

# Re: How to decode a TNEF Stream?

---

*Source:*

<http://www.tech-archive.net/Archive/Development/microsoft.public.win32.programmer.messaging/2006-06/msg0010>

---

- *From:* Dan Mitchell <[djmitchella@xxxxxxxxxx](mailto:djmitchella@xxxxxxxxxx)>
  - *Date:* Wed, 28 Jun 2006 08:34:29 -0700
- 

"Yanis Belgaid" <[ybel@xxxxxxxxxx](mailto:ybel@xxxxxxxxxx)> wrote in  
[news:d9c9a5feea9b4930b0ce8a101e18eb8f@xxxxxxxxxxxxx](mailto:news:d9c9a5feea9b4930b0ce8a101e18eb8f@xxxxxxxxxxxxx):

Does anybody know how to decode a tnef stream using c#  
to get a message body and possibly any attachments?  
I am using windows 2003.

Not C#, but here's some Delphi code Dmitry posted to mapi-l a while ago  
(<http://peach.ease.lsoft.com/scripts/wa.exe?A2=ind0203&L=mapi-l&T=0&P=28500>)

```
{  
*****
```

Unit to extract attachments from a TNEF stream \_without\_  
using MAPI32.DLL.

RTF text will replace Message body if MAPI32.DLL is installed  
(i.e. if any version of Outlook/Exchange client is present)  
and ExtractRTF = TRUE

Dmitry Streblechenko for WinShoes, 1999

```
*****  
}
```

```
unit WinshoeTNEFConverter;  
  
interface  
  
uses Classes, Windows, SysUtils, ActiveX, Winshoes, WinshoeMessage;  
  
procedure MSTNEFToAttachments(Msg : TWinshoeMessage; Stream :  
TStream; ExtractRTF : boolean);  
  
type
```

## Re: How to decode a TNEF Stream?

```
TNEFException = class(Exception);

TTNEFConverter = class(TObject)
private
FMsg :TWinshoeMessage;
FExtractRTF : boolean;
procedure NewAttachment;
function LastAttach:TAttachment;
procedure SetAttachmentData(Data : pointer; Length : integer);
procedure SetAttachmentFileName(Value : string; ReplaceOld :
boolean);
procedure ProcessRTF(Value : pointer; ValueSize : integer);
procedure OnReadProp(PropTag:ULONG; var Value; ValueSize:ULONG;
namedPropGUID:TGUID; namedKind,
namedIID:ULONG; namedPropName:string);
public
constructor CreateOnFile(Msg : TWinshoeMessage; FileName:string;
ExtractRTF : boolean);
constructor CreateOnStream(Msg : TWinshoeMessage; Stream:TStream;
ExtractRTF : boolean);
destructor Destroy;override;
end;

implementation

uses GlobalWinshoe, StringsWinshoe, SystemWinshoe;

// Borrow some declarations from MAPITags, MAPIDefs, TNEF
// so we don't need to use 'em

const
MV_FLAG = $1000; { Multi-value flag }
PT_UNSPECIFIED = ULONG(0); { (Reserved for interface use);
type doesn't matter to caller }
PT_NULL = ULONG(1); { NULL property value }
PT_I2 = ULONG(2); { Signed 16-bit value }
PT_LONG = ULONG(3); { Signed 32-bit value }
PT_R4 = ULONG(4); { 4-byte floating point }
PT_DOUBLE = ULONG(5); { Floating point double }
PT_CURRENCY = ULONG(6); { Signed 64-bit int (decimal w/
4 digits right of decimal pt); }
PT_APPTIME = ULONG(7); { Application time }
PT_ERROR = ULONG(10); { 32-bit error value }
PT_BOOLEAN = ULONG(11); { 16-bit boolean (non-zero
true); }
PT_OBJECT = ULONG(13); { Embedded object in a property
}
PT_I8 = ULONG(20); { 8-byte signed integer }
PT_STRING8 = ULONG(30); { Null terminated 8-bit
character string }
PT_UNICODE = ULONG(31); { Null terminated Unicode string
```

Re: How to decode a TNEF Stream?

```
}
PT_SYSTIME = ULONG(64); { FILETIME 64-bit int w/ number
of 100ns periods since Jan 1,1601 }
PT_CLSID = ULONG(72); { OLE GUID }
PT_BINARY = ULONG(258); { Uninterpreted (counted byte
array); }
PT_SHORT = PT_I2;
PT_I4 = PT_LONG;
PT_FLOAT = PT_R4;
PT_R8 = PT_DOUBLE;
PT_LONGLONG = PT_I8;
{$IFDEF UNICODE}
PT_TSTRING = PT_UNICODE;
{$ELSE}
PT_TSTRING = PT_STRING8;
{$ENDIF}

PR_ATTACH_LONG_FILENAME = (PT_TSTRING) or (($3707) shl 16);
PR_ATTACH_FILENAME = (PT_TSTRING) or (($3704) shl 16);
PR_RTF_COMPRESSED = (PT_BINARY) or (($1009) shl 16);

MNID_ID = 0;
MNID_STRING = 1;

TNEF_SIGNATURE : ULONG = $223E9F78;
LVL_MESSAGE = 1;
LVL_ATTACHMENT = 2;
atpTriples = $0000;
atpString = $0001;
atpText = $0002;
atpDate = $0003;
atpShort = $0004;
atpLong = $0005;
atpByte = $0006;
atpWord = $0007;
atpDword = $0008;
atpMax = $0009;

attAttachData = ((atpByte shl 16) or $800F); //
PR_ATTACH_DATA_xxx */
attAttachTitle = ((atpString shl 16) or $8010); //
PR_ATTACH_FILENAME */
attAttachMetaFile = ((atpByte shl 16) or $8011); //
PR_ATTACH_RENDERING */
attAttachCreateDate = ((atpDate shl 16) or $8012); //
PR_CREATION_TIME */
attAttachModifyDate = ((atpDate shl 16) or $8013); //
PR_LAST_MODIFICATION_TIME */
attAttachTransportFilename = ((atpByte shl 16) or $9001); //
PR_ATTACH_TRANSPORT_NAME */
attAttachRenddata = ((atpByte shl 16) or $9002);
```

Re: How to decode a TNEF Stream?

```
attMAPIProps = ((atpByte shl 16) or $9003);
attRecipTable = ((atpByte shl 16) or $9004); //
PR_MESSAGE_RECIPIENTS */
attAttachment = ((atpByte shl 16) or $9005);
attTnefVersion = ((atpDword shl 16) or $9006);
```

```
MAPI_NO_COINIT = $00000008;
MAPI_INIT_VERSION = 0;
```

```
type
TMAPIINIT_0 =
record
ulVersion : ULONG;
ulFlags : ULONG;
end;
```

```
procedure MSTNEFToAttachments(Msg : TWinshoeMessage; Stream : TStream;
ExtractRTF : boolean);
var Conv:TTNEFConverter;
begin
Conv:=nil;
try
Conv:=TTNEFConverter.CreateOnStream(Msg, Stream, ExtractRTF);
finally
Conv.Free;
end;
end;
```

```
function Pad(Number, Modulo:ULONG):integer;
begin
if (Number mod Modulo) > 0 then Result:=Number+(Modulo - (Number mod
Modulo))
else Result:=Number;
end;
```

```
function GetTempFileName:string;
var TempDir:string;
begin
SetLength(TempDir,MAX_PATH);
GetTempPath(MAX_PATH+1,PChar(TempDir));
SetLength(TempDir,Length(PChar(TempDir)));
SetLength(Result,MAX_PATH);
Windows.GetTempFileName(PChar(TempDir),'SLX',0,PChar(Result));
SetLength(Result,Length(PChar(Result)));
end;
```

```
//*****
// TTNEFConverter
//*****
```

```
constructor TTNEFConverter.CreateOnFile(Msg : TWinshoeMessage;
```

## Re: How to decode a TNEF Stream?

```
FileName:string; ExtractRTF : boolean);
var FS:TFileStream;
begin
FS:=nil;
try
FS:=TFileStream.Create(FileName,fmOpenRead or fmShareDenyWrite);
CreateOnStream(Msg, FS, ExtractRTF);
finally
FS.Free;
end;
end;

//*****

constructor TTNEFConverter.CreateOnStream(Msg : TWinshoeMessage;
Stream:TStream; ExtractRTF : boolean);

procedure Process_Mapi_Prop(Stream:TStream);
var PropTag:ULONG;
PropKind:ULONG;
i:integer;
NumValues:ULONG;
namedPropName:WideString;
namedPropNameLength:integer;
namedPropGUID:TGUID;
namedKind:ULONG;
namedIID:ULONG;
ObjectGUID:TGUID;
ValueSize:integer;
Value:ULONG; //can be a pointer too!!
I2:SmallInt;
long4:longint;
float4:single;
double8:double;
currency8:comp;
bool2:word;
systime8:comp;
apptime8:TDateTime;
error4:ULONG;
i8:Comp;
ClassID16:TGUID;
begin
Stream.Read(PropTag,SizeOf(PropTag));
PropKind:=PropTag and $FFFF;
namedKind:=0;
namedIID:=0;
namedPropName:="";
ZeroMemory(@namedPropGUID,SizeOf(namedPropGUID));
//Check if named prop and read named props data
if ((PropTag shr 16) >= $8000) and ((PropTag shr 16) <= $FFFE) then
begin
```

## Re: How to decode a TNEF Stream?

```
Stream.Read(namedPropGUID,SizeOf(namedPropGUID));
Stream.Read(namedKind,SizeOf(namedKind));
if namedKind = MNID_ID then begin
Stream.Read(namedIID,SizeOf(namedIID));
end
else if namedKind =MNID_STRING then begin
Stream.Read(namedPropNameLength,SizeOf(namedPropNameLength));
Setlength(namedPropName,namedPropNameLength);
Stream.Read(namedPropName[1],namedPropNameLength);
Stream.Position:=Stream.Position+Pad(namedPropNameLength,4)-
namedPropNameLength;
end
else raise TNEFException.Create('Invalid named property kind');
end;
NumValues:=1;
if ((PropKind and MV_FLAG) = MV_FLAG) or (PropKind = PT_BINARY) or
(PropKind = PT_STRING8) or (PropKind = PT_UNICODE) or (PropKind =
PT_OBJECT) then begin
Stream.Read(NumValues,SizeOf(NumValues));
end;
if ((PropKind and MV_FLAG) = MV_FLAG) or (PropKind = PT_BINARY) or
(PropKind = PT_STRING8) or (PropKind = PT_UNICODE) or (PropKind =
PT_OBJECT) then
begin
for i:=0 to NumValues-1 do begin
Stream.Read(ValueSize,SizeOf(ValueSize));
if (PropKind = PT_OBJECT) then Stream.Read(ObjectGUID,SizeOf
(ObjectGUID));
GetMem(pointer(Value),ValueSize);
Stream.Read(pointer(Value)^,ValueSize);
OnReadProp(PropTag, Value, ValueSize, namedPropGUID, namedKind,
namedIID, namedPropName);
FreeMem(pointer(Value));
Stream.Position:=Stream.Position+Pad(ValueSize,4)-ValueSize;
end;
end
else for i:=0 to NumValues-1 do begin
ValueSize:=0;
//must add calls to OnReadProp() in "case" below, but there's no
need for now:
//we are not interested in them at least in this version
case PropKind of
PT_UNSPECIFIED:begin ValueSize:=0; end;
PT_NULL : begin ValueSize:=0; end;
PT_I2 : begin Stream.Read(I2,SizeOf(I2)); ValueSize:
=SizeOf(I2); end;
PT_LONG : begin Stream.Read(long4,SizeOf(long4));
ValueSize:=SizeOf(long4); end;
PT_R4 : begin Stream.Read(float4,SizeOf(float4));
ValueSize:=SizeOf(float4); end;
PT_DOUBLE : begin Stream.Read(double8,SizeOf(double8));
```

## Re: How to decode a TNEF Stream?

```
ValueSize:=SizeOf(double8); end;
PT_CURRENCY : begin Stream.Read(currency8,SizeOf(currency8));
ValueSize:=SizeOf(currency8); end;
PT_APPTIME : begin Stream.Read(apptime8,SizeOf(apptime8));
ValueSize:=SizeOf(apptime8); end;
PT_ERROR : begin Stream.Read(error4,SizeOf(error4));
ValueSize:=SizeOf(error4); end;
PT_I8 : begin Stream.Read(i8,SizeOf(i8)); ValueSize:
=SizeOf(i8); end;
PT_BOOLEAN : begin Stream.Read(bool2,SizeOf(bool2));
ValueSize:=SizeOf(bool2); end;
PT_SYSTIME : begin Stream.Read(systime8,SizeOf(systime8));
ValueSize:=SizeOf(systime8); end;
PT_CLSID : begin Stream.Read(ClassID16,SizeOf(ClassID16));
ValueSize:=SizeOf(ClassID16); end;
end;
Stream.Position:=Stream.Position+Pad(ValueSize,4)-ValueSize;
end;
end;
```

```
procedure Process_MAPI_Props(Stream:TStream);
var NumProps:ULONG;
i:integer;
begin
Stream.Read(NumProps,SizeOf(NumProps));
for i:=0 to NumProps-1 do begin
try
Process_Mapi_Prop(Stream);
except
Break;
end;
end;
end;
```

```
procedure Read_attTnefVersion(Stream:TStream);
var LVL:byte;
att:ULONG;
AttLength:ULONG;
checksum:word;
Dummy:ULONG;
begin
Stream.Read(LVL,SizeOf(LVL));
if LVL <> LVL_MESSAGE then raise TNEFException.Create('Invalid
message');
Stream.Read(att,SizeOf(att));
if att <> attTnefVersion then raise TNEFException.Create('Invalid
attribute');
Stream.Read(AttLength,SizeOf(AttLength));
Dummy:=0;
Stream.Read(Dummy,AttLength);
if Dummy <> $00010000 then raise TNEFException.Create('Invalid
```

## Re: How to decode a TNEF Stream?

```
sequence');
Stream.Read(checksum,SizeOf(checksum));
end;

function Read_Att_Attribute(Stream:TStream):boolean;
var att:ULONG;
AttrLength:ULONG;
checksum:word;
var LVL:byte;
Data:pointer;
MemStr:TMemoryStream;
begin
Result:=false;
Stream.Read(LVL,SizeOf(LVL));
if LVL <> LVL_ATTACHMENT then begin
Stream.Seek(-SizeOf(LVL), soFromCurrent);
end
else begin
Result:=true;
Stream.Read(att,SizeOf(att));
Data:=nil;
try
Stream.Read(AttrLength,SizeOf(AttrLength));
GetMem(Data,AttrLength);
Stream.Read(Data^,AttrLength);
Stream.Read(checksum,SizeOf(checksum));
case att of
attAttachData :
begin
if LastAttach <> nil then SetAttachmentData(Data,
AttrLength);//LastAttach.SetContents(Data,AttrLength);
end;
attAttachTitle :
begin
if LastAttach <> nil then SetAttachmentFileName(PChar
(Data), FALSE);//LastAttach.FShortFileName:=PChar(Data);
end;
attAttachMetaFile :
begin
end;
attAttachCreateDate :
begin
end;
attAttachModifyDate :
begin
end;
attAttachTransportFilename :
begin
end;
attAttachRenddata :
begin
```

## Re: How to decode a TNEF Stream?

```
NewAttachment;
end;
attAttachment :
begin
try
MemStr:=nil;
try
MemStr:=TMemoryStream.Create;
MemStr.Write(Data^,AttrLength);
MemStr.Position:=0;
Process_MAPI_Props(MemStr);
finally
MemStr.Free;
end;
except
end;
end;
attMAPIProps :
begin
end
else
begin
end;
end;
finally
FreeMem(Data);
end;
end;
end;

function Read_Msg_Attribute(Stream:TStream):boolean;
var att:ULONG;
AttrLength:ULONG;
checksum:word;
var LVL:byte;
MemStr:TMemoryStream;
Data:pointer;
begin
Result:=false;
Stream.Read(LVL,SizeOf(LVL));
if LVL <> LVL_MESSAGE then begin
Stream.Seek(-SizeOf(LVL), soFromCurrent);
end else begin
Result:=true;
Stream.Read(att,SizeOf(att));
Stream.Read(AttrLength,SizeOf(AttrLength));
GetMem(Data,AttrLength);
Stream.Read(Data^,AttrLength);
try
Stream.Read(checksum,SizeOf(checksum));
case att of
```

## Re: How to decode a TNEF Stream?

```
attMAPIProps :
begin
try
MemStr:=nil;
try
MemStr:=TMemoryStream.Create;
MemStr.Write(Data^,AttrLength);
MemStr.Position:=0;
Process_MAPI_Props(MemStr);
finally
MemStr.Free;
end;
except
end;
end;
finally
FreeMem(Data);
end;
end;
end;

procedure Read_Att_AttributeSeq(Stream:TStream);
var p:integer;
begin
while Stream.Position < Stream.Size-SizeOf(ULONG) do begin
p:=Stream.Position;
try
Read_Att_Attribute(Stream);
except
Stream.Position:=p;
Break;
end;
end;
end;

procedure Read_Msg_AttributeSeq(Stream:TStream);
var p:integer;
begin
while Stream.Position < Stream.Size-SizeOf(ULONG) do begin
p:=Stream.Position;
if not Read_Msg_Attribute(Stream) then begin
Stream.Position:=p;
Break;
end;
end;
end;

procedure Read_Message_Seq(Stream:TStream);
var p:longint;
begin
```

## Re: How to decode a TNEF Stream?

```
p:=Stream.Position;
try
Read_attTnefVersion(Stream);
except
Stream.Position:=p;
end;
Read_Msg_AttributeSeq(Stream);
end;
```

```
procedure Read_Attach_Seq(Stream:TStream);
var p:longint;
begin
while Stream.Position < Stream.Size do begin
p:=Stream.Position;
try
Read_Att_AttributeSeq(Stream);
except
Stream.Position:=p;
Break;
end;
end;
end;
```

```
procedure Read_Object(Stream:TStream);
var Sig:ULONG;
Key:Word;
p:longint;
begin
Stream.Position:=0;
Stream.Read(Sig,SizeOf(Sig));
if Sig <> TNEF_SIGNATURE then raise TNEFException.Create('Invalid
TNEF signature');
Stream.Read(Key,SizeOf(Key));
p:=Stream.Position;
try
Read_Message_Seq(Stream);
Read_Attach_Seq(Stream);
except
Stream.Position:=p;
try
Read_Attach_Seq(Stream);
except
Stream.Position:=p;
end;
end;
end;
```

```
var p:integer;
begin
inherited;
FMsg:=Msg;
```

Re: How to decode a TNEF Stream?

```
FExtractRTF:=ExtractRTF;
p:=Stream.Position;
Stream.Position:=0;
try
Read_Object(Stream);
finally
Stream.Position:=p;
end;
end;

//*****

destructor TTNEFConverter.Destroy;
begin
inherited;
end;

//*****

procedure TTNEFConverter.NewAttachment;
begin
if FMsg <> nil then FMsg.Attachments.Add;
end;

//*****

function TTNEFConverter.LastAttach:TAttachment;
begin
if (FMsg <> nil) and (FMsg.Attachments.Count > 0)
then Result:=FMsg.Attachments[FMsg.Attachments.Count-1]
else Result:=nil;
end;

//*****

procedure TTNEFConverter.OnReadProp(PropTag:ULONG; var Value;
ValueSize:ULONG;
namedPropGUID:TGUID; namedKind,
namedIID:ULONG; namedPropName:string);
begin
case PropTag of
PR_ATTACH_LONG_FILENAME :
//Attachment long file name
begin
SetAttachmentFileName(PChar(pointer(Value)), TRUE);
end;
PR_ATTACH_FILENAME :
//Attachment short file name
begin
SetAttachmentFileName(PChar(pointer(Value)), FALSE);
```

Re: How to decode a TNEF Stream?

```
end;
PR_RTF_COMPRESSED :
// compressed RTF
begin
if FExtractRTF then ProcessRTF(pointer(Value), ValueSize);
end;
end;
end;

//*****

procedure TTNEFConverter.SetAttachmentData(Data: pointer; Length:
integer);
var strmOut:TStream;
begin
if LastAttach <> nil then
with LastAttach do begin
strmOut:=nil;
try
if Assigned (OnGetAttachmentStream) then OnGetAttachmentStream
(strmOut);
if strmOut = nil then begin
StoredPathName := MakeTempFilename;
strmOut := TFileStream.Create(StoredPathName, fmCreate );
end;
strmOut.Write(Data^, Length);
finally
strmOut.Free;
end;
end;
end;

//*****

procedure TTNEFConverter.SetAttachmentFileName(Value: string;
ReplaceOld: boolean);
begin
if (LastAttach <> nil) and ((LastAttach.FileName = "") or ReplaceOld)
then LastAttach.FileName:=Value;
end;

//*****

type TWrapCompressedRTFStream = function (lpCompressedRTFStream :
IStream;
ulflags : ULONG; var
lpUncompressedRTFStream :
IStream): HRESULT; stdcall;
TMAPIInitialize = function (lpMapiInit : Pointer) : HRESULT;
stdcall;
TMAPIUninitialize = procedure ; stdcall;
```

## Re: How to decode a TNEF Stream?

```
var MAPIDLLHandle : THandle = INVALID_HANDLE_VALUE;
WrapCompressedRTFStream : TWrapCompressedRTFStream = nil;
MAPIInitialize:TMAPIInitialize = nil;
MAPIUninitialize:TMAPIUninitialize = nil;

procedure TTNEFConverter.ProcessRTF(Value: pointer; ValueSize: integer);
var ComInitRes:HResult;
MemStream:TMemoryStream;
ComStream, UncomprStream : IStream;
Buffer : array[0..$FFFF-1] of byte;
cbRead,OldLen:longint;
strRTF:string;
MAPIInit : TMAPIINIT_0;
begin
if (FMsg <> nil) and (FExtractRTF) then begin
if MAPIDLLHandle = INVALID_HANDLE_VALUE then begin
MAPIDLLHandle := LoadLibrary('MAPI32.DLL');
if MAPIDLLHandle <> INVALID_HANDLE_VALUE then begin
WrapCompressedRTFStream:=GetProcAddress
(MAPIDLLHandle,'WrapCompressedRTFStream');
MAPIInitialize:=GetProcAddress(MAPIDLLHandle,'MAPIInitialize');
MAPIUninitialize:=GetProcAddress
(MAPIDLLHandle,'MAPIUninitialize');
end;
end;
if (@MAPIInitialize <> nil) and (@MAPIUninitialize <> nil) and
(@WrapCompressedRTFStream <> nil) then begin
ComInitRes:=CoInitializeEx(nil, COINIT_APARTMENTTHREADED);
with MAPIInit do begin
ulVersion := MAPI_INIT_VERSION;
ulFlags := MAPI_NO_COINIT;
end;
MAPIInitialize(@MAPIInit);
MemStream:=TMemoryStream.Create;
MemStream.Write(Value^, ValueSize);
MemStream.Position:=0;
ComStream:=TStreamAdapter.Create(MemStream, soReference);
if S_OK = WrapCompressedRTFStream(ComStream, 0, UncomprStream)
then begin
strRTF:="";
while (S_OK = UncomprStream.Read(@Buffer, SizeOf(Buffer),
@cbRead)) and (cbRead > 0) do begin
OldLen:=Length(strRTF);
SetLength(strRTF, OldLen + cbRead);
Move(Buffer[0], strRTF[OldLen+1], cbRead);
end;
if strRTF <> " then begin
FMsg.Text.Text:=strRTF;
end;
UncomprStream:=nil;
end;
```

## Re: How to decode a TNEF Stream?

```
end;  
ComStream:=nil;  
MemStream.Free;  
MAPIUninitialize;  
if (ComInitRes = S_OK) or (ComInitRes = S_FALSE) then  
CoUninitialize;  
end;  
end;  
end;
```

initialization

```
MAPIDLLHandle := INVALID_HANDLE_VALUE;
```

finalization

```
if MAPIDLLHandle <> INVALID_HANDLE_VALUE then FreeLibrary  
(MAPIDLLHandle);  
end.  
.
```