

# NTK Project Converter

---

This document describes the MacNTK to WinNTK project converter, a drag-and-drop utility that runs on a Macintosh® OS-based computer. This utility converts files created by Newton® Toolkit® on the Macintosh into the format used by Newton Toolkit version 1.6 for Windows.

You drag the files you want converted onto the icon labeled MacNTK to WinNTK. You can convert individual files or entire projects. Common usage is to drag the folder containing your project onto the converter icon.

The converter does not change the original files. Instead, it creates new files in a folder named Converted Files. If you drop a folder on the utility, the folder's entire hierarchy is duplicated within Converted Files, and the new files are placed in the appropriate subfolders. You can transfer the new files to a Windows system by the usual channels: a network, for example, or a floppy disk.

You have the option of either preserving the Macintosh filenames or converting them to DOS's 8.3 format. When the files are transferred onto a volume that supports only the short 8.3 filenames, long filenames are converted into 8.3 format, even if you preserved them during the conversion. Because this change occurs outside the MacNTK to WinNTK converter or Windows NTK itself, it invalidates file references within the project. If you're transferring files by way of a volume that support only short filenames, turn

on the 8.3 conversion option, but remember that you might have to make some changes yourself, as described in “What It Doesn’t Do” beginning on page 1-3.

## What It Does

---

MacNTK to WinNTK converts Macintosh NTK projects into Windows NTK projects. The converter handles

- NTK project files
- NTK layout files
- Macintosh resource files
- text files

The converter logs all changes made to your project during the conversion in a text file named Conversion log stored in the Converted Files folder. You can open it with a text editor to see what’s been changed.

### NTK Project Files

---

- Internal format is changed to Windows NTK format.
- Resource files are removed from the project list and replaced with their Windows equivalents, if appropriate, as described in “Resource Files” beginning on page 1-5.
- Resources in the resource fork are saved to their own files (see “Resources in Non-Resource Files” beginning on page 1-7).

### NTK Layout Files

---

- Internal format is changed to Windows NTK format.

## NTK Project Converter

- Resources in the resource fork are saved to their own files (see “Behind the Scenes” beginning on page 1-4).

## Text Files

---

- Line endings are converted to DOS/Windows format.
- Characters are translated, as described in “Text Files” beginning on page 1-5.

## Resource Files

---

- Individual resources are saved to their own files (see “Resource Files” beginning on page 1-5).

## What It Doesn't Do

---

The MacNTK to WinNTK converter can't update all references to filenames. You must make changes by hand if your source project references

- *layout\_filename* and *printFormat\_filename* variables or
- HFS-specific pathnames.

### *layout\_filename* and *printFormat\_filename* Variables

---

These variables are references to views created when NTK compiles files that contain protos and print layouts. If the names of your files change during the conversion process, you'll need to change the names of these variables to match the new names.

## HFS-Specific Pathnames

---

You must change HFS-specific pathnames in NewtonScript code from Macintosh format to Windows format. For example:

```
OpenResFile ("HD:Folder:MyNextResourceFile.rsrc");
```

must change to:

```
OpenResFile ("D:\\Folder\\MyNextResourceFile.rsrc");
```

Note the doubling of the backslashes. This is because the NewtonScript interpreter uses the backslash (\) as an escape character the same way C does.

### Note

At this release, NTK cannot use the usual mechanism for resolving files whose Macintosh filenames include either the slash (/) or the backslash (\) character. Remove all slash characters from your source filenames before the conversion—and, of course, update any pathnames in NewtonScript code to reflect the new names. ♦

“Filename Changes” on page 1-9 describes how the converter changes Macintosh filenames for compatibility with Windows and how NTK resolves references to files whose names have changed during conversion.

## Behind the Scenes

---

The converter converts project and layout files to WinNTK format. It also

- changes line-endings and translates special characters in text files
- breaks up resource files
- converts selected resources to other formats
- creates and maintains a file that’s used to resolve filename changes

## Text Files

---

The converter changes line-endings in text files to follow Windows conventions and does some translation of special characters.

Both Macintosh and Windows represent special characters with the values 128 through 255. Some characters, like the copyright symbol, are represented by the same numerical value (177) on both platforms. Some characters, like the bullet, are represented by different values (165 on the Macintosh, 149 on Windows). Some characters exist only on one platform (like the Greek omega on the Macintosh). Taking the differences into account, the converter performs the necessary and possible translations in text files.

## Resource Files

---

The converter breaks each resource file in your project into a collection of new files, one or more for each resource. The names of the new files follow this format:

*<resource type><resource ID in hex><resource name if any>.res*

Thus, a Macintosh resource of type DATA with the resource ID 128 and the name "My Private Data" is saved to a file named DATA0080(My Private Data).res. An unnamed resource of type TEXT with ID 100 is saved to TEXT0064.res. The converter creates multiple files from a PICT resource, as described later in this section.

It may not be possible to generate a valid FAT or NTFS filename from a resource specification (because the resource type, for example, contains characters not allowed in those file systems). Some resources may therefore be saved to files with slightly different names. Illegal characters are simply dropped in the conversion. For instance, the resource type F/RD contains a slash character (/), which is not allowed under FAT or NTFS. A resource with type F/RD and ID 200, for example, would be written to the file FRD00C8.res.

Whenever it converts resource files, the converter creates a file called RsrcMap.txt that maps the names of the new resource files to the type, ID,

## NTK Project Converter

and name of the resource each holds. The `GetResource` and `GetNamedResource` functions use the `RsrcMap.txt` file to find the specified resource.

The `RsrcMap.txt` file contains one line for each resource file; each line follows this format:

```
<filename><tab><resource type><tab><resource ID><tab><resource name>
```

When your NewtonScript functions execute `GetResource` or `GetNamedResource` calls, those functions open the `RsrcMap.txt` file, find the entry corresponding to the parameters passed to them, find the appropriate filename, and open the file with that name. The data within that file is read in and returned as if a resource were being returned on the Mac.

### Note

The `CloseResFile` function does not immediately reclaim all memory allocated by `GetResource` and `GetNamedResource`—the memory is reclaimed when you quit Windows NTK. You can avoid the memory leak by using the Windows NTK equivalents: `GetBMPAsBits`, `GetWAVSound`, `GetWAVSound11`, and `LoadDataFile`. ♦

Named PICT resources receive special treatment in the conversion. First, a PICT is saved to its own Windows-style resource file just like any other resource; you can use the `GetResource` or `GetNamedResource` calls to access this data.

Second, the PICT is rasterized and saved to a file with the name:

```
bits<ID in Hex>( <resource name> ) .res
```

This file is read when you execute the `GetPICTAsBits` function call.

Finally, the rasterized bitmap is also saved to a `.BMP` file so that you can access the picture from an icon or picture slot editor. These `.BMP` files are automatically added to the project during the conversion process. The name of the `.BMP` file is the same as the resource's name, with the `.BMP` suffix; if the resource does not have a name, it's neither converted nor added to the project.

## NTK Project Converter

When a resource file is broken up, a new directory is created to hold the new files. This directory has the same name as the resource file that generated it. In turn, all of these newly created resource directories are created in a directory called Resources. Consider, for example, two resource files: one named Fred containing a PICT resource with ID 128 and the name "My Button", and one named Ethyl containing a DATA resource with ID 42 and no name. After the conversion, the project folder contains the following directories and files:

```
MyProject/  
    converted project and layout files  
Resources/  
    Fred/  
        PICT0080(MyButton).res  
        bits0080(MyButton).res  
        MyButton.BMP  
        RsrcMap.txt  
    Ethyl/  
        DATA002A.res  
        RsrcMap.txt
```

## Resources in Non-Resource Files

---

Project, layout, and text files contain resources of their own, for use by NTK and the Macintosh system software. During conversion, some of these resources are saved into individual files, like resources in resource files. Many, however, are either ignored or converted into a different format for use in the Windows environment.

Individual files are not created for resources of the following types and IDs:

■ In project files:

```
'ckid' any ID  
'PRNT' any ID  
'STR ' -16396
```

## NTK Project Converter

```
'PJPF' 9999
```

```
'PJST' 9999
```

### ■ In layout files (including user protos):

```
'ckid' any ID
```

```
'PRNT' any ID
```

```
'STR ' -16396
```

```
'FMST' 9999
```

### ■ In script files:

```
'ckid' any ID
```

```
'PJST' any ID
```

```
'SLCT' any ID
```

```
'styl' any ID
```

```
'MPSR' any ID
```

```
'MWBB' any ID
```

```
'BBSR' any ID
```

```
'BBST' any ID
```

```
'PRNT' any ID
```

```
'STR ' -16396
```

Most of those resources are skipped because they contain information only relevant to a particular development system. For instance, MPSR resources are added by MPW to contain font information, window size and location, and user-defined markers. Other resources are removed because they are handled by a different part of the conversion. For instance, the PJPF resource contains the project settings. These settings are moved into the contents of the WinNTK project file.

## Project Converter Resources

---

The project converter contains some resources you can edit in ResEdit if necessary.



## NTK Project Converter

The STR# resource with ID 1000 contains the mapping from MacNTK platform file names to WinNTK platform file names. The current mapping is:

MessagePad®	->	MsgPad
Junior	->	MsgPad
Newton 2.0	->	Newton20
Dante	->	Newton20

The STR# resource with ID 1001 contains a list of common file suffixes to be removed from a Macintosh filename before a DOS file suffix is added. On DOS and Windows machines, a file's suffix is a key indicator of its type. On a Macintosh computer, a file's suffix is largely conventional and ad hoc. The converter removes the following conventional file suffixes and replaces them with DOS/Windows suffixes:

".f"	".res"	".rsrc"	" rsrc"
".t"	".text"	".txt"	".π"
" π"	".prj"	".proj"	".stream"

## Filename Changes

---

The converter removes some conventional Macintosh filename suffixes and replaces them, when appropriate, with Windows file-type suffixes, as described in the previous section. It also removes from each name any characters that are invalid in Windows filenames.

If a file's name changes as part of the conversion process, internal references to those files are not changed. Instead, an entry that maps the old name to the new name is written to a file named NameMap.txt. Thus, if a filename is changed from MyLayout.t to MyLayout.lyt, an entry containing both names is written to NameMap.txt. When Win NTK is called upon to open a file, it first looks for it with the specified name. If the file cannot be found, it opens the NameMap.txt file to see if the file's name has changed. If so, it performs another search using the mapped name.

## NTK Project Converter

Macintosh, Newton, Newton Toolkit, and MessagePad are trademarks of Apple Computer, Inc., registered in the United States and other countries. Windows is a registered trademark of Microsoft Corporation.