Twitter Thread by Knarf



Knarf
@Knarf93

I'm in the position that I actually find npm / yarn the best ecosystem. Whenever I use something else I always end up stubbing my toe into something thats missing / feels wrong.

Ex. Cargo seems to neither have a concept of devDependencies nor peerDependencies.

@zkat__ I also can't understand why it wouldn't have an "add" command to add a new dependency. And I'm no fan of Toml, json is greate (easy to parse and build tooling around), and the better option in my opinion would be json5.

@zkat__ C / C++ seems to just not have language package managers. The linux / bsd crowd seem to have decided that the system package manager also should be the language package manager. Which might have been fine if every Linux distro used the same system package manager.

@zkat__ Instead we end up with a N x M problem. Where we have a bunch of different operating systems and they all support multiple system package managers. So there's no easy way of distributing, referencing and updating C / C++ packages.

@zkat__ It is also my opinion that the compiler / runtime should be a package dependency. I don't like Rust's split between rustup and cargo (they should have been one tool). Similarly it would be better if you added Node as a dependency to package.json, that way we wouldn't need NVM.

@zkat__ Lock-files are great, but I'm always surprised that they aren't built in a way so that Git can more easily automatically resolve merge conflicts. Maybe package managers could supply a Git hook for fixing merge conflicts in lock-files?

@zkat__ I'm not to happy that Cargo doesn't have a dedicated command for downloading dependencies. I don't want it to download all its dependencies when I run the build, I would want to do that beforehand as its own step. How else am I to cache the dependencies in buildpipelines / Docker

@zkat__ I still don't know how Go handles its dependencies. Whatever they did with requiring a GOPATH when it first came out was horrible. I feel like any new programming language that comes out should solve their package management first before releasing something into the public.

@zkat__ In fact I feel like any new programming language should be built around package management! Semver is okay, but not great. There should be no reason to manually have to set version numbers. But that would mean that the compiler would have to come up with a version number.

@zkat__ That should be possible if the language was built around supporting it.

@zkat__ I feel like every package manager should have a command to output their dependency tree as a graphviz Dot file, so that you could easily graph it. Especially if you have a monorepo with multiple workspaces.

@zkat__ And why don't package managers come with petter tooling around reviewing and upgrading dependencies? Let me easily get a list of dependencies and filesize. Give me a linter to ensure that packages gets updated.

@zkat__ Let me set max size for libraries so that I can ensure that I don't pull in to big libraries.

@zkat__ Yarn Berry's idea of committing the yarn executable to the Git repo so that it to is versioned (and therefor is versioned between developers / CI servers) is a great idea! How well it works in practice I have yet to see.

@zkat__ I'm not a fan of Yarn keeping a single lock file for all of it's workspaces in a monorepo. It makes building things inside docker a bit weird, it creates more opportunities for merge conflicts. I feel like there should be a better solution for this.

@zkat__ Lock-files in general feel like they store to much information. There should be an algorithm to reduce what information is needed, and to find a minimal set. I remember seeing a project that supposedly fixed this in another ecosystem.

@zkat__ This came out as a bit of a rant, I'm afraid that I could keep going for quite a while. Feel free to hit me up if you'd like to talk more about this. Then again you probably have know a lot more than me about all this having worked with building this kind of stuff.

@zkat__ @UnrollHelper