Quote:ow thinking about it, maybe it would be actually possible to ship the whole SDK as part of package somehow? Like introducing "bin" subfolder of package or something like that.

If we take the Vulkan SDK as an example, the installation executable will set up some path variable. this assertion could be true for some other package, so installing it as part of the Ctrl like a bin directory is not, IMO, a good idea. It mean that all users which already have the SDK installed in a proper way will have 2 differents SDK located at 2 differents places.

Quote:I do not think there is a simple/clean way how to do that
Indeed, maybe the simplest way to do it is by simply, in case of Vulkan Ctrl, reference somewhere in the package that this one need to have a SDK (or lib) installed to work and, in downloading (or even starting compilation of code) TheIDE could just ask to the user if the install executable should be downloaded ("The LunarG Vulkan SDK is necessary to run this package, would you like to download the SDK ?") and as soon the user say "yes" then TheIDE download the executable (in case of windows) from official website and execute him. Reste of scenario is left to the user which will be able to configure all insallation setup.