

U++ - Feature #1190

post-commit should update version number

07/17/2015 07:48 PM - Zbigniew Rebacz

Status:	Rejected	Start date:	07/17/2015
Priority:	Normal	Due date:	
Assignee:	Zbigniew Rebacz	% Done:	0%
Category:	General	Estimated time:	0.00 hour
Target version:		Spent time:	0.00 hour
Description			
<p>Now our visioning system doesn't work correctly. After doing "svn up" - ide/version.h doesn't contain version number. We can change it by using server side svn scripts.</p> <p>I have attached script file to this topic (file post-commit). It should be added in repo/hooks/post-commit. Of course executable rights is required - chmod 755 post-commit.</p> <p>External resources http://svnbook.red-bean.com/en/1.7/svn.ref.reposhooks.post-commit.html - post commit documentation</p>			

History

#1 - 07/17/2015 09:52 PM - Jan Dolinár

Did you test the script? I'm afraid it won't work :(I studied this couple years ago and found out that there is no way to modify the commit content from hook. See for example the discussion here: <http://stackoverflow.com/questions/650168/post-commit-hook-to-update-a-file-under-version>

#2 - 07/17/2015 10:06 PM - Zbigniew Rebacz

So, it seems that we should using svn keyword? Then get correct content from it - <http://svnbook.red-bean.com/en/1.4/svn.advanced.props.special.keywords.html>.
What do you think?

#3 - 07/17/2015 10:12 PM - Zbigniew Rebacz

- Status changed from Patch ready to In Progress
- Assignee changed from Miroslav Fidler to Zbigniew Rebacz

But it seems that commiter needs to modify this file.

#4 - 07/17/2015 11:54 PM - Zbigniew Rebacz

- Status changed from In Progress to Rejected

#5 - 07/18/2015 12:26 PM - Jan Dolinár

Yes, svn keywords unfortunately don't work very well (at least for what you'd like to do) either...

The only way correct way to do this is to use external tools, usually at build time, just as the link you posted metnrions.

One thing we could do, is to add a special build step into the ide package, that would check the version and modify the version.h file before building. That

of course doesn't help much if the file is used from different package.

Another option would be to hardcode this behavior into theide itself. The version of sources might be exposed to any package simply by setting a macro, e.g. UPP_VERSION=8765. Similar to the build time/date macros. This value could then override the value in ide/version.h.

IMHO, the second way is better, because it is more generic and provides the version information to any package, not just theide.

What do you think? Also: Mirek, do you have any comments on this?

#6 - 07/20/2015 01:02 PM - Zbigniew Rebacz

- Status changed from Rejected to New

#7 - 07/31/2015 11:49 AM - Miroslav Fidler

I never considered this too important... :)

There is also the issue that ide could be build from source not in svn.

However, I think it could be possible to extend build info. We can run svn info on main package dir if it has .svn subfolder, then add lines

```
#define bmSVN_REVISION 8774
#define bmSVN_URL      "svn://www.ultimatepp.org/upp/trunk/uppsrc/ide"
```

The about dialog(s) then could #ifdef for these macros.

Later we can add GIT support there (but I am not sure what GIT informations are relevant here... :)

#8 - 07/31/2015 01:54 PM - Miroslav Fidler

- Status changed from New to Ready for QA

...resolved

See reference/build_info

#9 - 07/31/2015 08:42 PM - Jan Dolinár

I'm not sure, if it is correct to write fixed values into generated Makefile. If it were used with source in svn, which were updated after the makefile was generated, it would result in incorrect information. It should be easy to take care about the corner-case and do something like (pseudocode):

```
if (acting on svn sources)
    svnversion $dir
else
    echo $version_generated_by_ide
```

It is probably ok to leave it as is for now, but I'll try to fix it soon.

#10 - 07/31/2015 09:06 PM - Jan Dolinár

Miroslav Fidler wrote:

| Later we can add GIT support there (but I am not sure what GIT informations are relevant here... :)

Something like this should be enough for most cases when using git:

```
git rev-parse HEAD # shows last commit
git config --get remote.$(git config --get branch.$(git rev-parse --abbrev-ref HEAD).remote).url # shows upstream url, if available
```

#11 - 04/09/2016 09:24 PM - Zbigniew Rebacz
- Status changed from Ready for QA to Rejected

Files			
post-commit	219 Bytes	07/17/2015	Zbigniew Rebacz