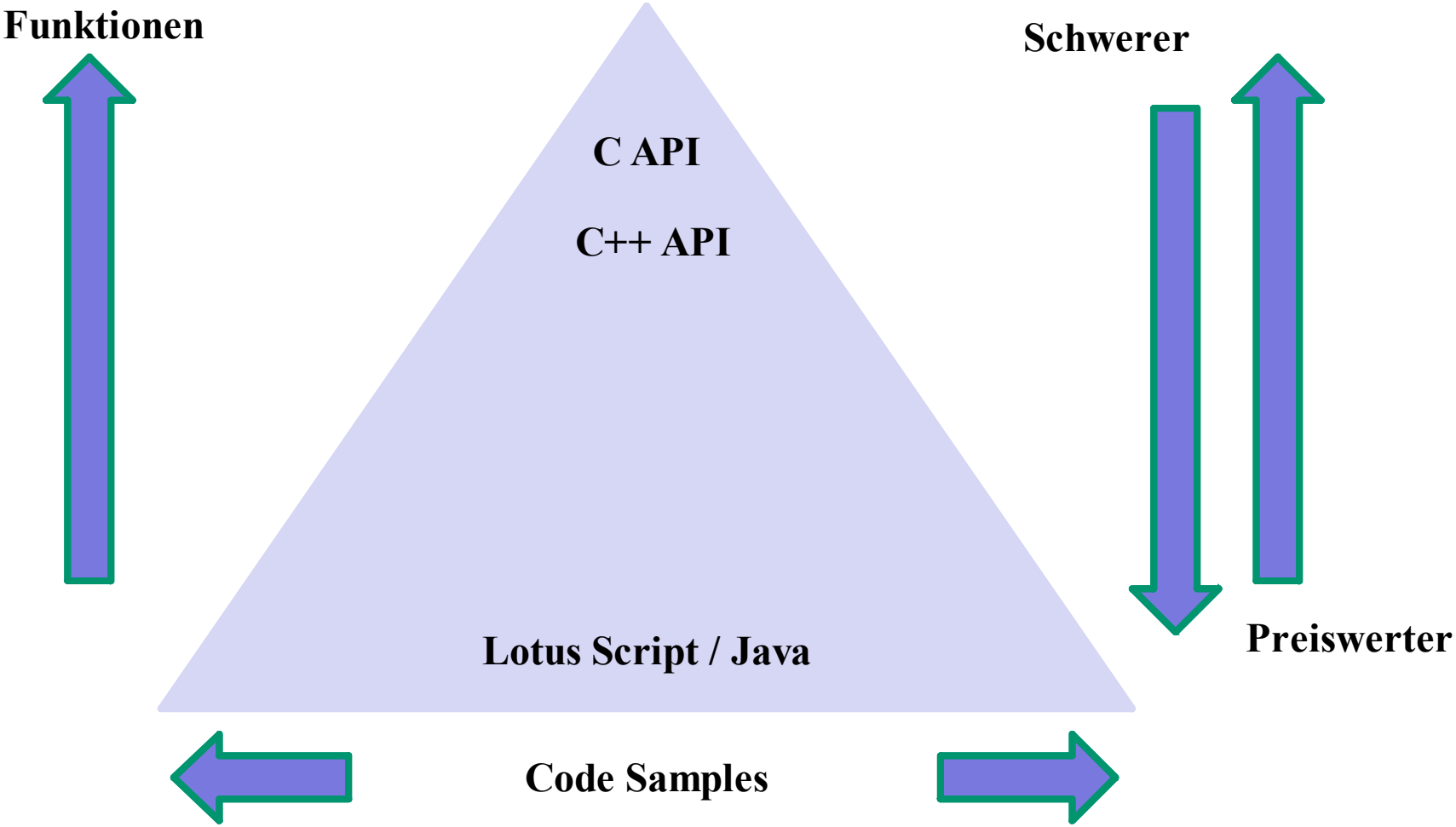
LotusScript eXtension toolkit (a.k.a LSX Toolkit)

LS2API

# Was kann ich damit machen ?

- Stand Alone Anwendungen
- Addin
  - Server
  - Workstation
- Notes workstation  Import / Export
- Notes workstation menu add-ins
- Extension Manager
- Datenbank Hooks
  - OpenNTF:Save any Note as MIME
- Drivers for external (non-Notes) databases

# Dreicksgeschichten



# Welche Tools brauche ich?

## Toolkits

http://www.ibm.com/developerworks/lotus/downloads/toolkits.html

## Compiler

- Windows
  http://www.microsoft.com/germany/Express/
  *Visual C++ 2005 EE (http://www.codezone.de/VisualStudioExpressImageISO.Codezone)*
- Linux
  Gcc (in der jeweiligen Distribution enthalten)
- AIX
  Using the GNU C/C++ compiler on AIX
  *http://www.ibm.com/developerworks/aix/library/au-gnu.html*

## Tools

- NotesPeek
  (http://www-01.ibm.com/support/docview.wss?uid=swg24005686)
- Ytria ScanEZ (trial)
  (http://www.ytria.com/WebSite.nsf/Er_Download?ReadForm&Lang=en)

# Compiler Matrix ( Windows )

| Produkt | C / C++ | LSX | kostenlos |
|---|---|---|---|
| Visual Studio 2005 | ja | ja | nein |
| Visual C++ 2005 EE | ja | ja | ja |
| Visual C++ 2008 EE | ja | nein * | ja |
| Visual C++ 2010 | n/a | n/a | n/a |

*= *nicht mit SDK 6.0a und 6.1 wg. Typunverträglichkeit in WinReg.h (LSTATUS)*

- Andere Compiler / IDE
  - **Code::Blocks (MinGW)**
    - http://www.codeblocks.org/downloads
    - http://www.darkmist.net/~schallee/notes/mingw_notesapi.html
  - **Eclipse**

# C vs. C++

| | C | C++ |
|---|---|---|
| granularer | ja | nein |
| aktuell | ja | nein |
| zusätzliche DLL erforderlich | nein | ja |
| Code | mehr/ prozedural | weniger/ objektorientiert |

# Setup (Windows)

# Setup

- **Visual Studio 2005 Express C++**
- Servicepack SP1 (KB926748)
- Patch für Vista / Windows 7 (KB932234)
- Platform SDK ( Windows Server 2003 SP1 )
- SDK in Compiler anmelden

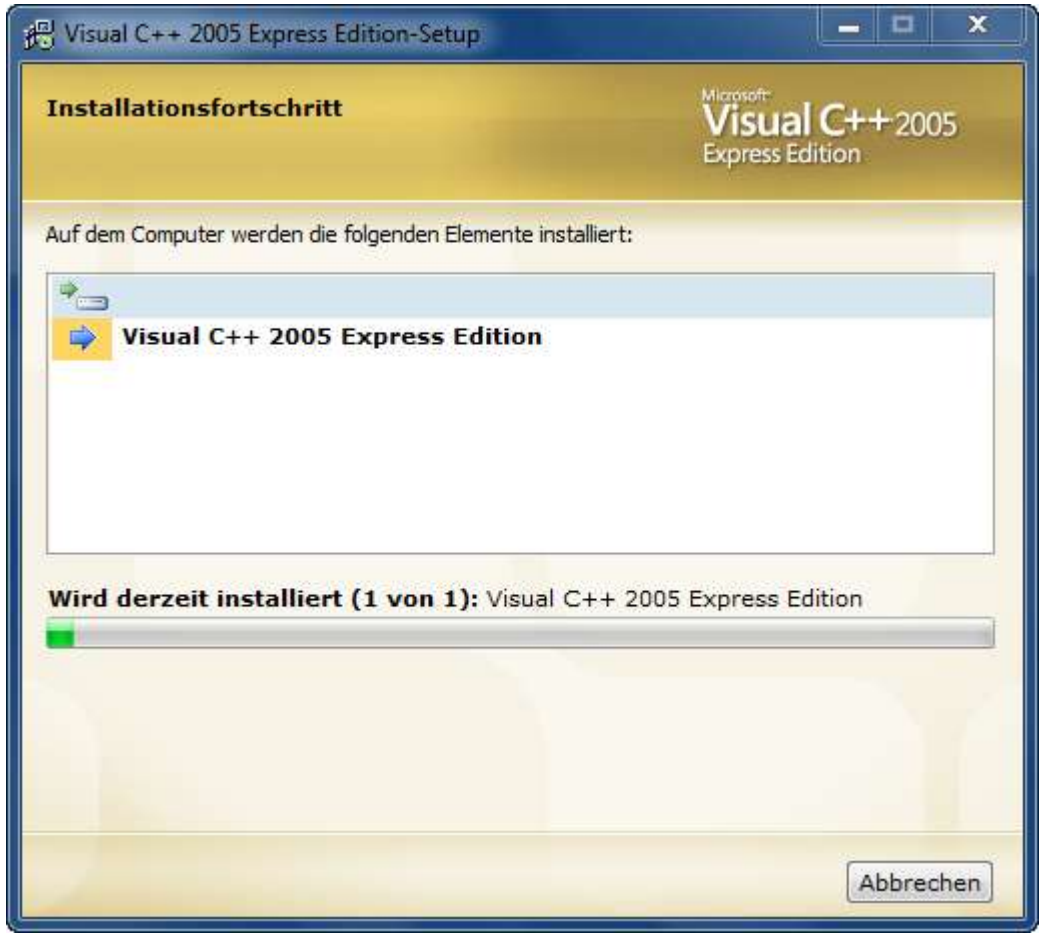
# Visual Studio 2005 Express C++

IDE nur für Code Erstellung erforderlich !

# Visual Studio 2005 Express C++

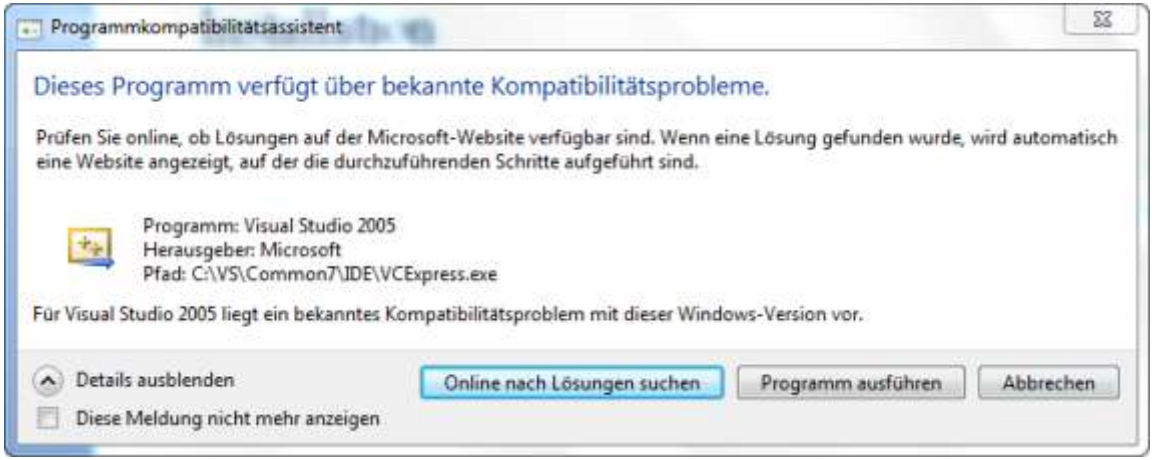
Zielordner für Installation

# Visual Studio 2005 Express C++



# Visual Studio 2005 Express C++

Bekanntes Kompatibilitätsproblem unter Windows 7
Weiter mit „Programm ausführen“
Windows 7 Patch wird später installiert



# Setup

- Visual Studio 2005 Express C++

- **Servicepack SP1 (KB926748)**
  - *http://www.microsoft.com/downloads/details.aspx?FamilyId=7B0B0339-613A-46E6-AB4D-080D4D4A8C4E&displaylang=de*

- **Patch für Vista / Windows 7 (KB932234)**
  - *http://www.microsoft.com/downloads/details.aspx?displaylang=de&FamilyID=90E2942D-3AD1-4873-A2EE-4ACC0AACE5B6*

- Platform SDK ( Windows Server 2003 SP1 )

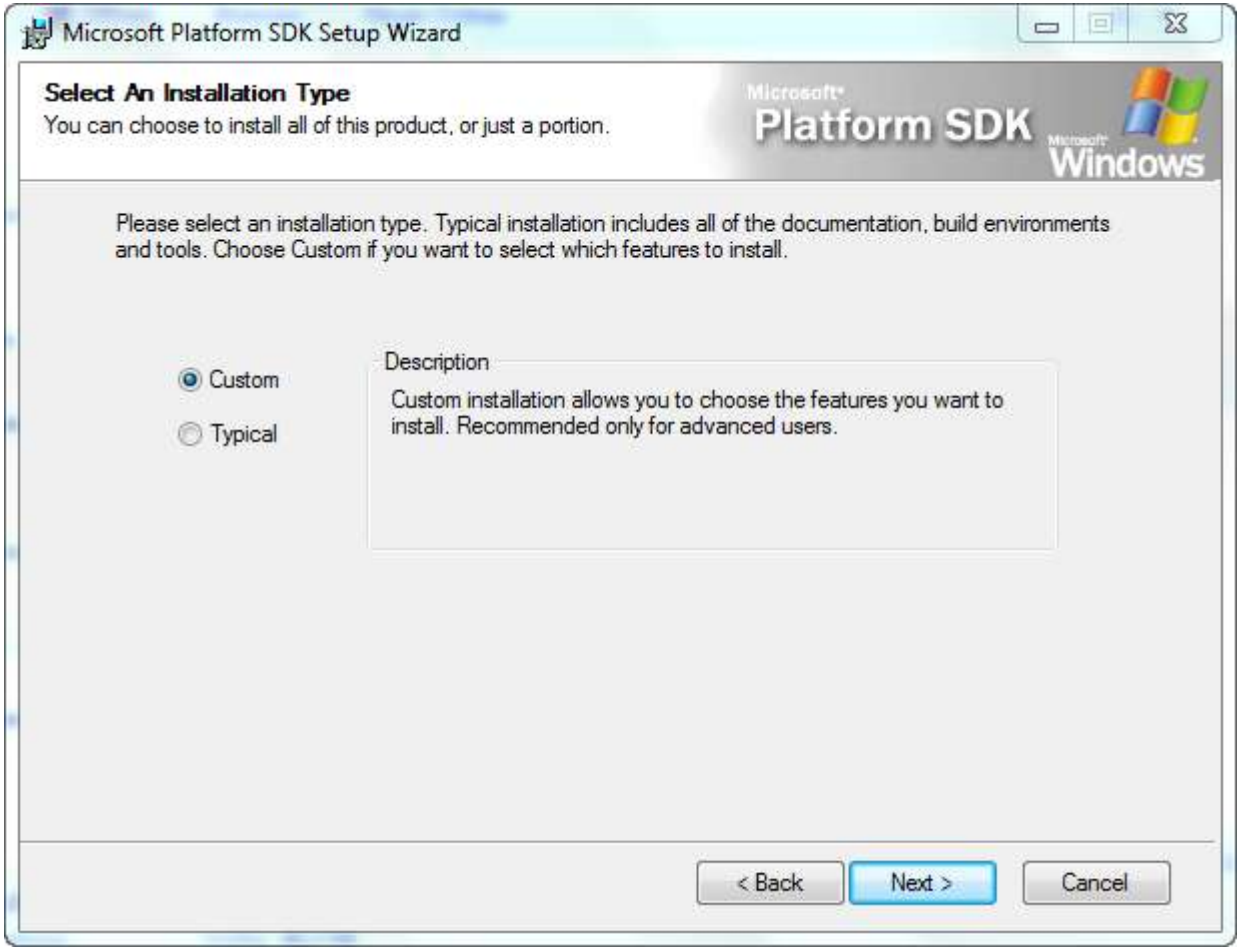
- SDK in Compiler anmelden

# Setup

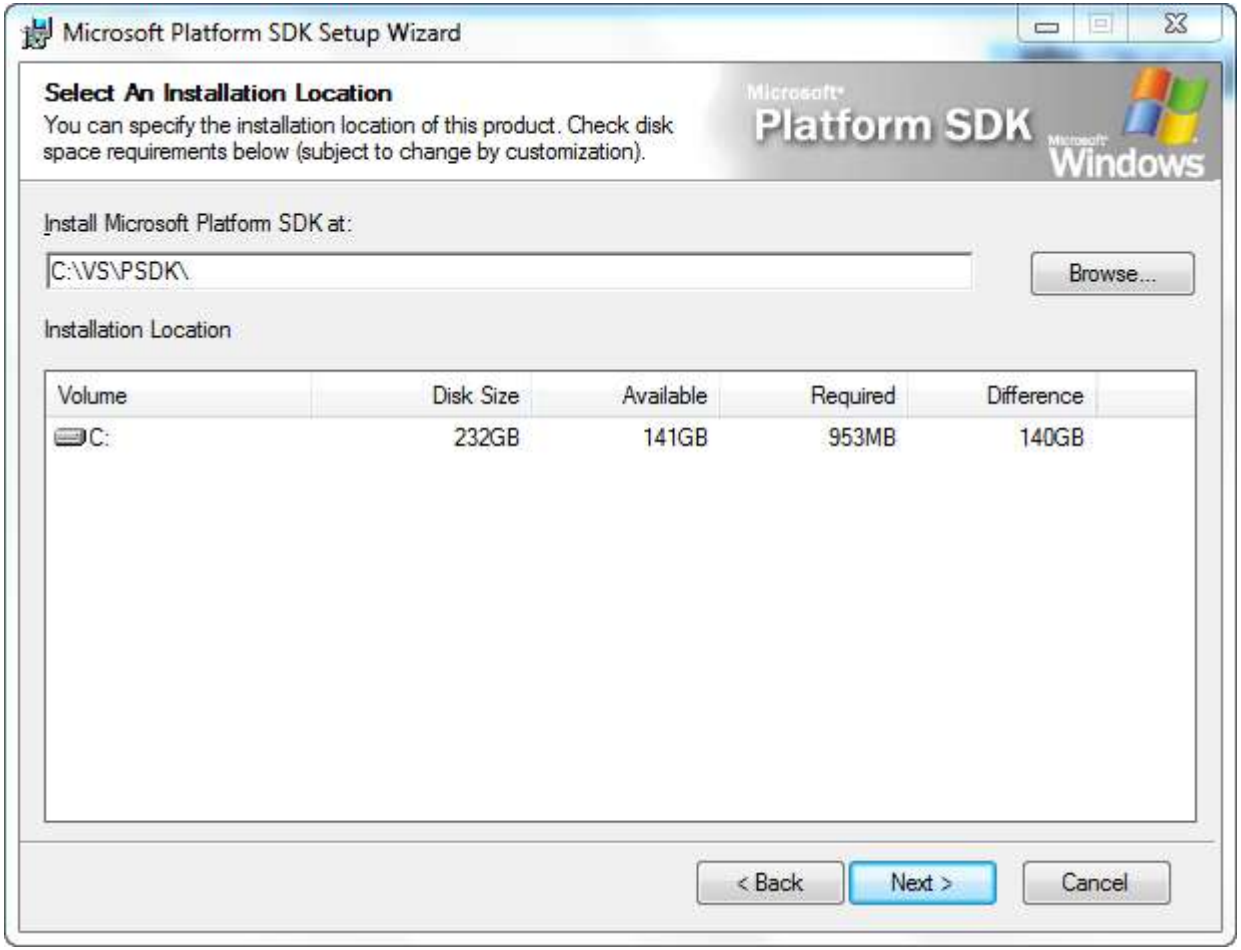
- Visual Studio 2005 Express C++
- Servicepack SP1 (KB926748)
- Patch für Vista / Windows 7 (KB932234)
- **Platform SDK ( Windows Server 2003 SP1 )**

  *http://www.microsoft.com/downloads/details.aspx?FamilyId=A55B6B43-E24F-4EA3-A93E-40C0EC4F68E5&displaylang=en*
- SDK in Compiler anmelden

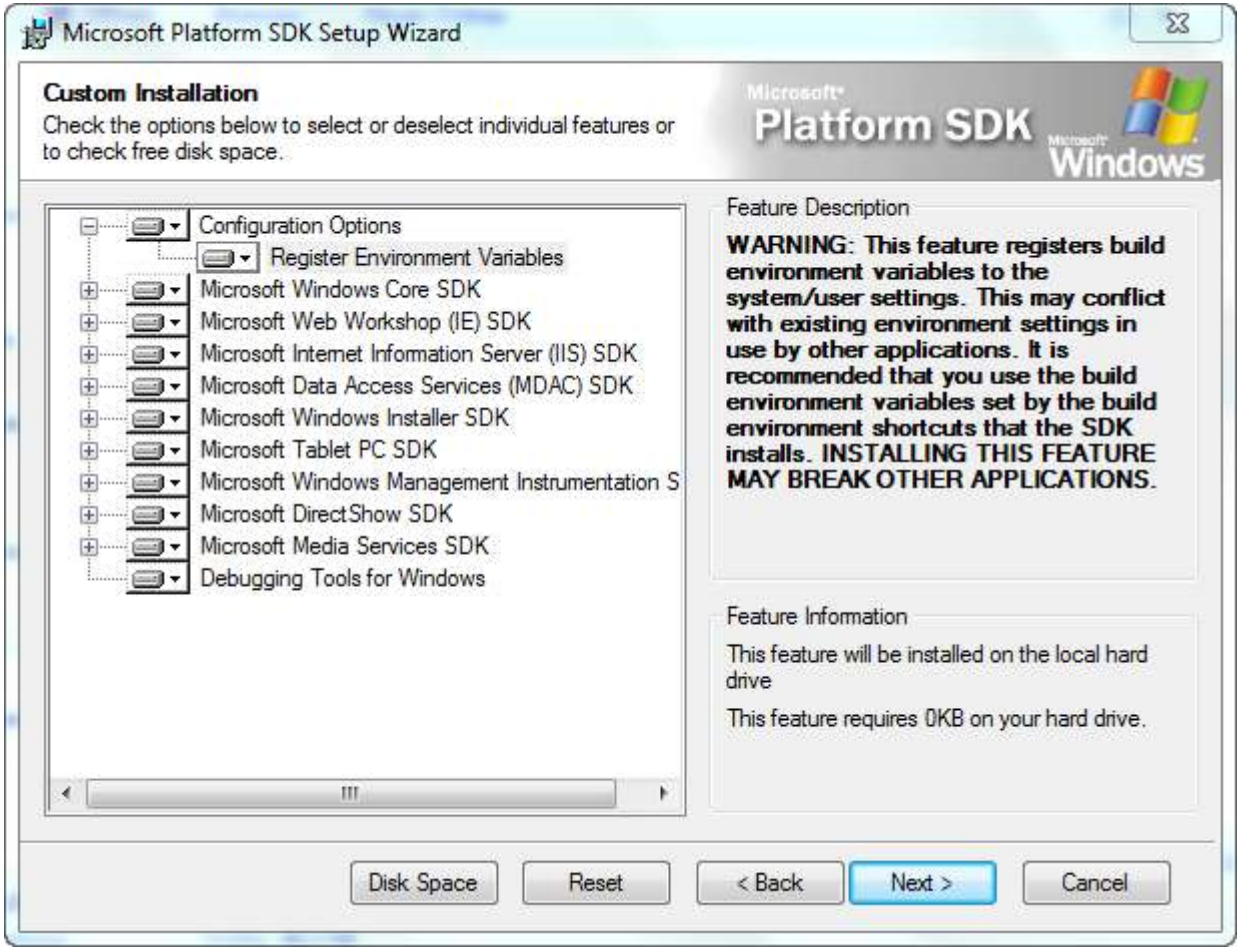
# Platform SDK



# Platform SDK



# Platform SDK

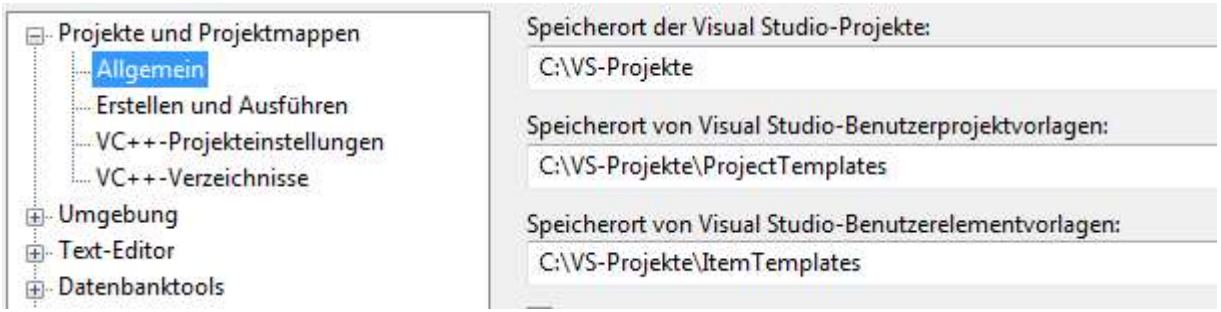


# Setup

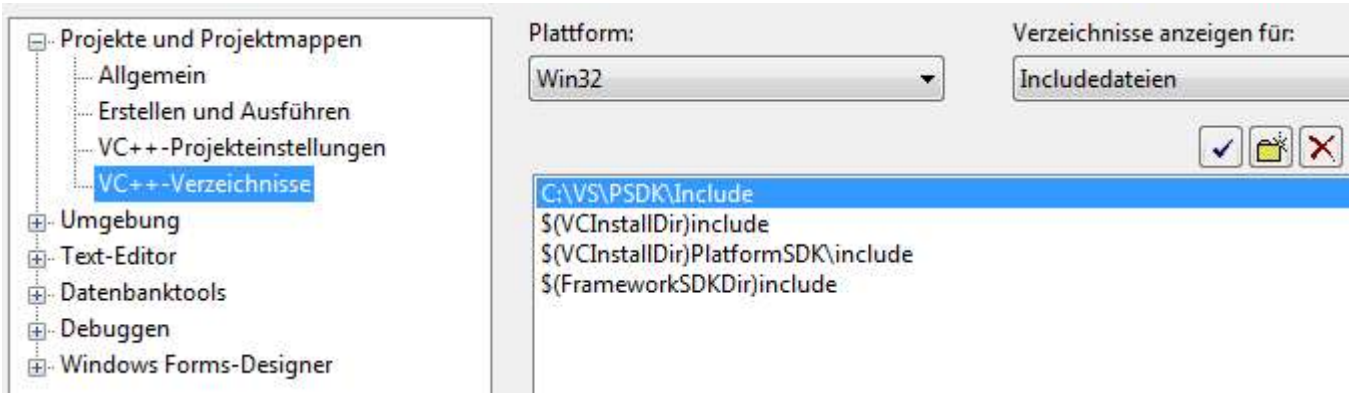
- Visual Studio 2005 Express C++
- Servicepack SP1 (KB926748)
- Patch für Vista / Windows 7 (KB932234)
- Platform SDK ( Windows Server 2003 SP1 )
- **SDK in Compiler anmelden**

# Konfiguration

•Optionen → Projekte und Projektmappen → Allgemein

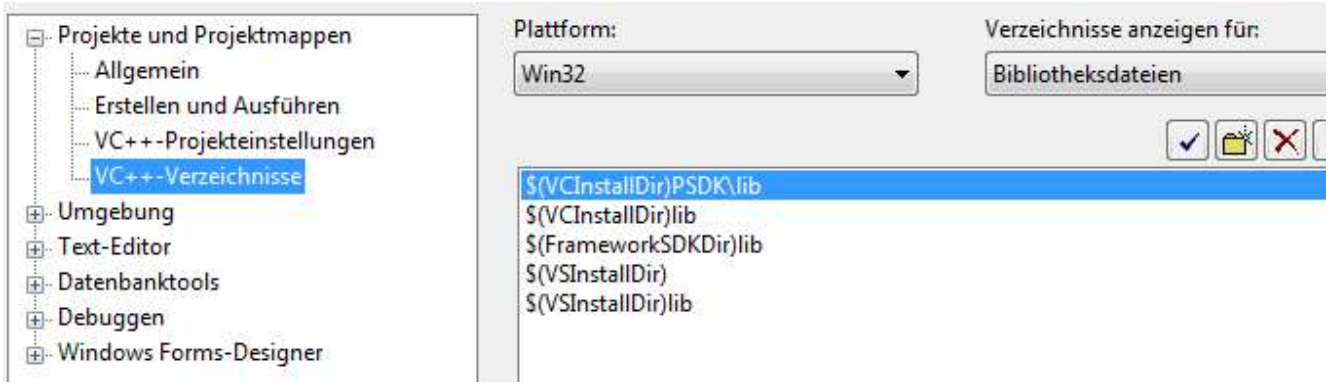



•Optionen → Projekte und Projektmappen → VC++ Verzeichnisse → Includes



# Konfiguration

- Optionen → Projekte und Projektmappen → VC++ Verzeichnisse → LIB



# Setup (Linux)

# Installation (Linux)

- Linux installieren (Linux SLES 10.0)
  - + GNU Compiler Collection (gcc) version 4.1.0
  - + Lotus Domino 8.0 ( muss nicht konfiguriert sein)
- Notes C-API unterhalb von /opt/ibm/lotus = /opt/ibm/lotus/notesapi
- Notes CPP unterhalb von /opt/ibm/lotus = /opt/ibm/lotus/notescpp

```
ec:/opt/ibm/lotus # ls -al
total 28
drwxr-xr-x 6 root root 4096 Nov 21 06:47 .
drwxr-xr-x 3 root root 4096 Nov 20 17:40 ..
-rw-r--r-- 1 root root  608 Nov 20 17:52 .install.dat
drwxr-xr-x 3 root root 4096 Nov 20 17:52 bin
drwxr-xr-x 3 root root 4096 Nov 20 17:51 notes
drwxrwxr-x 8 root root 4096 Feb 24  2009 notesapi
drwxr-xr-x 7 root root 4096 Nov 21 06:47 notescpp
```

# /etc/profile

```
# -------------- notesapi  --------------
LOTUS=/opt/ibm/lotus
export LOTUS
#
Notes_ExecDirectory=/opt/ibm/lotus/notes/latest/linux
export Notes_ExecDirectory
#
NOTES_DATA_DIR=/local/notesdata
export NOTES_DATA_DIR
#
DOMINO_RES_DIR=/opt/ibm/lotus/notes/latest/linux/res/C
export DOMINO_RES_DIR
#
PATH=$PATH:$Notes_ExecDirectory:$NOTES_DATA_DIR:$DOMINO_RES_DIR
export PATH
# -------------- notescpp  --------------
CPPAPI_ID=80
export CPPAPI_ID
#
LD_LIBRARY_PATH=$LOTUS/notescpp/lib/linux
export LD_LIBRARY_PATH
```

# Symbolische Links

**Lotus Notes C API**

```
ln -s /opt/ibm/lotus/notes/latest/linux/libnotes.so /usr/lib/libnotes.so
ln -s /opt/ibm/lotus/notes/latest/linux/libndgts.so /usr/lib/libndgts.so
ln -s /opt/ibm/lotus/notes/latest/linux/libxmlproc.so /usr/lib/libxmlproc.so
```

**Lotus Notes C++ API**

```
cp libcpplin.so.80 $Notes_ExecDirectory/
ln -s /opt/ibm/lotus/notes/latest/linux/libcpplin.so.80 /usr/lib/libcpplin.so.80
```

# Installation (Test)

- cd $INTRO
- make –f linux.mak
- chown / chgrp / chmod
- su notes
- ./intro log.ntf

```
ec:/opt/ibm/lotus/notesapi/samples/basic/intro # ./intro log.ntf
Error Code 493: Do not run as root.

Error initializing Notes.
ec:/opt/ibm/lotus/notesapi/samples/basic/intro # su notes
notes@ec:/opt/ibm/lotus/notesapi/samples/basic/intro> ./intro pubnames.ntf


Lotus C API for Notes/Domino 8.5.0 Sample Application


The title for the database, pubnames.ntf, is:

Domino Directory (8.5.1 Server)
```

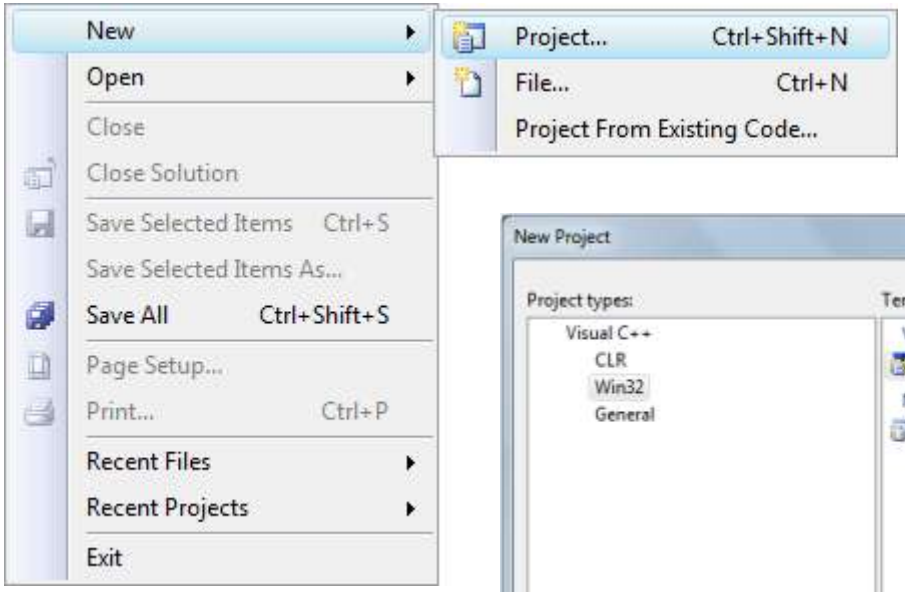
# Makefile

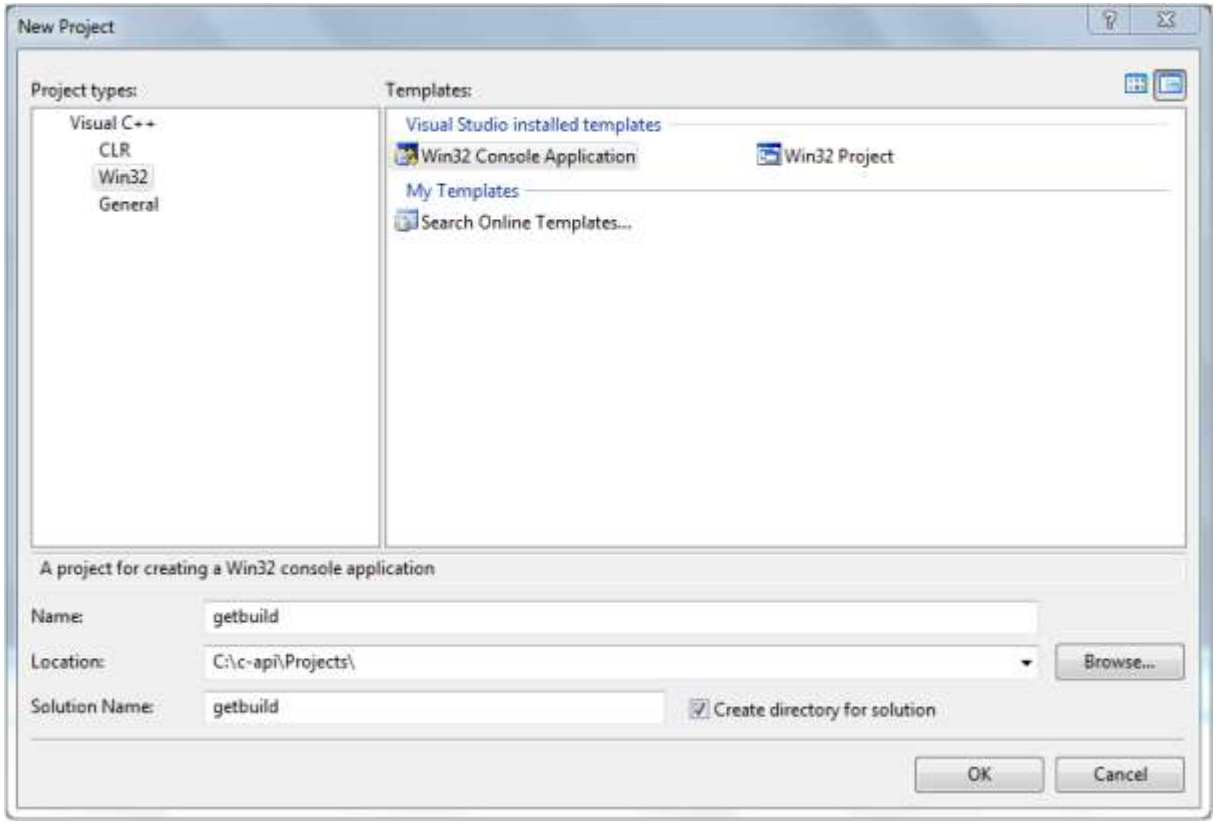
- Steuerung
  - Compiler
  - Linker
- „make“ verwendet makefile

```
# set TARGET to the name of the executable to create
TARGET = libmailscan.so
# set SOURCES to the list of C source files in this program
SOURCES = mailscan.c log.c
# set HEADERS to the list of C include files in this program
HEADERS =
# set OBJECTS to the list of object files that must be linked
OBJECTS = mailscan.o log.o
# CC defines the compiler.
CC = g++
# Set CCOPTS - the compiler options.
CCOPTS = -c -march=i486
# You may use -g flag for debugging:
#CCOPTS = -c -march=i486 -g
# set NOTESDIR to specify where to search for the Notes library file
NOTESDIR = /opt/lotus/notes/latest/linux
# Set LINKOPTS - the linker options passed to CC when linking.
# -o $(TARGET) causes compiler to create target rather than a.out
LINKOPTS = -o $(TARGET) -shared -fpic -Wl,-R$(NOTESDIR),-znodefs,-ztext
# Notes API header files require UNIX to be defined.
DEFINES =  -DUNIX -DLINUX -DHANDLE_IS_32BITS
# set INCDIR to specify where to search for include files
INCDIR = /opt/lotus/notesapi/include
# set LIBS to list all the libraries ld should link with.
LIBS = -lnotes -lm -lnsl -lpthread -lc -lresolv -ldl
# the executable depends on the objects.
$(TARGET): $(OBJECTS)
    $(CC) $(LINKOPTS) $(OBJECTS) -L$(NOTESDIR) $(LIBS)
# the object files depend on the corresponding source files
.c.o:
    $(CC) $(CCOPTS) $(DEFINES) -I$(INCDIR) $(SOURCES)
clean :
    rm -rf mailscan.o
```

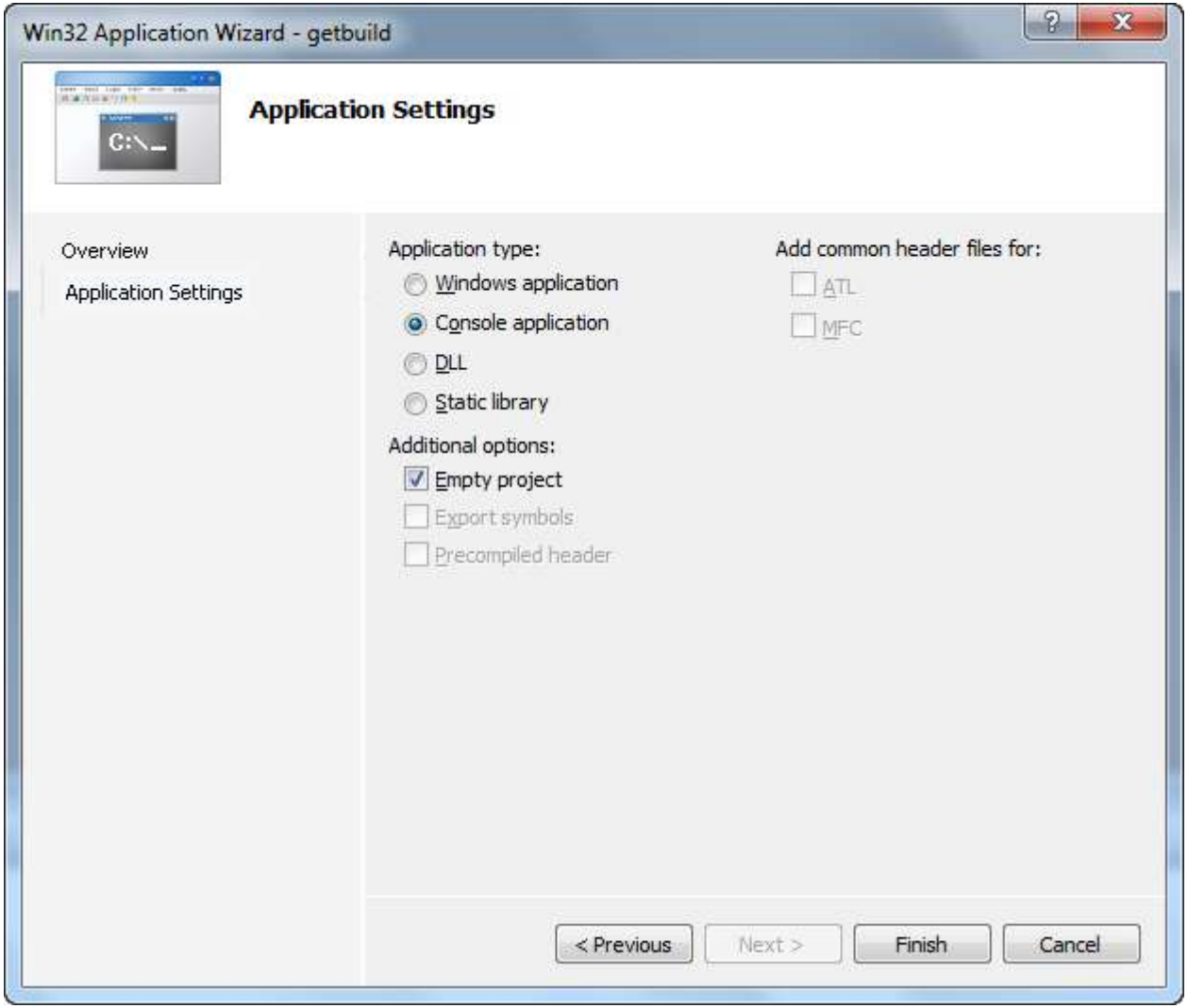
# Erstellen von Anwendungen

# Neues Projekt

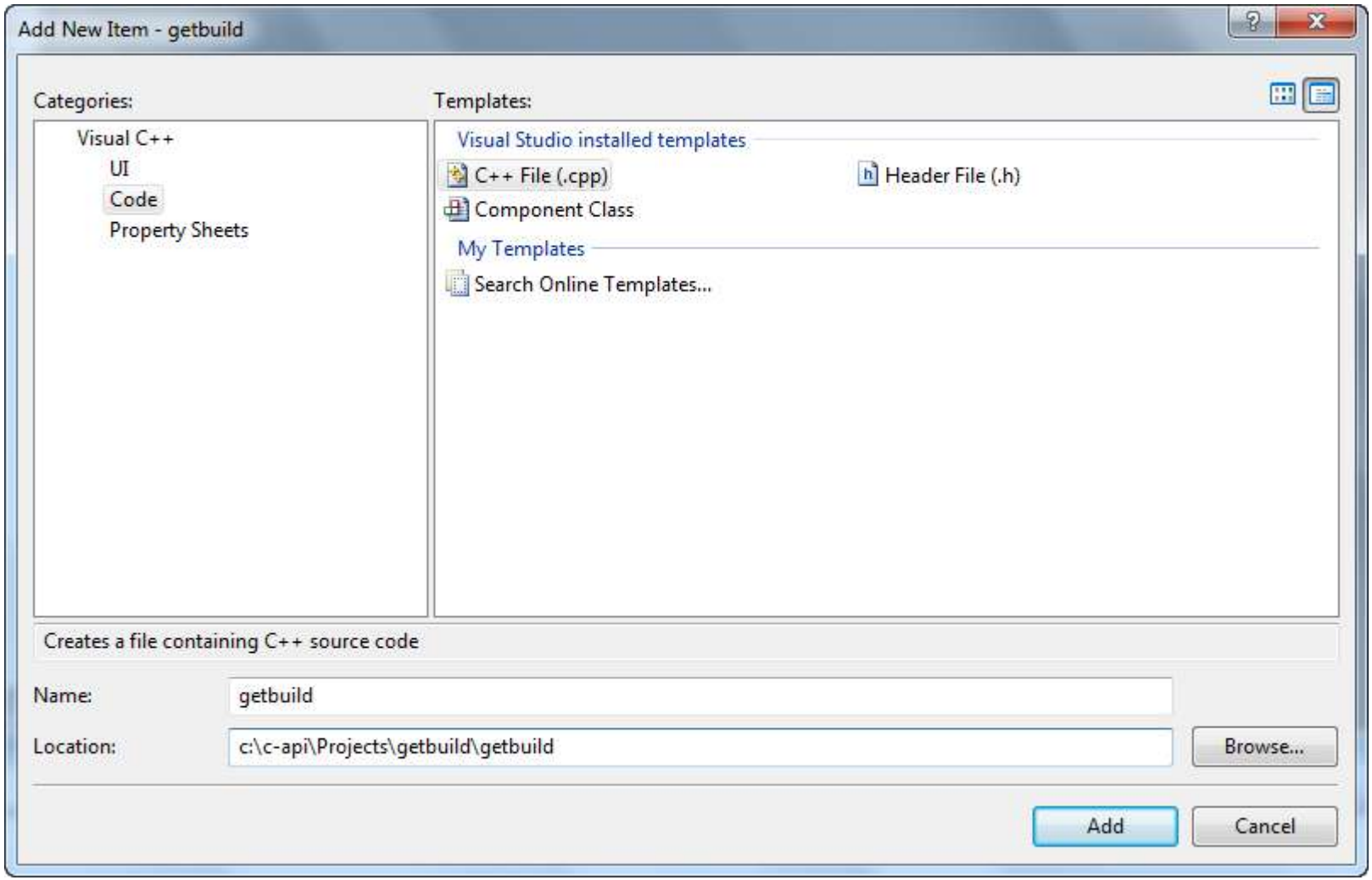


# Neues Projekt



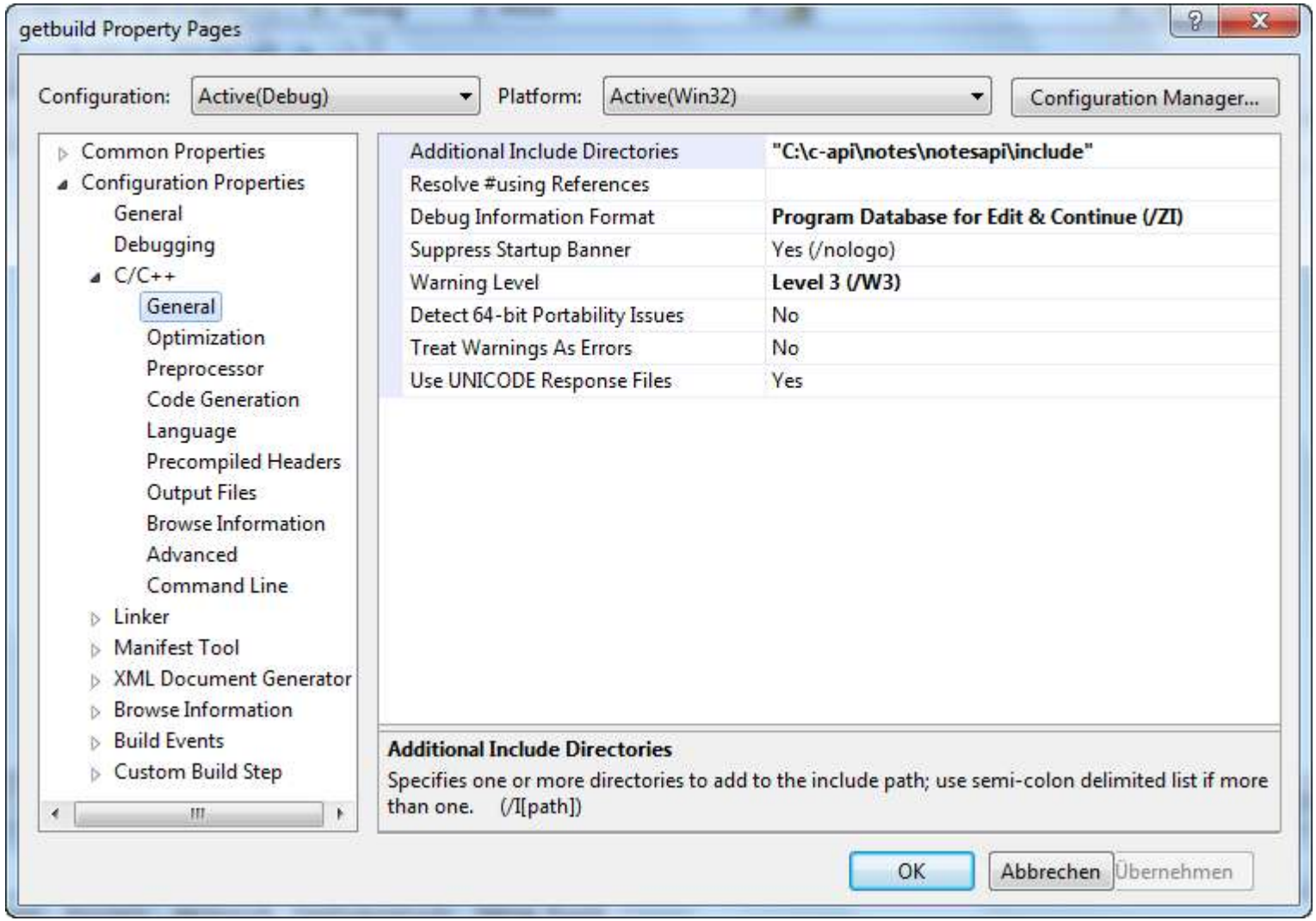
# Neues Projekt



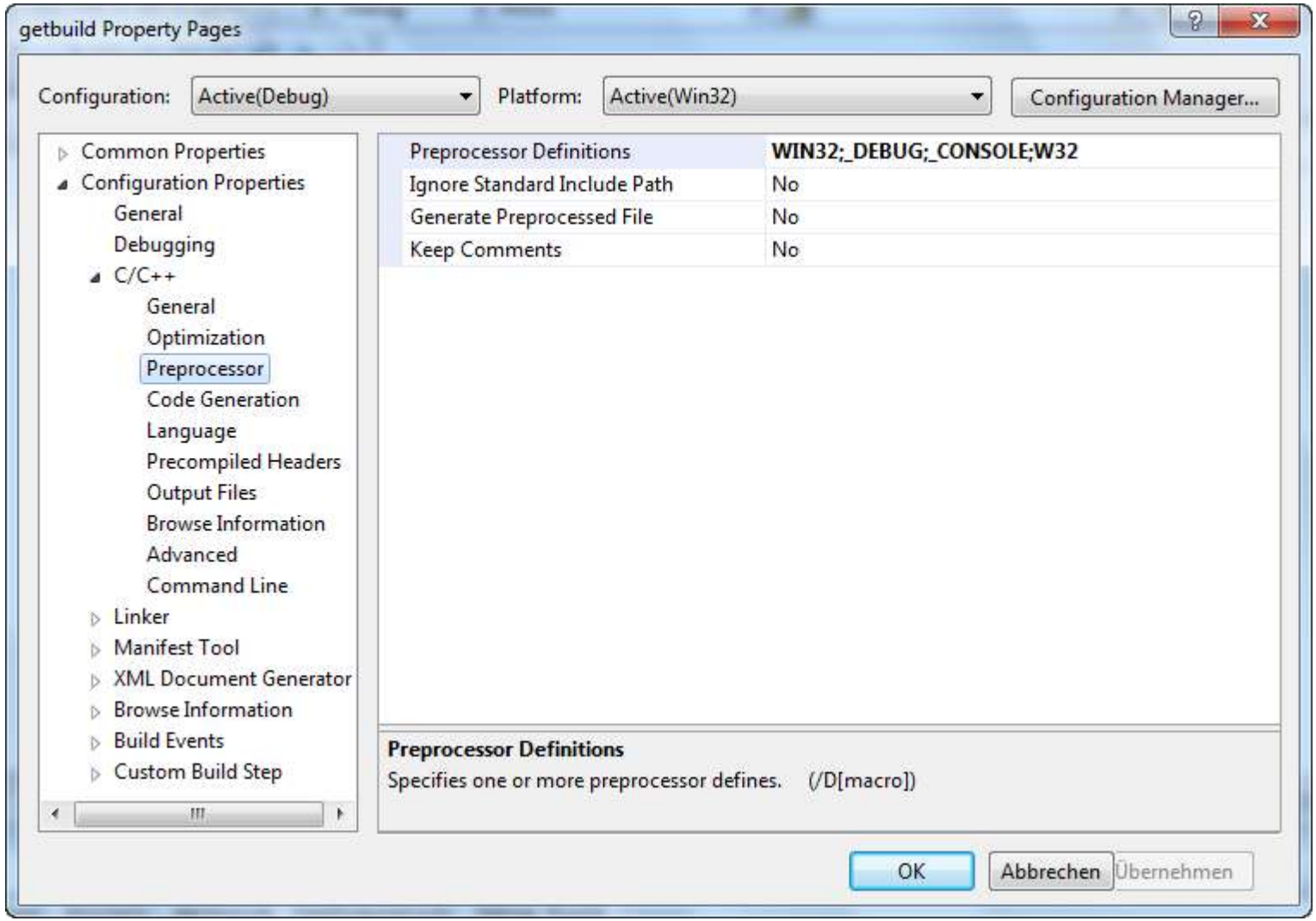
# Source Code

```
#include <stdio.h>
#include <string.h>
#include <global.h>
#include <nsfdb.h>

int main(int argc, char *argv[])
{   char db_filename[]="names.nsf";
    DBHANDLE db_handle;
    char buffer[NSF_INFO_SIZE] = "";
    char title[NSF_INFO_SIZE] = "";
    STATUS error = NOERROR;

    if (error = NotesInitExtended (argc, argv))
    {   printf("\n Unable to initialize Notes.\n");
        return (1); }

    if (error = NSFDbOpen (db_filename, &db_handle))
    {   NotesTerm();
        return (1); }

    if (error = NSFDbInfoGet (db_handle, buffer))
    {   NSFDbClose (db_handle);
        return (1); }

    NSFDbInfoParse (buffer, INFOPARSE_TITLE, title, NSF_INFO_SIZE - 1);
    printf ("\n\nDBTitle: %s\n\n", title);

    if (error = NSFDbClose (db_handle))
        return (1);

    NotesTerm();
    return (0);
}
```

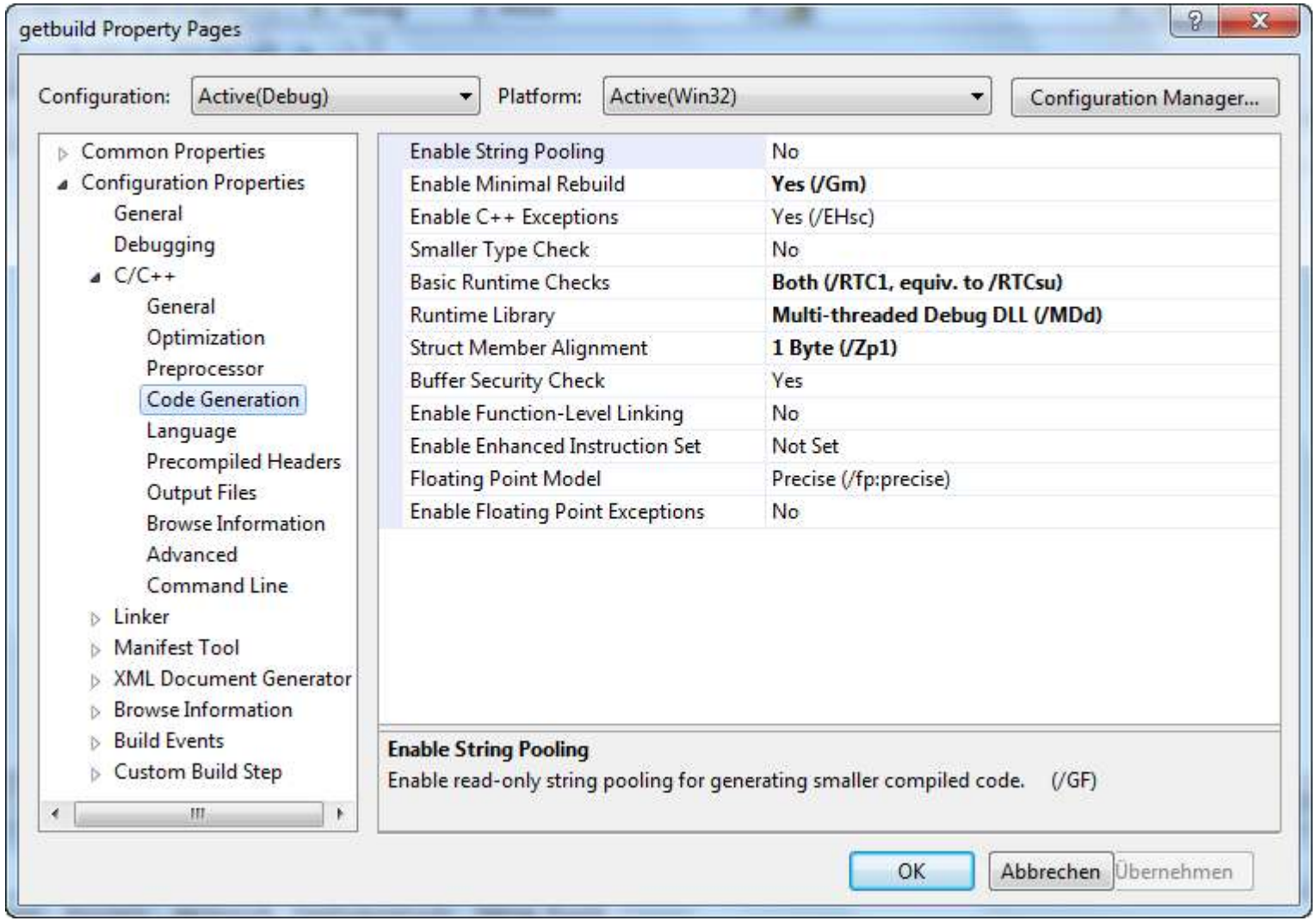
# Projekt Properties (C / C++)



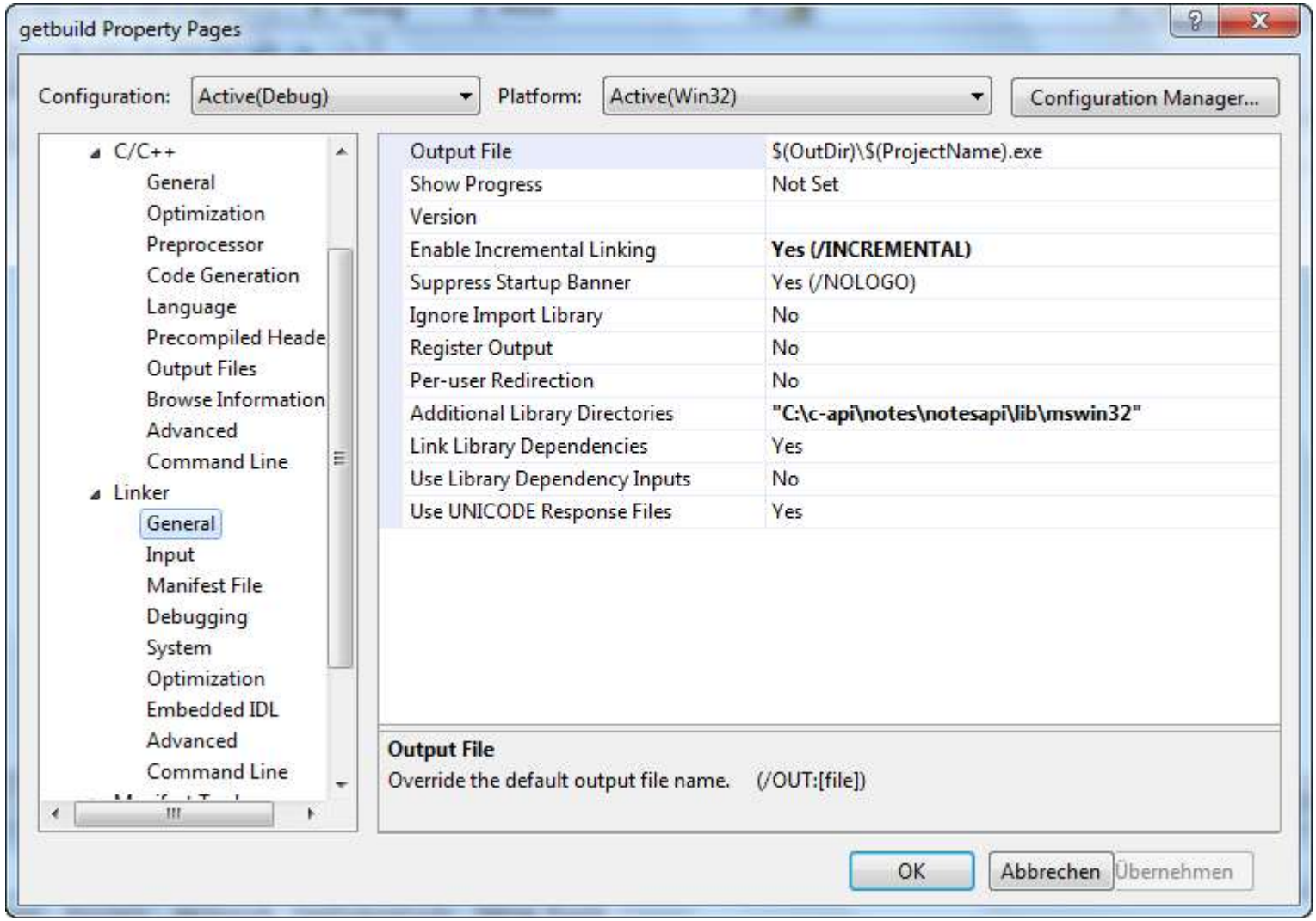
# Projekt Properties (C / C++)



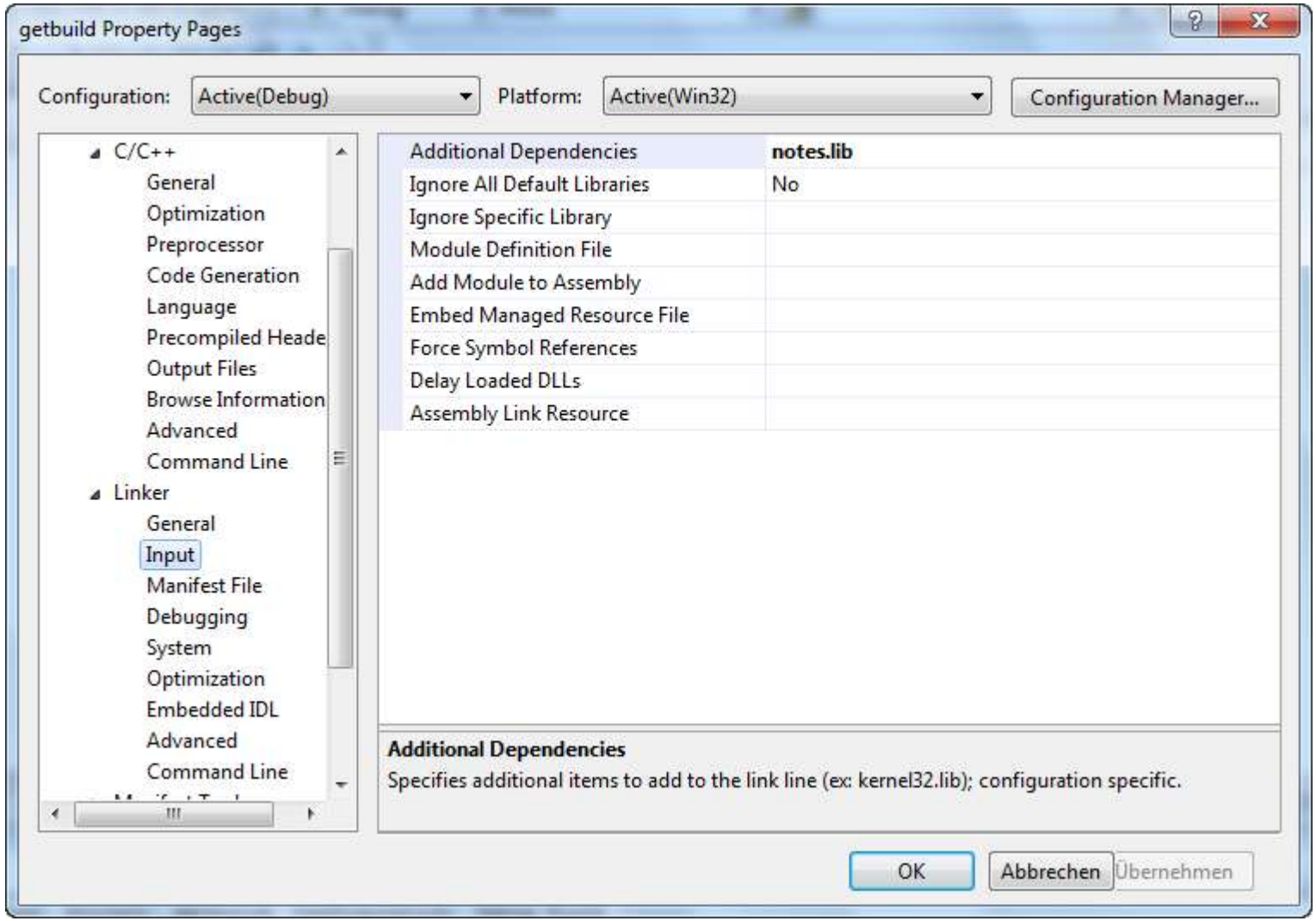
# Projekt Properties (C / C++)



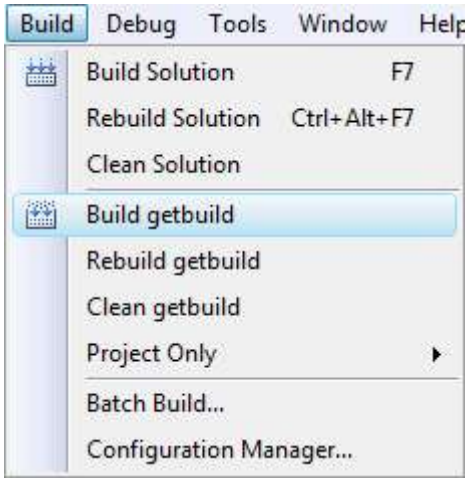
# Projekt Properties (Linker)

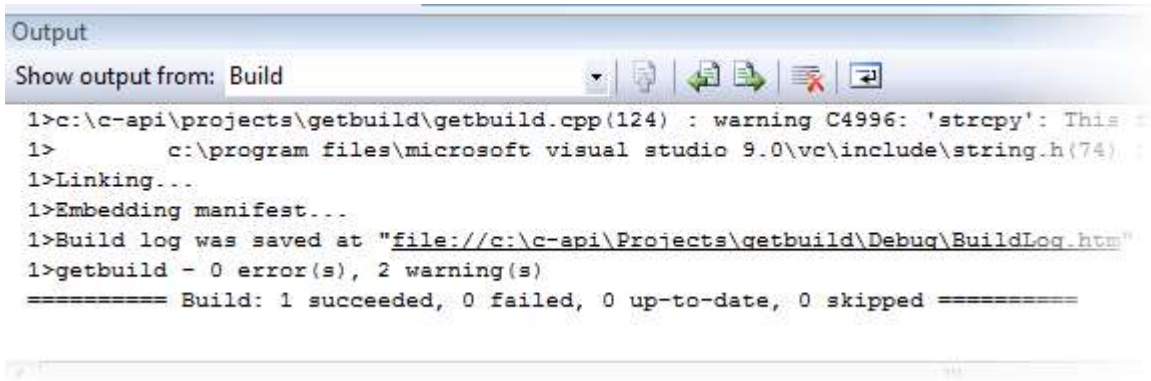


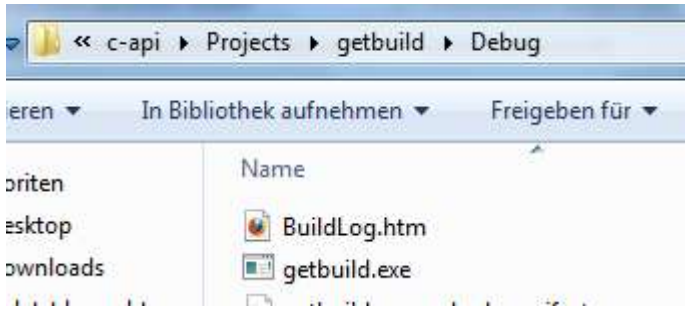
# Projekt Properties (Linker)



# Build Solution

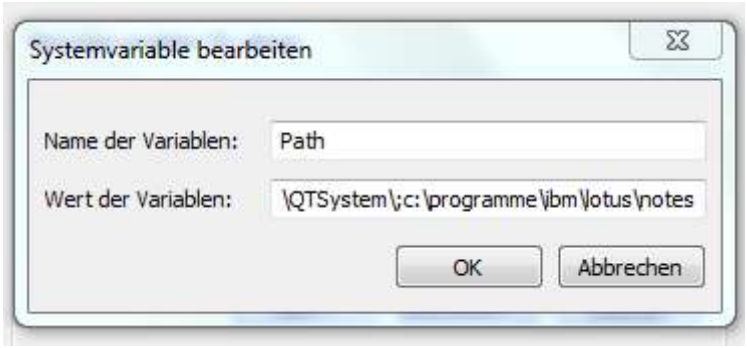


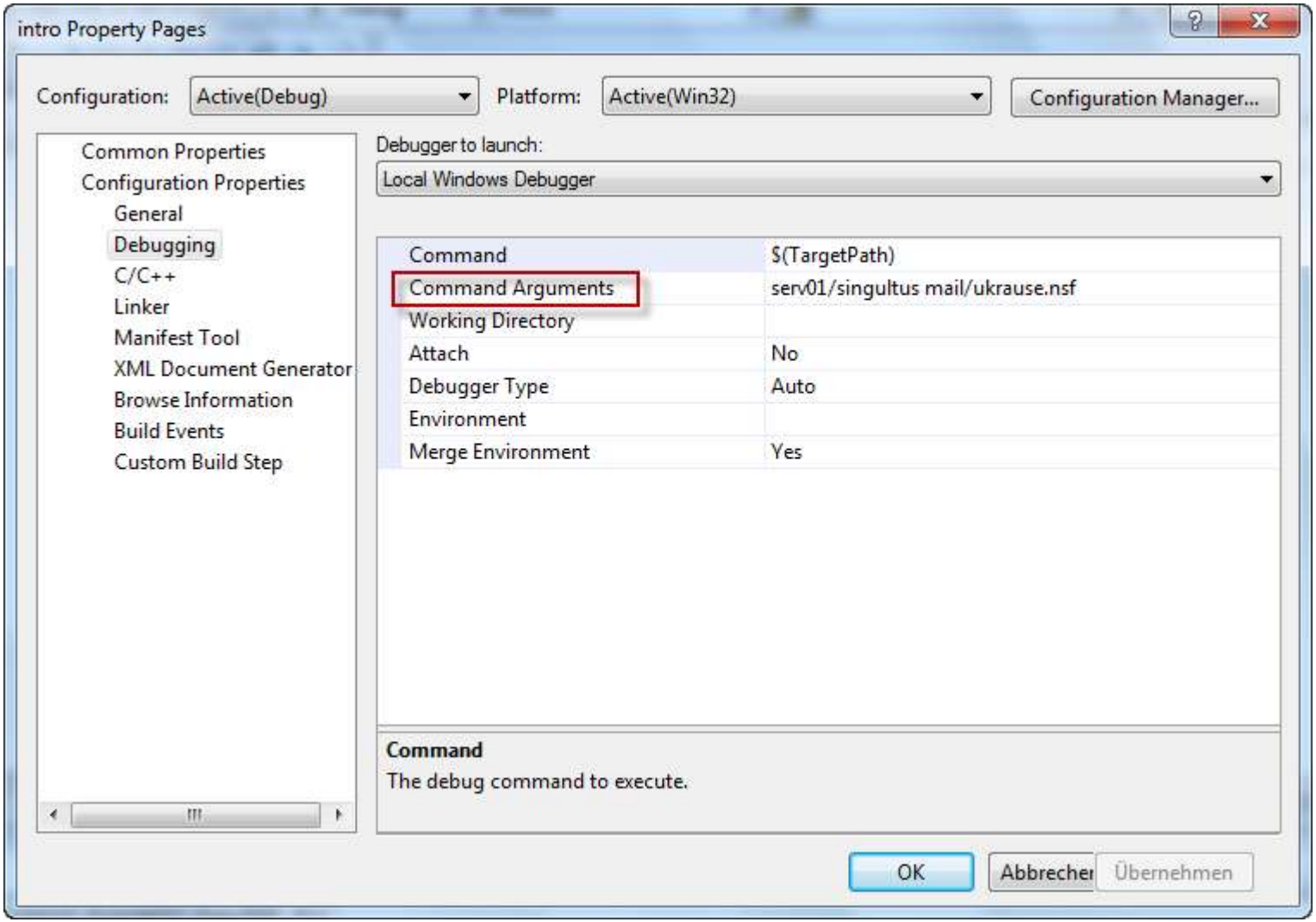



# Debug

# Pfad erweitern



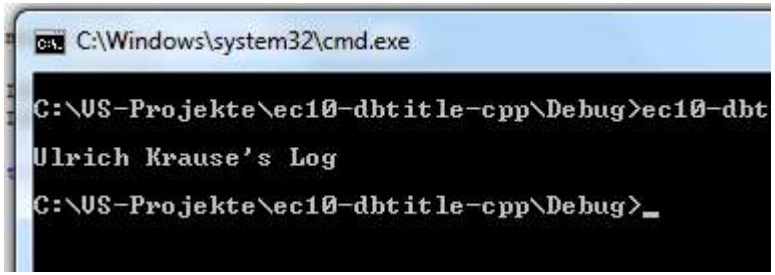
# Parameter vorbelegen



# Jetzt mit C++

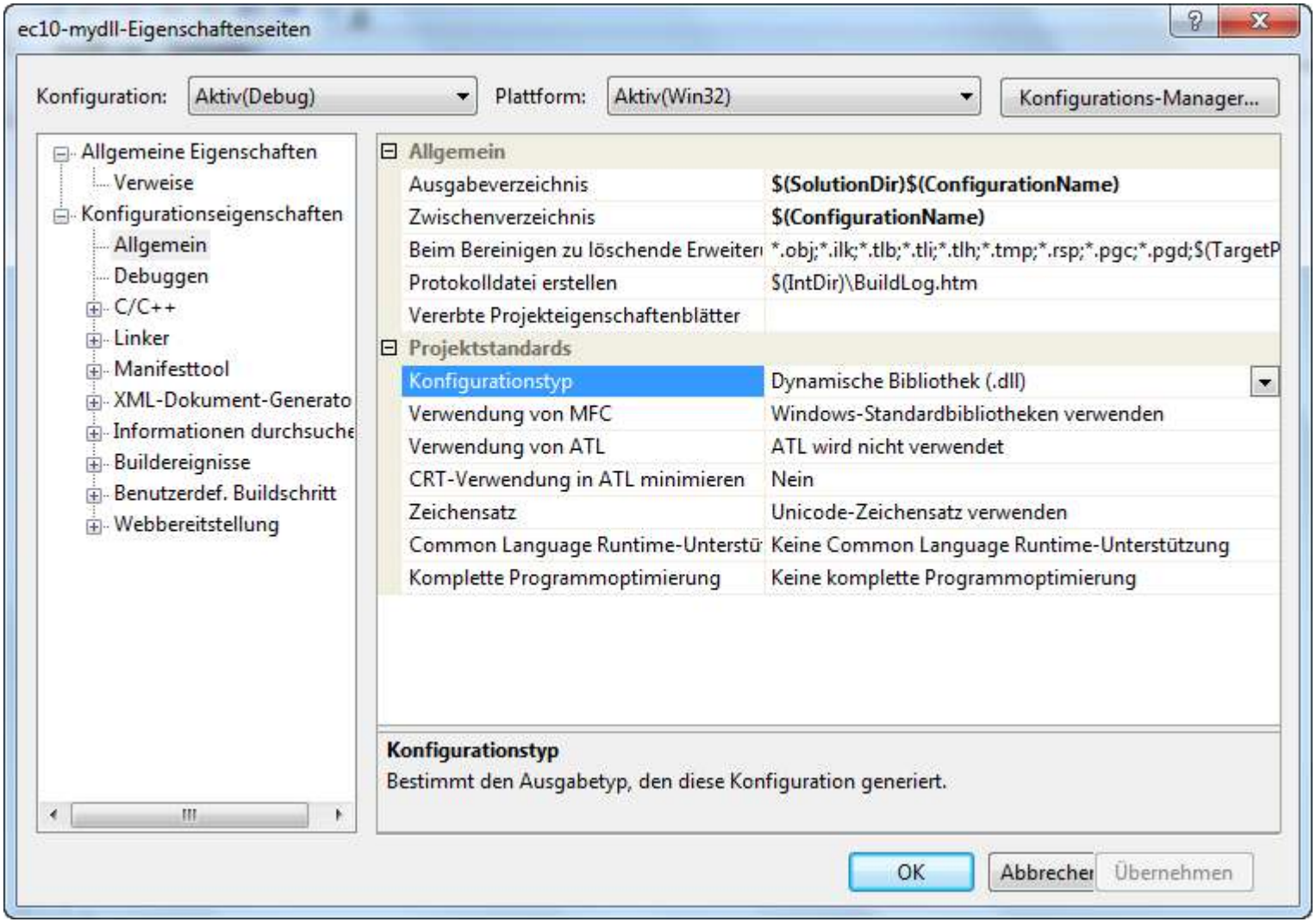
# C++

```
#include <stdio.h>
#include <lncppapi.h>

int main(int argc, char *argv[]){

    LNSetThrowAllErrors(true);

    LNNotesSession session;
    LNDatabase db;
    LNString dbtitle;

    try {
        session.Init();

        session.GetDatabase("log.nsf", &db);
        db.Open();
        dbtitle = db.GetTitle();

        printf("\n");
        printf(dbtitle);
        printf("\n");
    }

    catch (LNSTATUS error)  {
        printf("\nERROR\n");
        return 1;
    }

        db.Close();
        session.Term();
        return NOERROR;
}
```



# Neues Projekt - DLL

# Neues Projekt - DLL

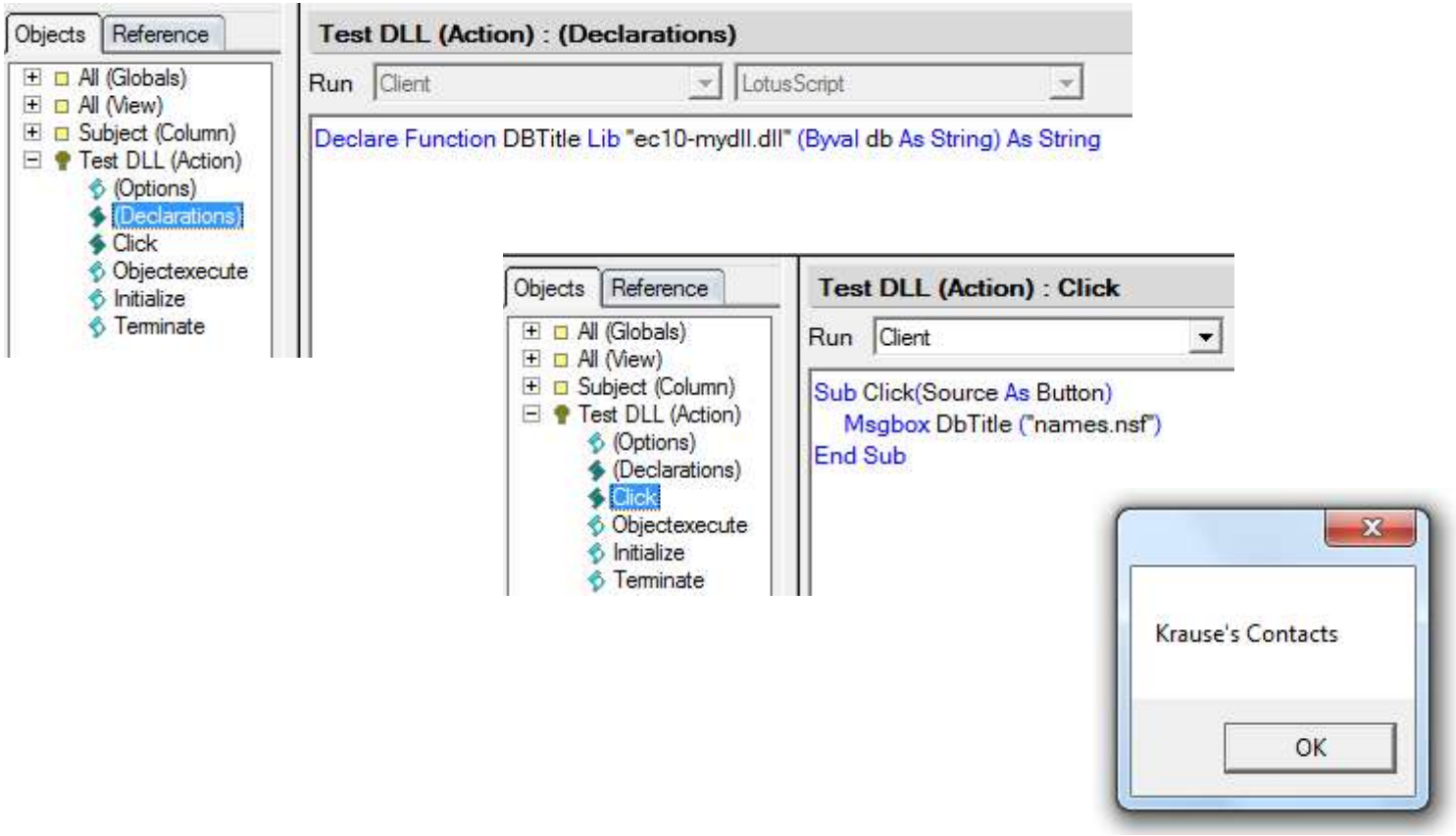


# DLL – Source Code

```
#include <string.h>
#include <global.h>
#include <nsfdb.h>
#define DllExport   __declspec( dllexport )

extern "C"
{   extern DllExport char* DBTitle(char *db_filename);

    char* DBTitle(char* db_filename)
    {
        DBHANDLE    db_handle;
        STATUS      error = NOERROR;

        static char     buffer[NSF_INFO_SIZE] = "";
        static char     title[NSF_INFO_SIZE] = "";

        if (error = NSFDbOpen (db_filename, &db_handle))
        {
            return title;   // empty string
        }

        NSFDbInfoGet (db_handle, buffer);
        NSFDbInfoParse (buffer, INFOPARSE_TITLE, title, NSF_INFO_SIZE - 1);
        NSFDbClose (db_handle);

        return title;
    }
}
```

# DLL – Aufruf aus LotusScript

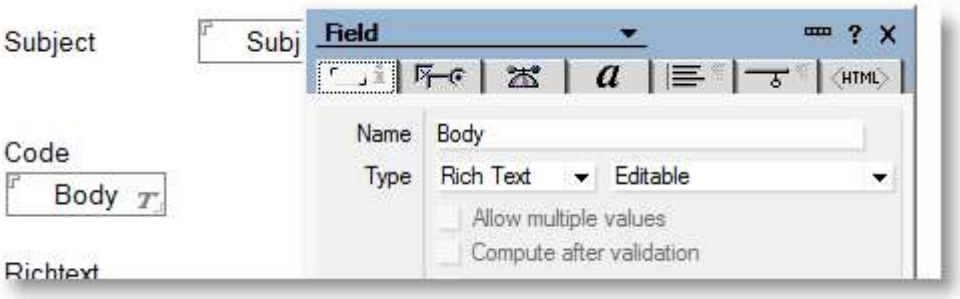
- DLL ins Notes Verzeichnis kopieren
- Deklaration der Funktionen / Subs
- Aufruf der Funktion / Sub



# C-API und Rich Text

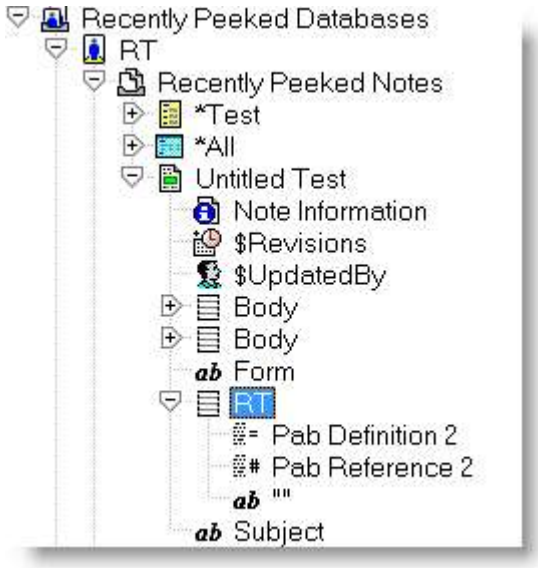
# C-API und Rich Text

- Container für unstrukturierte Daten
  - Formatierter Text
  - Grafik
  - OLE
  - Hotspots
  - Doclinks
  - …




- Mehrere Felder mit gleichem Namen möglich

- Compound Document Records (CD Records)
  - Feste Struktur ( CD Record Type ) + variable Daten
  - Über 200 unterschiedliche CD Record Types
    - *C API Reference* → *Ansicht: DataTypes\ODS*




- Rich Text ist niemals „leer“

# C-API und Rich Text

Frage im atnotes.de Forum:

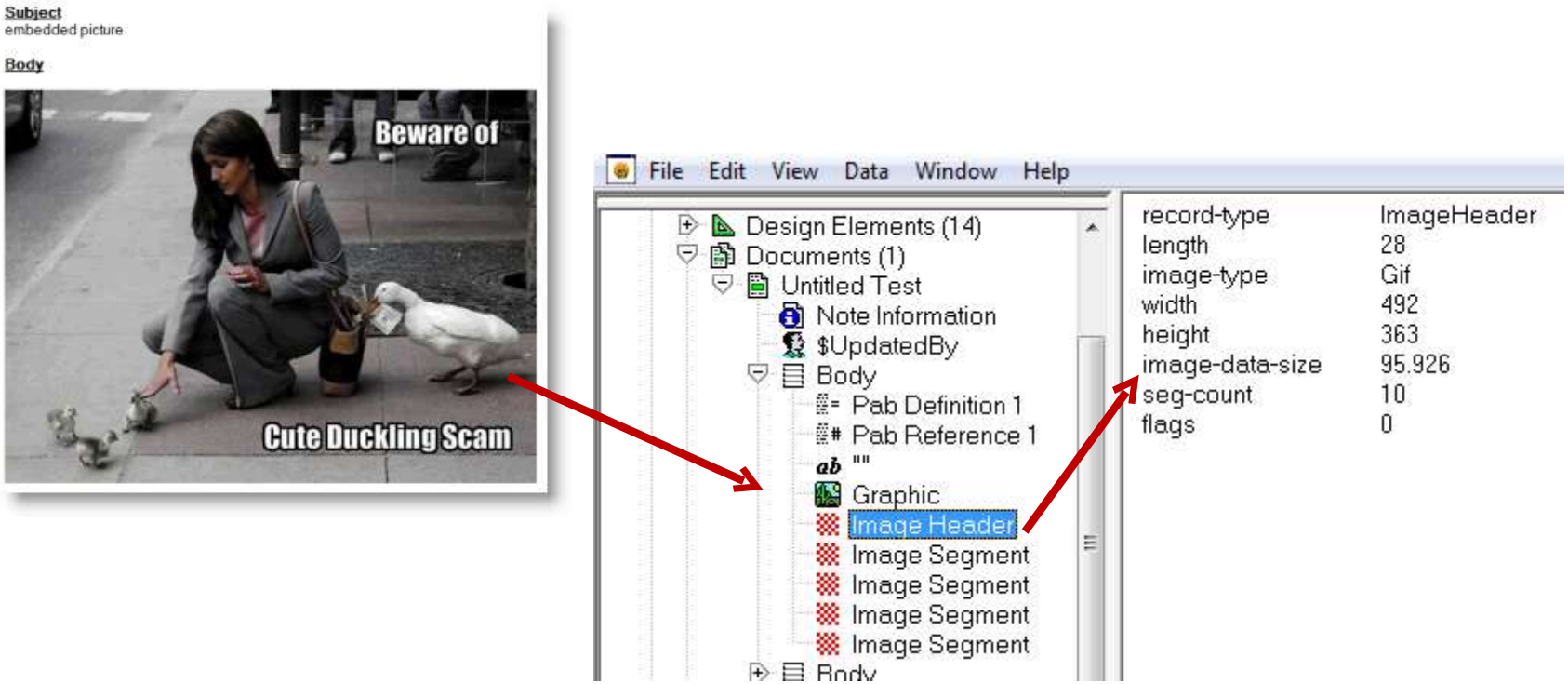
*„Kennt ihr zufällig eine Möglichkeit mir der ich die Größe von eingebetteten Bildern in einem RTF ermitteln kann?“*


- @Formula ?
  - es gibt keine Formel
- LotusScript?
  - es gibt keine Methode oder Eigenschaft

➔ Dann bauen wir uns eine solche Funktion einfach !!

# C-API und Rich Text

Analyse mit Notes Peek



# C-API und Rich Text

## Benötigte Funktion ( DIE zentrale Funktion bei Rich Text Aufgaben )

```
#include <ods.h>

STATUS LNPUBLIC EnumCompositeBuffer(
        BLOCKID  ItemValue,
        DWORD  ItemValueLength,
        ActionRoutinePtr  ActionRoutine,
        void far *vContext);
```

## Benötigte Struktur

```
#include <editods.h>

typedef struct {
        LSIG  Header;               /* Signature and Length */
        WORD  ImageType;            /* Type of image (e.g., GIF, JPEG) */
        WORD  Width;                /* Width of the image (in pixels) */
        WORD  Height;               /* Height of the image (in pixels) */
        DWORD ImageDataSize;        /* Size (in bytes) of the image data */
        DWORD SegCount;             /* Number of CDIMAGESEGMENT records expected to follow *
        DWORD Flags;                /* Flags (currently unused) */
        DWORD Reserved;             /* Reserved for future use */
} CDIMAGEHEADER;
```

# C-API und Rich Text

Schauen wir uns das einmal in der Entwicklungsumgebung an (DEMO)

```
void ProcessImageHeader(char *RecordPtr, DWORD RecordLength, WORD RecordType)
{
    char far *          p = RecordPtr;
    char ImgType[4];
    CDIMAGEHEADER       cdImageHeader;

    ODSReadMemory( &p, _CDIMAGEHEADER, &cdImageHeader, 1 );
    printf("\n");
    printf("Embedded Image Groesse: %d Bytes\n", cdImageHeader.ImageDataLen);
    printf("Embedded Image Breite: %d Pixel\n", cdImageHeader.Width);
    printf("Embedded Image Hoehe: %d Pixel\n", cdImageHeader.Height);

    switch (cdImageHeader.ImageType)

    {
    case CDIMAGETYPE_GIF:
         strcpy (ImgType,"GIF");
         break;

    case CDIMAGETYPE_JPEG:
         strcpy (ImgType,"JPG");
         break;

    case CDIMAGETYPE_BMP:
         strcpy (ImgType,"BMP");
         break;

    }

    printf("Embedded Image Type: %s \n", ImgType);
    return;
}
```

# C-API und Rich Text

Das Ergebnis

**Subject**
embedded picture

**Body**


```
void ProcessImageHeader(char *RecordPtr, DWORD RecordLength, WORD RecordType)
{
    char far *          p = RecordPtr;
    char ImgType[4];
    CDIMAGEHEADER       cdImgageHeader;

    ODSReadMemory( &p, _CDIMAGEHEADER, &cdImgageHeader, 1 );

    printf("\nEmbedded Image Groesse: %d Bytes\n", cdImgageHeader.ImageDataSi
    printf("Embedded Image Breite: %d Pixel\n", cdImgageHeader.Width);
    printf("Embedded Image Hoehe: %d Pixel\n", cdImgageHeader.Height);

    switch (cdImgageHeader.ImageType)
    {
    case CDIMAGETYPE_GIF:
        strcpy (ImgType,"GIF");
        break;
    case CDIMAGETYPE_JPEG:
        strcpy (ImgType,"JPG");
        break;
    case CDIMAGETYPE_BMP:
        strcpy (ImgType,"BMP");
        break;
    }
    printf("Embedded Image Type: %s
    re
```

```
C:\VS-Projekte\richtext\Debug>richtext

Embedded Image Groesse: 95926 Bytes
Embedded Image Breite: 492 Pixel
Embedded Image Hoehe: 363 Pixel
Embedded Image Type: GIF

C:\VS-Projekte\richtext\Debug>
```

# C-API und Rich Text

**Control rich text content with precision using the Lotus C API**
by Ethann Castell, THE VIEW, November/December 2007, Volume 13, Issue 6

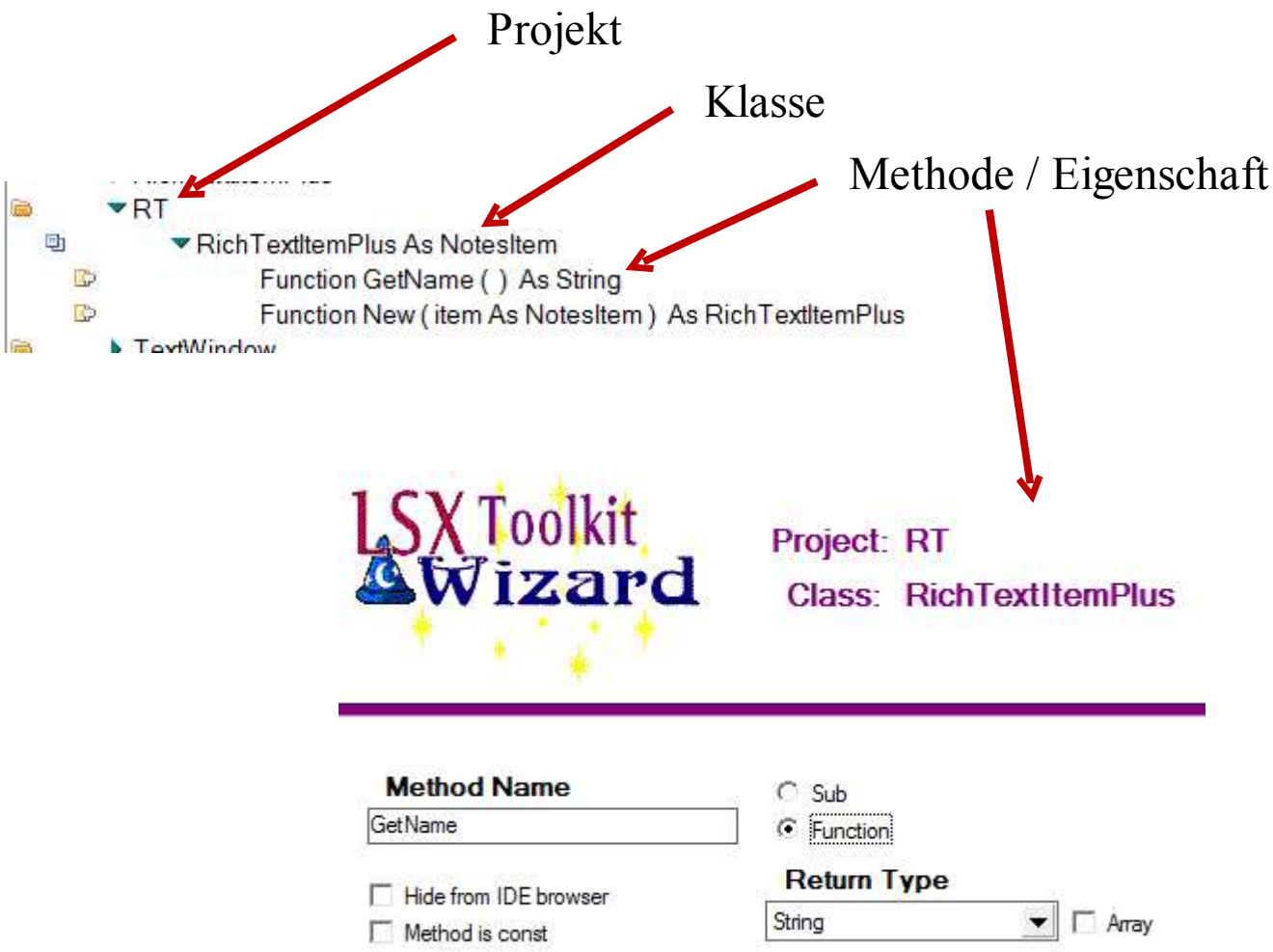
http://www.eview.com/eview/volr6.nsf/852561460065adc3852561130021446a/60e10c63f2e130328525738b00709d51?OpenDocument

# LSX Toolkit

# LSX

- Neuer Name LotusScript eXtension Toolkit
- Version 8.0 seit 2009
- Windows und Unix
- Erweitert LotusScript
- VC 2008 EE nicht unterstützt
  - Winreg.h Problem http://www.eknori.de/2009-12-21/lstatus-definition-problem-with-winreg-h/
  - Workaround: http://www.eknori.de/2009-12-22/build-lsx-8-0-solutions-with-visual-c-2008-express/
- Code Grundgerüst wird über Wizard erstellt
- Code Samples
  - Im Toolkit
  - Active Directory Picker ( Bill Buchan ) http://www.hadsl.com/hadsl.nsf/Documents/Active+Directory+Pickers!OpenDocument

# LSX


Projekt
Klasse
Methode / Eigenschaft
RT
RichTextItemPlus As NotesItem
Function GetName ( ) As String
Function New ( item As NotesItem ) As RichTextItemPlus
TextWindow
LSX Toolkit Wizard
Project: RT
Class: RichTextItemPlus
Method Name
GetName
Sub
Function
Hide from IDE browser
Method is const
Return Type
String
Array

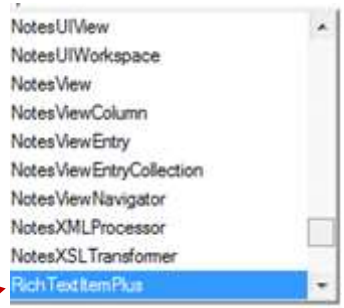
# LSX

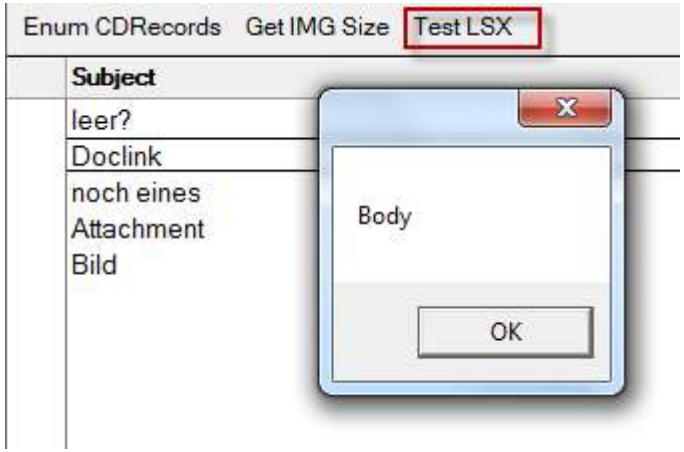

```
/*****************************************************************************
 *
 * Method Definitions for methods that are exposed to LotusScript
 *
 *****************************************************************************/
LSXString RichTextItemPlus::GetName()
{
    LSXString item_name;
    item_name += m_BaseNotesItem.GetProp(CNOTES_IPROP_NAME).getString();

    return item_name;

   //}}
}
```

# LSX

Uselsx „rt“

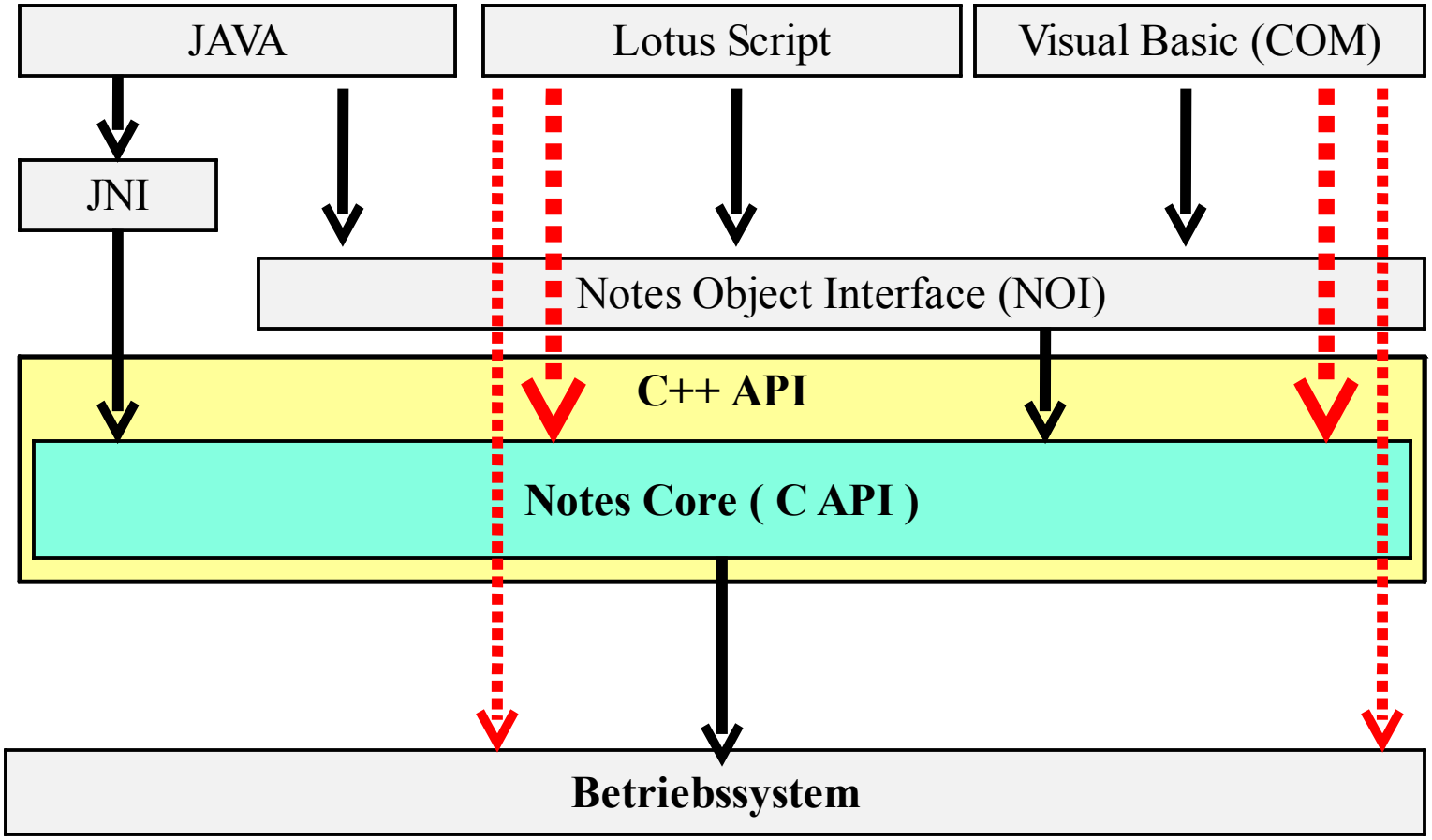
```
Sub Click(Source As Button)
    Dim s As New NotesSession
    Dim ndb As NotesDatabase
    Dim dcl As NotesDocumentcollection
    Dim doc As NotesDocument
    Set ndb = s.currentDatabase
    Set dcl=ndb.unprocesseddocuments
    Set doc = dcl.getfirstdocument
    Dim item As NotesItem
    Set item = doc.GetFirstItem("Body")
    Dim item_name As String

    Set RtItemPlus = New RichTextItemPlus(item)

    item_name =  RtItemPlus.GetName()

    Msgbox item_name
End Sub
```



# LS2CAPI

# LS2CAPI


JAVA
Lotus Script
Visual Basic (COM)
JNI
Notes Object Interface (NOI)
C++ API
Notes Core ( C API )
Betriebssystem

# Zur Erinnerung!!

*"A word of warning. This is difficult, horrible, tedious frustrating error prone work that WILL take far more time than you think, and WILL come back and bite you in the .. leg .."*

( Bill Buchan )

# LS2CAPI

**Vorteile**

- Kann direkt aus LotusScript / COM verwendet werden
- Keine zusätzlichen Tools erforderlich
- Keine zusätzlichen Dateien erforderlich

**Nachteile**

- Unterschiedliche Deklarationen für unterschiedliche Plattformen
- Nicht alle Funktionen können verwendet werden
  - Callback
  - Extension Manager
  - Addin
- Keine Verwendung der C++ API

# LS2CAPI – Signaturen umsetzen

## C Signatur

```
STATUS LNPUBLIC NSFDbGetBuildVersion(
DBHANDLE hDB,
WORD far *retVersion);
```

## Lotus Script Deklaration

```
Declare Function NSFDbGetBuildVersion Lib "nnotes.dll" (
Byval hDB As Long,
retVersion As Integer)
As Integer
```

# LS2CAPI – Signaturen umsetzen

| API | Win | Tux | AIX | Solaris | Mac |
|---|---|---|---|---|---|
| **BYTE** | BYTE | BYTE | BYTE | BYTE | BYTE |
| **BOOL** | Long | Long | Long | Long | Integer |
| **int** | Long | Long | Long | Long | Long |
| **LONG int** | Long | Long | Long | Long | Long |
| **WORD** | Integer | Integer | Integer | Integer | Integer |
| **SWORD** | Integer | Integer | Integer | Integer | Integer |
| **DWORD** | Integer | Integer | Integer | Integer | Integer |
| **HANDLE** | Long | Integer | Integer | Integer | Integer |
| **NOTEHANDLE** | Long | Integer | Integer | Integer | Integer |
| **DBHANDLE** | Long | Integer | Integer | Integer | Integer |
| **MEMHANDLE** | Long | Long | Long | Long | Long |
| **STATUS** | Integer | Integer | Integer | Integer | Integer |
| **Char *** | String | String | String | String | String |

# LS2CAPI – Platform Libraries

Die plattformspezifische Bibliothek/ DLL ergibt sich wie folgt

| | |
|---|---|
| • Windows/32: | nnotes.dll |
| • Solaris, Linux: | libnotes.so |
| • AIX: | lnotes_r.a |
| • HPUX: | libnotes.sl |
| • MacOs, OS/X: | NotesLib |
| • OS/400: | libnotes.srvpgm |
| • OS/390 | libnotes |
| • OS/2: | lnotes.dll |

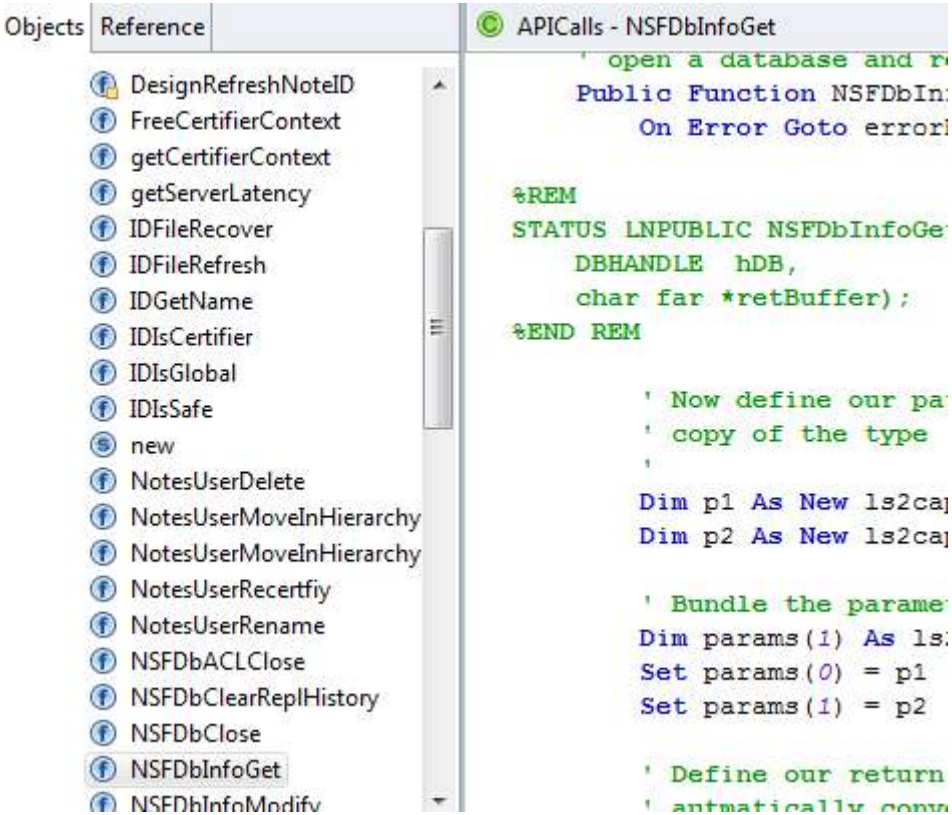
→ DEMO

# LS2CAPI - *class:ls2capiBaseClass*

- Definition der API Calls in Klassen
- Keine manuelle Deklaration
- Platformunabhängiger Aufruf

Download:

http://www.hadsl.com/hadsl.nsf/Documents/LS2CAPI+-+Calling+Notes+C-API+from+LotusScript!OpenDocument


→ DEMO

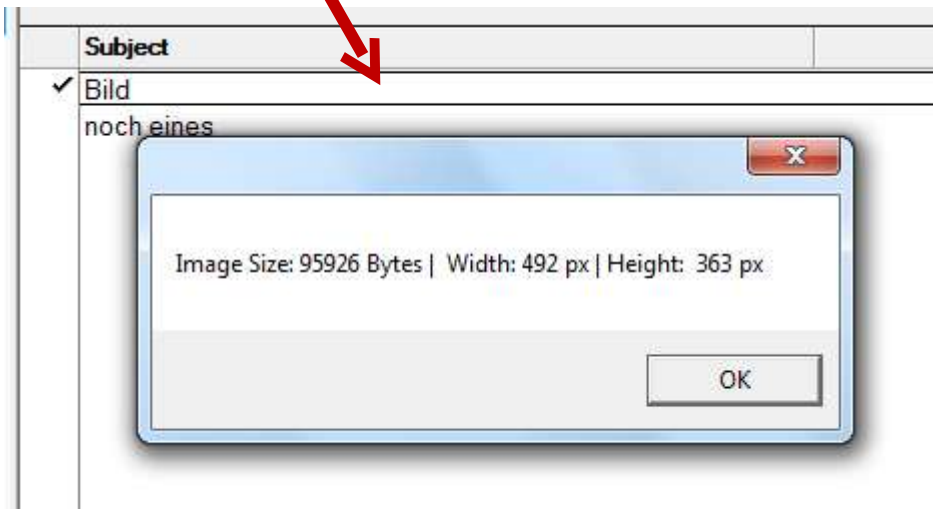
# LS2CAPI und Rich Text

# LS2CAPI und Rich Text

```
Sub Click(Source As Button)
    Dim s As New NotesSession,  imgHeader As CDIMAGEHEADER
    Dim ndb As NotesDatabase
    Dim item_blockid As blockid, value_blockid As blockid
    Dim idatatype As Integer, itype As Integer, irc As Integer
    Dim value_len As Long, pBuff As Long, ODS_OFFSET As Long
    Dim dcl As NotesDocumentcollection
    Dim doc As NotesDocument
    Set ndb = s.currentDatabase
    Set dcl=ndb.unprocesseddocuments
    Set doc = dcl.getfirstdocument
    irc = W32_NSFItemInfo (doc.handle, ITEM_OBJ, Len(ITEM_OBJ), item_blockid , idatatype , value_blockid , value_len )
    If irc = 0 Then

        ODS_OFFSET = &h84
        pBuff = W32_OSLockObject (value_blockid.pool) + ( Clng( value_blockid.block) And &H0FFFF& )

        Call GetImageInfo ( pbuff ,   ODS_OFFSET, imgheader )

        W32_OSUnLockObject (value_blockid.pool)
    End If

    Msgbox "Image Size: " + Cstr( imgheader.ImageDataSize) + " Bytes |  Width: " + Cstr( imgheader.width) +" px | Height:  " + Cstr( imgheader.height) + " px"
End Sub
```

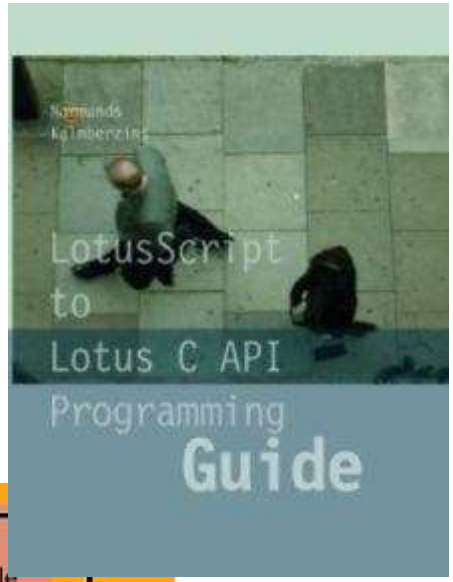
# LS2CAPI und Rich Text

```
Sub GetImageInfo ( p As Long, offset As Long , imgHeader As CDIMAGEHEADER )
    Call W32_ODSReadMemory (p +  offset, ODS_WORD, imgheader.header.signature, 1)
    Call W32_ODSReadMemory (p + offset+   2, ODS_DWORD, imgheader.header.length, 1)
    Call W32_ODSReadMemory (p + offset+   6, ODS_WORD, imgheader.ImageType, 1)
    Call W32_ODSReadMemory (p + offset+   8, ODS_WORD, imgheader.width, 1)
    Call W32_ODSReadMemory (p + offset+  10, ODS_WORD, imgheader.Height, 1)
    Call W32_ODSReadMemory (p + offset+  12, ODS_DWORD, imgheader.ImageDataSize, 1)
    Call W32_ODSReadMemory (p + offset+  16, ODS_DWORD, imgheader.SegCount, 1)
End Sub
```

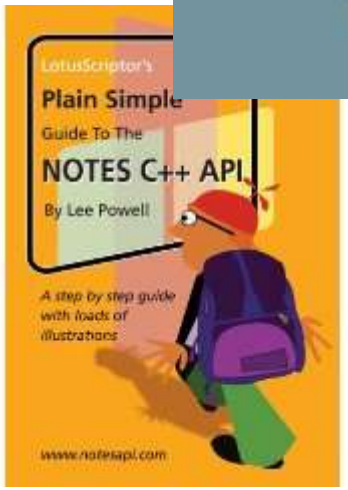
# Literatur

Normunds Kalbernzin –
*LotusScript to Lotus C API Programming Guide* ( ls2capi.com )




Lee Powell –
*LotusScriptor's Plain Simple Guide to the Lotus Notes C++ API*

**Download** : http://www.eknori.de/downloads/capibook.pdf




Ethann Castell –
*Getting started with the C-API*
*Control rich text content with precision using the C-API*
The View, 09 / 11 , 2007

# Samples

**TriggerHappy**
*http://www.openntf.org/Projects/pmt.nsf/ProjectLookup/Trigger%20Happy*

**DominoDefrag**
*http://www.openntf.org/Projects/pmt.nsf/ProjectLookup/DominoDefrag*

**MailScan**
*http://www.openntf.org/Projects/pmt.nsf/ProjectLookup/MailScan*

**Creating a custom Administration Process Request Handler**
*http://www.ibm.com/developerworks/lotus/library/ls-Custom_AdminP_Handler/*

**Demo Applikation zur Session**
*http://openntf.org/Projects/codebin/codebin.nsf/CodeByDate/F8207E308820C2FA862576DD00255B67*

# Fragen

Contact me at:
**ulrich.krause@eknori.de**

Bitte das Ausfüllen der Session Bewertung nicht vergessen !

# E N D E