Subject: [PROPOSAL] U++ version macro
Posted by dolik.rce on Sun, 10 Jul 2011 15:43:13 GMT
View Forum Message <> Reply to Message

Hi!

Sometimes I want to have an access to information about what sources were used when compiling my application (e.g. to show it in About box  ). As of now, the only way to get this piece of information is to include ide/version.h and use IDE_VERSION macro. As you can see, that is not really intuitive and has some drawbacks, for example if you export project sources, the version.h won't be exported (unless you add ide package to your project, which is absurd overkill).

I propose to change the name of this version macro and also move it into Core package. Following solution should do it and maintains backward compatibility:

Add file Core/version.h:#ifndef UPP_VERSION
  #define UPP_VERSION 3629
#endifChange ide/version.h to:#include<Core/version.h>
#ifndef IDE_VERSION
  #define IDE_VERSION UPP_VERSION
#endif
Finally, add #include "version.h" into Core/Core.h.

What do you think?

Best regards,
Honza

Subject: Re: [PROPOSAL] U++ version macro
Posted by mirek on Mon, 11 Jul 2011 13:54:03 GMT
View Forum Message <> Reply to Message

dolik.rce wrote on Sun, 10 July 2011 11:43Hi!

Sometimes I want to have an access to information about what sources were used when compiling my application (e.g. to show it in About box  ). As of now, the only way to get this piece of information is to include ide/version.h and use IDE_VERSION macro. As you can see, that is not really intuitive and has some drawbacks, for example if you export project sources, the version.h won't be exported (unless you add ide package to your project, which is absurd overkill).

I propose to change the name of this version macro and also move it into Core package. Following solution should do it and maintains backward compatibility:

Add file Core/version.h:#ifndef UPP_VERSION
  #define UPP_VERSION 3629
#endifChange ide/version.h to:#include<Core/version.h>
#ifndef IDE_VERSION

```
  #define IDE_VERSION UPP_VERSION
#endif
```

Finally, add #include "version.h" into Core/Core.h.

What do you think?

Best regards,
Honza

Well, why not, but it would only work for "released" sources, not for svn....

Mirek

---

Subject: Re: [PROPOSAL] U++ version macro
Posted by dolik.rce on Mon, 11 Jul 2011 15:06:24 GMT
View Forum Message <> Reply to Message

mirek wrote on Mon, 11 July 2011 15:54Well, why not, but it would only work for "released" sources, not for svn....

Mirek

Sure, I'm aware of the SVN limitation. But SVN is mainly for developers. Skilled developer can surely write a simple script to keep the file up to date  Something along the lines of #!/bin/sh

```
UPPROOT=/path/to/upp
UPPREV=`svn info $UPPROOT | sed -n 's/Revision: \([0-9]*\)/\1/p'`
echo "#ifndef UPP_VERSION
  #define UPP_VERSION $UPPREV
#endif" > $UPPROOT/uppsrc/Core/version.h
```

Also, the update system in theide can take care of this (as it does now with IDE_VERSION), even when using SVN repository.

Honza

---

Subject: Re: [PROPOSAL] U++ version macro
Posted by mirek on Sat, 16 Jul 2011 07:44:30 GMT
View Forum Message <> Reply to Message

What about:

During build, if theide finds specific file (say 'version') inside package and svn is present, it would set it to the current svn revision

```
#define PACKAGE_VERSION 3333
```

---

Subject: Re: [PROPOSAL] U++ version macro
Posted by dolik.rce on Sun, 17 Jul 2011 14:51:34 GMT
View Forum Message <> Reply to Message

mirek wrote on Sat, 16 July 2011 09:44What about:

During build, if theide finds specific file (say 'version') inside package and svn is present, it would set it to the current svn revision

#define PACKAGE_VERSION 3333

Do I understand correctly that you would like to do this in theide builder code? Well, that would be just an additional bonus, as it could be used in end user packages as well... as long as they use SVN.

Honza