Tachyon

What is it?

Tachyon is a Meson-inspired build system that intends to be extreamly flexible and extensible (like Scons) but without sacrificing speed and simplicity. It uses lua to acheive this.

Name

Like Meson, it was also named after a subatomic particle. I first thought about photon(light and fast) but there was already a pretty popular project named photon. so I used tachyon.

How is it differant from Meson?

Language choice

Tachyon uses c++ instead of python. This is a bit counterintuitive since c++ is not nearly as portable as python or as easy to understand.

Tachyon makes up for this by using lua(jit). Not only is it faster because JIT but lua was designed to extend and to be extended.

Also the core is written in c++ because the core is pretty small.

Extensibility

Due to its use of lua(jit) you can extend Tachyon in whatever language you want as long as it can compile into a shared library, anything that compiles to lua or lua itself(pretty op).

This means you can extend lua in pretty much any language eg c++, c, go, rust, typescript etc

Very minimal

Tachyon is literally just a wrapper around ninja with some additional features. And features like executables are implemented by plugins. Heck, plugins are implemented by lua modules so you could technically rewrite this as a lua module.

Turing completeness

This is as bad as it is good. This means that I can do more things with Tachyon but that also means that a lot of people will reinvent the entire thing in pure lua(aka shoot themselves in the foot) or not make use of libraries provided by the project(eg using `Rule.new` and `Rule:generate` instead of using a language module)

Easier to deal with multiple languages

```lua
local cc = require'lang.c.compiler'
cc.executable(...)
local rust = require'lang.rust.compiler'
-- Languages can use whatever they want and aren't forced to use executables
rust.crate(...)
local cpython = require'cpython-tachyon' -- example of a module not part of tachyon
```

```
cpython.executable(...)
```