Subject: [PROPOSAL] Changes to packaging scheme
Posted by dolik.rce on Sun, 29 May 2011 11:41:28 GMT
View Forum Message <> Reply to Message

Hi everyone

I would like to propose a couple of changes in the packaging scheme and hear your opinions. I will talk mainly about ubuntu packages and arch PKGBUILDs, which I maintain, but the ideas are applicable to any packaging system.

As of now, we produce three packages: two packages for theide (normal and nogtk version) and 'upp' for all the sources. I would like to split 'upp' into several smaller packages. The reason is simple, not all parts of U++ are changed equally often. For illustration, last changed revisions of the top level directories in svn trunk: for r in trunk/*; do echo "$r: `svn info $r | sed -n 's/Last Changed Rev: \(.*\)/\1/p;'`"; done;# bold directories are those that are usually shipped to end users
archive: 1997
bazaar: 3464
benchmarks: 958
examples: 3375
reference: 3453
tutorial: 1700
uppbox: 3467
uppdev: 3418
uppsrc: 3467
uppsrc2: 1175
upptst: 2925As you can see, some of them are seldom updated, so downloading all of them each time is quite an overkill. So I propose to split the current upp package into several smaller ones (upp-bazaar, upp-uppsrc, upp-reference, ...). To ensure a) fluent transition and b) easy handling for users, there would be still an 'upp' package, but it would carry no files, just dependency on all the upp-* packages (i.e. what's called meta-package in most package managers).

The advantages of the proposed changes:
- Faster updates, smaller download sizes
- Possible to explicitly install only packages you want (e.g. only uppsrc)
- Less work for the packaging server (we can build packages only for parts that changed).

The possible disadvantages:
- More packages
- Possibly confusing for user

I am not sure if the first one could affect anyone negatively, but I listed it as possible problem, just to be fair. As for the second, it should be IMHO easily solved by big bold warning (in each packages description and on download page) saying that the best choice is to install 'upp' and let package manager do the rest of the work.

I plan to do some upgrades in the uppbox/lpbuild packaging scripts in the near future, so if there is a general agreement with this proposal, I would incorporate it as well. So what are your opinions? Are there any serious drawbacks that I overlooked?

Best regards,
Honza