

Technical Hurdles for Win32 rsync port

Asked 16 years, 3 months ago Modified 4 years, 11 months ago

Viewed 5k times



9

Despite primarily being a windows user, I am a huge fan of rsync. Now, I don't want to argue the virtues of rsync vs any other tool...this is not my point.



The only way I've ever found of running rsync on windows is via a version that is built to run on top of Cygwin, and as Cygwin has issues with Unicode, so does rsync.



Is anyone familiar enough with the workings of rsync to say if there are any real technical programming hurdles to porting rsync to a native Win32 binary?

Or is it maybe that there has just never been enough interest from windows users to care to port it over?

Partly I ask because I'm am considering trying to take on the task of starting a port, but I want to make sure there's not something I'm missing in terms of why it may not be possible.

winapi

rsync

porting

Share

Improve this question

Follow

edited Jan 23, 2020 at 22:14



Christophe

73.2k ● 7 ● 82 ● 165

asked Aug 29, 2008 at 15:33



Adam Haile

31.3k ● 60 ● 195 ● 290

5 Answers

Sorted by:

Highest score (default)



5

The way that windows locks open files might cause an issue requiring you to hook into the Volume Shadowcopy Service.



About two years ago this fellow ported the algorithm to C#. I haven't taken a look at the code (or the provided binary), but it might be a place to start looking or someone to try contacting.



<http://www.russiantequila.com/wordpress/?p=8>



Share Improve this answer

answered Aug 29, 2008 at 16:04

Follow



sig11

539 ● 5 ● 13

Quick update to speed things up for someone casually browsing: The author, @kolosy, put the source on github in 2009, and since then the only activity has been updates until mid-2010 by a Matthew Steeples:

github.com/MatthewSteeple/rsync.net – Tao Oct 20, 2011 at 15:07



1



I've been evaluating an effort to undertake a win32 port as well. I don't believe anything major would block it, but evidence from both the [rsync mailing list](#) and another discussion points to a heavy reliance on unix fork() system calls. Using threads appears the way to go for win32.

[Threads vs. Fork discussion](#)

Share Improve this answer

answered Sep 27, 2008 at 8:04

Follow



[Michael Regan](#)

1,598 ● 16 ● 17

"Using threads appears the way to go for win32" - or I/O Completion Ports if you want to scale really well...

– [Len Holgate](#) Apr 8, 2009 at 19:18



1



(disclaimer: i promise, i don't google myself, but google analytics brought me here)

i went through porting rsync to .net (sig11's link is my blog). there are no technical hurdles, just practical ones. as was already said, the code is rather... dense. difficult to follow, and complete lack of comments. i'm more than happy to make my work available, but unfortunately, since it was part of a commercial effort, it's not in significantly better shape.

i have, on a number of occasions, messed around with the idea of reverse-engineering the protocol and doing a

ground-up implementation that's wire-compatible with the existing one, but ... a bit cleaner to work with. i've even started a wiki to that effect, but... as you can see from the lack of contents there, other item have taken priority. if anyone would like to work with me on this, that may be the impetus i need to get going.

the concept of the tool is great, as is the functionality it offers, however it's rather limited outside the *ix space, and could definitely benefit from an api.

wiki link for reference:

http://www.russiantequila.com/wiki/index.php?title=Main_Page

Share Improve this answer

answered Apr 8, 2009 at 15:41

Follow



kolosy

3,099 ● 3 ● 30 ● 49

Kolosy, I would love to get a hold of a copy of what you've got. Maybe even lend a hand if I can. I agree, a cleaner version with an API would be great, but honestly at this point, my first priority is a Windows version that doesn't rely on Cygwin – [Adam Haile](#) Apr 8, 2009 at 21:19

I'll help out however, I would like to implement it in C/C++ and I would also like to create a LIB version of the code so that we could separate it and use it in a native Win32/GUI as well as a Windows service with GUI configuring the service. – [Eric](#) Jul 27, 2012 at 15:35

Have you seen this:



0



<http://www.itefix.no/i2/taxonomy/term/39>

I have used cwrsrc without any problem (and with the much of the usual cygwin misery), but I haven't had any need for unicode filenames, so I've not seen that problem.

I don't really know why there isn't a native Win32 port, but I did look at the source a while back because I implemented a similar delta-copy system in C#. As one would expect from the world of brilliant *nix hackers, the source is largely single-character variable names and a total absence of comments, which isn't terrible helpful and might be rather off-putting to would-be porters.

Share Improve this answer

answered Aug 29, 2008 at 16:23

Follow



[Will Dean](#)

39.5k ● 11 ● 92 ● 118



0



I would really appreciate a port of rsync to MS-Windows such that it can be built using Visual Studio. I am encountering various protocol errors at random, somewhat intermittently. I am using rsync to distribute sw to a grid of around 200 machines and typically get around a a dozen failures. I am using GCC 4.4.2 and the latest cygwin to build rsync v3.0.7. It would help me alot if I could experiment with a version that does not require cygwin. This is because the machines in the grid already have another cygwin-based app running that is a different version to the one I have.

Having spent some time on the rsync mailing list opinion seems to be divided as to cause of protocol errors on MS-Windows. Some say it is a bug in rsync where it failed to do a clean socket shutdown, a bug that was fixed a while ago. Others say that it is a fundamental protocol error in rsync where the client does not tell the server that it is finished, it just shuts down, causing MW-windows servers to get a RST signal on the socket, something that does not happen on Unix.

[Share](#) [Improve this answer](#)

answered Jun 4, 2010 at 12:08

[Follow](#)



[Andrew Marlow](#)

1
