Exiv2 - Patch #993

Generating svn_version.h with CMake

21 Sep 2014 11:14 - Daniel Kaneider

Status:	Closed	**Start date:**	21 Sep 2014
Priority:	Normal	**Due date:**	
Assignee:	Robin Mills	**% Done:**	100%
Category:	build	**Estimated time:**	0.00 hour
Target version:	0.25		

Description

Hi,

I took the liberty to provide you a patch to enable the svn version detection in a (almost) clean way. It works for CMake only (including windows), and there is a fallback if the source directory is not under svn version control.

On linux, svn should be automatically detected. On windows, svn can be either on the system path, detected via installed TortoiseSVN client (including binaries), or using the following additional command on cmake:

cmake.exe ... -DCMAKE_PROGRAM_PATH=C:\Subversion-Directory\bin

Pay attention, that the cygwin svn binary doesn't work in combination with CMake.

There is still a small room for improvement: if header files (svn_version.h) are changed, then all files using an import to that file will get recompiled. This behaviour was already present. You might think about changing that it to an external definition in a header (extern int svnVersion;), which then gets resolved at link time (char g_GIT_SHA1[] = "123";). This is how we do it in LuminanceHDR, and on svn changes it reduces the files required to recompile to basically one.

Tested on Windows VS 2012 x64, with CMake 3.0.2. The patch is against the 3364 version.

Best,
Daniel

Associated revisions

Revision 3396 - 30 Nov 2014 07:28 - Robin Mills

#993. svn_version.h generated in wrong directory for out of source cmake builds.

History

#1 - 21 Sep 2014 11:47 - Robin Mills

- Status changed from New to Assigned

- Assignee set to Robin Mills

- Target version set to 0.25

Thanks, Daniel. I appreciate this very much. I will commit this (or a slightly modified version) shortly. However I'd like to hear your response to my thoughts before I commit:

1. I don't think we can remove ./svn_version.sh because it's used by the autotools to generate svn_version.h
2. Will this code keep rewriting svn_version.h which will cause recompilation of version.cpp (and possibly other code). One of the features of svn_version.sh is to grep svn_version.h and only update svn_version.h when necessary.
3. Does svn_version.h get updated if required on every build. Or does this code only update svn_version when we run/rerun CMake?
4. If CMake/Cygwin/svn are in collision, should we continue to use svn_version.sh for Cygwin (as I believe we have to retain it for the autotools anyway)?
5. I'm uneasy about requiring TortoiseSVN (which I really like) or an additional command-line -DCMAKE_PROGRAM_PATH argument. I'm sure we can invoke svn and test ERRORLEVEL. Perhaps we need a little batch file svn_version.bat (very similar to svn_version.sh) to generate svn_version.h from DOS. I can run svn_version.bat from CMake and Visual Studio. One reason I didn't do that was because grep is not guaranteed available on DOS and I've never understood the DOS find command.

Thanks for doing this, Daniel. Very good work.

Robin

#2 - 03 Oct 2014 21:50 - Daniel Kaneider

Robin,

First I'm glad that the web site appears to be up and running. AWS sounds definitely nice ;) If you can't view my attached patch, plz let me know. See my comments below

> Thanks, Daniel. I appreciate this very much. I will commit this (or a slightly modified version) shortly. However I'd like to hear your response to my thoughts before I commit:
>
> 1. I don't think we can remove ./svn_version.sh because it's used by the autotools to generate svn_version.h

The solution for the future would be to drop autotools and to just relay on CMake. Since I guess you want to postpone that step, I suggest to incorporate those 5 lines of code of the svn_version.sh file into autotools/config. Then you should be able to remove it without any problems

> 1. Will this code keep rewriting svn_version.h which will cause recompilation of version.cpp (and possibly other code). One of the features of svn_version.sh is to grep svn_version.h and only update svn_version.h when necessary.

As far as I tested it a few weeks ago, CMake will take care of that. This means that the file gets recompiled if and only if the svn version gets changed. CMake takes care about a lot of things like that.

> 1. Does svn_version.h get updated if required on every build. Or does this code only update svn_version when we run/rerun CMake?

You have to rerun CMake, which you should do every time after an svn update

> 1. If CMake/Cygwin/svn are in collision, should we continue to use svn_version.sh for Cygwin (as I believe we have to retain it for the autotools anyway)?

A small misunderstanding: Just if svn is used through cygwin without the cygwin console (you get thousands of warnings if you do so anyway) you get problems. Using the regular cygwin pipeline should work (didn't test it).

> 1. I'm uneasy about requiring TortoiseSVN (which I really like) or an additional command-line -DCMAKE_PROGRAM_PATH argument. I'm sure we can invoke svn and test ERRORLEVEL. Perhaps we need a little batch file svn_version.bat (very similar to svn_version.sh) to generate svn_version.h from DOS. I can run svn_version.bat from CMake and Visual Studio. One reason I didn't do that was because grep is not guaranteed available on DOS and I've never understood the DOS find command.

The thing here is that you gonna need to tell CMAke somehow where to find the svn binary. For windows I suggest installing the command line svn version and making sure it can be found on the command line. Thats it. Normally the installers already take care about that. Alternatively you can use the methods described above. The DCMAKE_PROGRAM_PATH might be interesting for your Jenkins system.
I don't get your point with the .bat file. It does not make any sense to me. svn_version is currently not generated for VS, so why should it? Official packagers should use CMake, and public folk doesn't need svn version generation.

--Daniel

#3 - 04 Oct 2014 14:22 - Robin Mills

I'm going to go along with you on this, Daniel. I'll submit your patch on Sunday.

I don't think we're going to be dropping autotools any time soon. However our long-term goal is to make CMake our primary build platform. We're not there yet.

#4 - 05 Oct 2014 19:08 - Robin Mills

Patch submitted: r3371

I made two changes to the patch:

1. I am going to continue with svn_version.sh for the autotools build. It's called from src/Makefile which is not a config generated file.
2. There was a block of code for #722 removed by the patch. I've retained that code. If you have a strong reason to remove that code, please let me know and I'll remove it.

```
IF( NOT MSVC )
  # Issue #722: out of source builds compiled against standard include files such as /usr/local/lib/include/ex
iv2
  # do not use CREATE_SYMLINK or CMAKE_CAN_SYMLINK as they don't work on CYGWIN
  EXECUTE_PROCESS( WORKING_DIRECTORY ${CMAKE_CURRENT_BINARY_DIR}           COMMAND ln -sf ${CMAKE_CURRENT_SOUR
CE_DIR}/src exiv2)

  IF( EXIV2_ENABLE_BUILD_SAMPLES )
    EXECUTE_PROCESS( WORKING_DIRECTORY ${CMAKE_CURRENT_BINARY_DIR}/samples COMMAND ln -sf ${CMAKE_CURRENT_SOUR
CE_DIR}/src exiv2)
```

```
  ENDIF( EXIV2_ENABLE_BUILD_SAMPLES )

  IF( EXIV2_ENABLE_BUILD_PO )
    EXECUTE_PROCESS( WORKING_DIRECTORY ${CMAKE_CURRENT_BINARY_DIR}/po      COMMAND ln -sf ${CMAKE_CURRENT_SOUR
CE_DIR}/src exiv2)
  ENDIF( EXIV2_ENABLE_BUILD_PO )
ENDIF()
```

I apologise for confusing you about svn_version.bat. I have thought about having a little batch file which has the same purpose an svn_version.sh and would be called by the Visual Studio 'native' project files. I'm not going to bother with this. You've put something very nice into the CMake files to generate svn_version.h and that's great. Let's leave it be. The primary purpose of svn_version.h is for developers and I think it's OK for this not to be supported in Visual Studio 'native' builds.

Thanks very much for your patience with this, Daniel. As you know, we've relocated the Exiv2 server and the project has been "off the air" for 2 weeks.

Robin

#5 - 05 Oct 2014 19:19 - Robin Mills

- Status changed from Assigned to Resolved

#6 - 30 Nov 2014 07:29 - Robin Mills

Fix submitted: r3396. svn_version.h was being generated in the wrong directory for out of source cmake builds.

#7 - 30 Nov 2014 07:30 - Robin Mills

- Subject changed from svn_version for CMake windows (patch) to Generating svn_version.h with CMake

#8 - 08 May 2015 16:13 - Robin Mills

- % Done changed from 0 to 100

#9 - 21 Jun 2015 16:41 - Andreas Huggel

- Status changed from Resolved to Closed

Files

cmake-svn.patch	3.86 KB	21 Sep 2014	Daniel Kaneider