Audacious - OLD, PLEASE USE GITHUB DISCUSSIONS/ISSUES - Bug #859

.wsz skins cannot be used on Windows

January 10, 2019 05:13 - John Lindgren

Status:	New	**Start date:**	January 09, 2019
Priority:	Minor	**Due date:**	
Assignee:		**% Done:**	0%
Category:	win32	**Estimated time:**	0.00 hour
Target version:			
Affects version:	3.10.1		

Description

The current scheme uses the external "unzip" program which generally isn't present on Windows.

We could ship an "unzip.exe" along with the Windows build, but it would be better if we could read the .wsz/.zip files directly through the VFS layer, without running an external .exe and creating temporary files on the C: drive.

History

#1 - January 10, 2019 05:14 - John Lindgren

- Description updated

#2 - March 09, 2019 14:19 - Carlo Bramini

It is possible to handle directly ZIP files on Windows with some API, but in my opionion, since Audacious is already released with zLib, perhaps it would be worth to just import the small unzip.c and unzip.h from zlib sources, into contrib/minizip:

https://github.com/madler/zlib/tree/master/contrib/minizip

and free all platforms from the dependency to an external tool.
In this way, the skins will be loaded directly in memory from the ZIP file, without the need to unpack them on disk, like you said.

Sincerely.

#3 - March 13, 2019 16:35 - John Lindgren

minizip looks like a good option.

#4 - May 04, 2019 21:28 - Hieronymous Anonymous

Has this bug been fixed?

#5 - November 21, 2019 10:49 - Ariadne Conill

What would URIs look like in a zipfile?

Something like file:///home/ariadne/foo.zip/bar.mp3 ?

#6 - November 21, 2019 16:29 - John Lindgren

I was thinking of something along the lines of:

```
VFSFile foo ("file:///home/ariadne/foo.zip", "r");
ZipReader reader (foo);
```

```
VFSFile bar = reader.open ("bar.mp3");
```

And ZipReader would provide its own VFSImpl functions (e.g. fread/fseek/etc.) that would wrap the minizip functions.

But I haven't worked it out in detail.

#7 - November 22, 2019 10:39 - Ariadne Conill

Well, the reason I ask is because some module sets are distributed as archives.

So, I think instead of using minizip, it makes more sense to just use libarchive and have a VFS plugin to do it. I just am not quite sure how the API should be, perhaps we need to extend VFS API to handle containers of files somehow.

My suggestion right now would be to abstract VFS API to add directory tools, where we can iterate over contents and also fopenat(3) using the "directory" handle as the reference. In stdio this means you use a DIR, in the theoretical libarchive adapter, it would be whatever libarchive uses.

#8 - November 22, 2019 19:54 - John Lindgren

I've no problem with supported other types of archives as well.

Maybe we could start with an ArchiveReader class providing something like:

```
ArchiveReader (VFSFile && archive_file);
Index<String> read_folder (const char * path);
VFSFile open (const char * path, const char * mode);
```

#9 - December 22, 2019 22:44 - Carlo Bramini

Libarchive provides many types of archives, for both reading and writing.
If I can say my opinion, it seems a bit exagerated for reading .wsz files which are known to be just ZIP archives.
In that case, you will be forced to set libarchive as a required dependency, otherwise the player will loose the ability to access to compressed skins if it is absent.
Perhaps you may consider to support libarchive as an option and allow to decompress .wsz with minizip.
Afterall, if you consider the size in bytes of unzip.c/h, it does not seem a big problem to me.

#10 - January 01, 2020 02:48 - Ariadne Conill

I think adding libarchive as a dependency is fine tbh. It's already on most systems and builds fine on Windows.