

---

Subject: Re: Small fixe for OPENBSD package organizer configuration

Posted by [mirek](#) on Fri, 24 Apr 2020 12:07:54 GMT

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

---

amrein wrote on Fri, 24 April 2020 11:51 mirek wrote on Fri, 24 April 2020 08:04 There should be no absolute paths listed in .upp files, that is the role of build methods (to act as "interface" between the source tree and build machine).

Now well, if these are autogenerated, maybe we can make it less ugly to detect OPENBSD when autogenerating them and add /usr/local just in that case?

(it would be better to detect in runtime, just in case that because of some miracle umks32 works, can we do that? What is the 'uname' output on OpenBSD?)

For me, adding internal include and internal libpath inside Package Organizer is the best way to have control over this without having to add hidden code to check for the OS and add /usr/local/include and /usr/local/lib if it's OpenBSD or any other new OS requiring this trick.

It's not user .bm includes list nor user .bm libpath list. It's an internal umk/ide/Core include and libpath and need to be added only for some OS like for instance here, OpenBSD.

Don't we already have include withing Package Organize (without the "When" clause)? I mean, if it already exist, why should we fight against it?

"need to be added only for some OS" is exactly the build method domain... This is the same as in Win32, where there is autodetection code for localizing C++ compiler folder and setting paths to it.

In either case, adding 4 lines of code to bm generator is much simpler than changing .upp structure, changing Package organizer GUI code and ide/Builders code....

Mirek

---