

---

Subject: Re: Possible solution of "icpp problem"

Posted by [dolik.rce](#) on Sun, 28 Aug 2016 18:37:37 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Sun, 28 August 2016 17:49 Well, first of all, let us not to forget that the problem we really need to have solved is to force linker to include some .obj files from .lib (or .o from .a ... :) even if none of symbols is referenced from the rest of the code. What I am proposing is basically putting a dummy call from header to such .obj, just to have it 'activated'.

For me the issue always was "there is something non-standard that must be done when compiling/linking". If a static dummy object everything that needs to be done to turn this into "everything works with fine with all common C++ build systems" situation, than it will make me really happy :)

If the code stays as simple as you suggested, you could easily hide it into pair of simple macros. That way, user can just add one line to a file that would previously be .icpp and another line to header or anywhere else to force correct linking. It would be handy, for example in the Skylark case I already mentioned.

Honza

---