

Adapting FancyLibs & MinGW-Pkg from Debian 9 Stretch to 10 Buster, Mingw-7 to Mingw-8

Introduction

About every 2 years Debian introduces a new release.

At the time I'm writing this Debian [Buster](#) is on it's way from Full Freeze to GA.

The following notes shall provide the order in which FancyLibs shall be adapted and released for a new Debian release or maybe an Ubuntu release.

Adapting FancyLibs to new Debian Release

The steps per sub project are the same as described for "HelloWorld projects" in [Adapting PBA to new Debian Release](#), with -o unix (before [MinGW](#) adaption) or -o "arch|unix|7" after MinGW adaption.

FancyLibs subproject rebuild order phase 1

The recommended order for phase 1 (before MinGW adaptations) is:

Project (subdir)	Archs (-o parameter)	Remarks general	Status & Remarks stretch/buster	Version with Buster
fancylog	arch 8		OK. in pba-rpmrelease, call to rpmsign can be reduced to: rpmsign --addsign -v --key-id 9...7 -D "%__gpg /usr/bin/gpg" ../foobar-1.2.3-1.src.rpm	1.0.3-5
cute-headers	arch 8	Data only, could be done anytime	OK	1.6.1-9
fancylibs	arch		OK. libstdc++-8-dev	3.7.3-1
fancydb	-	Unix only	OK	0.6.5-7
fancydrawing	arch		OK	0.6.2-3
fancygpsdclient	unix		OK	0.4.3-5
fancyhttpd	arch		OK	0.8.0-2
fancylogd	unix		OK	0.1.0-3
fancylogview	.	Debian unix & amd64 only	Half OK: Does not build for armhf, maybe never did, libqt4-dev seems incomplete for arm	0.3.0.5
fancypkcs11	unix	Debian unix & amd64 only	Half OK: Does not build for armhf, maybe never did	0.4.1-1
fancypcsc	unix	Debian unix & amd64 only	Half OK: Does not build for armhf, maybe never did	0.2.5-2
fancymodbus	unix		OK Required minor adaption, with libmodbus-dev going from 3.0.6 to 3.1.4	0.2.4-3
fancysax	arch		OK	0.6.1-7
fancyqt4	arch		OK	0.9.3-3
fancyqt4widgets	arch		OK	0.5.2-2
fancysane	unix		OK, but libsane-dbg not available for buster. Reduced that to suggestion	0.3.2-5
fancyqmi	unix	Debian unix & amd64 only	Half OK: Does not build for armhf, maybe never did	0.2.1-5
fancyssl-tools	unix		OK?	0.4.0-5

Tests with released Buster on 2019-08-02 showed that all unix builds work for amd64, most work for armhf.

MinGW-Pkg rebuilds

Stretch came with Mingw 7, Buster comes with Mingw 8, so we should rebuild that carefully, pulling up versions of libraries.

Order based on [mingw 6 rebuild order](#)

Project (mingw-pkg subdir pattern)	Dep.	Version in mingw-pkg MinGW-7 Stretch	Version in mingw-pkg MinGW-8 Buster	Buster status
------------------------------------	------	--------------------------------------	-------------------------------------	---------------

(osslsigncode)	-	-	-	ignored
mingw-w64-binutils / binutils-mingw-w64		(2.27.90.20161231-1+7.4)	(2.31.1-11+8.3)	provided by Debian now
(mingw-w64-libmysq)		(6.1.11-1clazzespba1)	(6.1.11-1clazzespba2)	obsolete since mingw-6, empty now
mingw-w64-libmariadb		-	-	not required for now
mingw-w64-gcc-rt		6.3.0, uses Debian's g++-mingw64-*	8.3.0 (-6+21.3-deb10u1), uses Debian's g++-mingw64-*	OK
mingw-w64-zlib	mingw-w64-gcc-rt	1.2.11	1.2.11	OK after rules changes: common-build-indep to build-indep, common-install-indep to install-indep
mingw-w64-openssl	mingw-w64-zlib	1.1.0f-0	1.1.1c	OK
mingw-w64-libpng	mingw-w64-zlib	1.6.31	1.6.37	OK? libtool: link: x86_64-w64-mingw32-gcc -O2 -Wall -o .libs/pngfix.exe contrib/tools/pngfix.o ../libs/libpng16.dll.a -lz -L/usr/x86_64-w64-mingw32/lib libtool: error: Could not determine the host path corresponding to libtool: error: '.../mingw-pkg/trunk/mingw-w64-deb/mingw-w64-libpng-1.6.37/build/.libs' libtool: error: Continuing, but uninstalled executables may not work. ...
mingw-w64-libjpeg	mingw-w64-libpng	9b	9c	OK
mingw-w64-libtiff	mingw-w64-libjpeg	4.0.8	4.0.10	OK
mingw-w64-libmng	mingw-w64-libjpeg	2.0.3-2	2.0.3-3	OK
mingw-w64-bzip2	mingw-w64-gcc-rt	1.0.6-3	1.0.6-4	OK
mingw-w64-libboost	mingw-w64-bzip2	-	-	(ignored)
mingw-w64-libiconv	mingw-w64-gcc-rt	1.15-1	1.16-1	OK
mingw-w64-libxml2	mingw-w64-libiconv	2.9.3-1	2.9.9-1	OK
mingw-w64-gettext	mingw-w64-libxml2	0.19.8-1	0.20.1-1	OK New patch pthread_sigmask.m4.patch based on this post .
mingw-w64-qt4-qmake		4.8.7-0clazzespba1	4.8.7-1clazzespba0	OK Builldep change: libstdc++-6-dev to libstdc++-6-dev
mingw-w64-qt4	mingw-w64-qt4-qmake, many of above	4.8.7-0clazzespba3	4.8.7-1clazzespba0	TBC – Building without SSL support since mingw-7. Needed these: dpkg --add-architecture i386 apt-get update apt-get install wine32
mingw-w32-qt4-qmake		4.8.7-0clazzespba1	4.8.7-1clazzespba0	Builldep change: libstdc++-6-dev to libstdc++-6-dev OK
mingw-w32-qt4	mingw-w32-qt4-qmake, many of above	4.8.7-0clazzespba3	4.8.7-1clazzespba0	TODO Building without SSL support since mingw-7.
mingw-w64-jni-headers		1.6.0-4clazzespba5	1.6.0-4clazzespba6	OK. Using headers from OpenJDK-11 now. Should we raise the version to 11?

TODO FancyLibs subproject rebuild order phase 3

The recommended order for phase 2 (before MinGW-8 adaptions) is:

Project (subdir)	Archs (-o parameter)	Buster Status	Version with MinGW-8 package for Buster	Remarks, changes
------------------	----------------------	---------------	---	------------------

fancylog	- (arch 8)	OK	1.0.3-7	
cute-headers	- (arch 8)	OK	8_1.6.1-10	
fancylibs	- (arch 8)	OK	3.7.3-2	In debian-mingw-8/rules: # see https://bugs.debian.org/cgi-bin/bugreport.cgi?bug=930492 w32pcdir = /usr/i686-w64-mingw32/lib/pkgconfig w64pcdir = /usr/x86_64-w64-mingw32/lib/pkgconfig ... common-build-indep:: cd \$(w32builddir) && \$(MAKE) arch="i686-w64-mingw32" PKG_CONFIG_LIBDIR=\$(w32pcdir) cd \$(w64builddir) && \$(MAKE) arch="x86_64-w64-mingw32" PKG_CONFIG_LIBDIR=\$(w64pcdir)
fancydrawing	- (arch 8)	OK	0.6.2-5	In debian-mingw-8/rules: # see https://bugs.debian.org/cgi-bin/bugreport.cgi?bug=930492 w32pcdir = /usr/i686-w64-mingw32/lib/pkgconfig w64pcdir = /usr/x86_64-w64-mingw32/lib/pkgconfig ... common-build-indep:: cd \$(builddir) && PKG_CONFIG_LIBDIR=\$(w64pcdir) \$(MAKE) arch=\$(arch) cd \$(builddir32) && PKG_CONFIG_LIBDIR=\$(w32pcdir) \$(MAKE) arch=\$(arch32)
fancygpsdclient	- (unix 8)	OK	0.4.3-7	debian-mingw-8/rules adapted like above
fancyhttpd	- (arch 8)	OK	0.8.0-3	debian-mingw-8/rules adapted like above
fancylogd	- (unix 8)	OK	0.1.0-6	debian-mingw-8/rules adapted like above Also had to add FANCYLOGD_COMMON_API in fancylogdVersion.cpp
fancylogview	- (nsis directory)	OK	0.3.0.6	Deferred, depends on QT
fancykcs11	- (unix only)	-	-	Probably never build for MinGW, despite debian-mingw-?/ directories
fancykcs11	- (unix only)	-	-	
fancypcsc	- (arch 8)	OK	0.2.5-4	
fancymodbus	- (unix only)	-	-	
fancymodbus	- (unix only)	-	-	
fancysax	- (arch 8)	OK	0.6.1-9	debian-mingw-8/rules adapted like above
fancyqt4	- (arch 8)	OK	0.9.3-5	debian-mingw-8/rules adapted like above
fancyqt4widgets	- (arch 8)	OK	0.5.2-3	debian-mingw-8/rules adapted like above
fancysane	- (unix only)	-	-	
fancysane	- (unix only)	-	-	
fancyqmi	- (unix only)	-	-	
fancyqmi	- (unix only)	-	-	
fancyssl-tools	- (arch 8)	OK	0.4.0-6	debian-mingw-8/rules adapted like above

Obsolete subdirs

Project (subdir)	Remarks
base64	never even built for stretch
gpsfake-runner	shell scripts only
jnix	never even built for stretch
fancyxmlsock	Looks like old experimental code now in fancysock